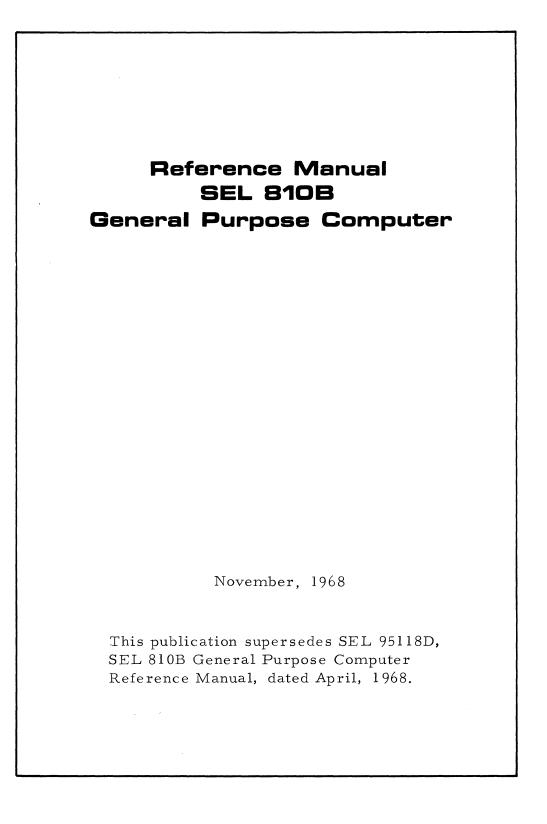
Reference Manual SEL 810B General Purpose Computer



6901 West Sunrise Blvd., Fort Lauderdale, Florida 33310 Area Code 305 587-2900

©1968. Systems Engineering Laboratories Printed In ILS A

LIST OF EFFECTIVE PAGES

The total number of pages in this ma	anual is 116,
consisting of the followin	g:

Issue

Page Number

Page Number

Issue

TitleOriginal
AOriginal
i thru vi
1-1 thru 1-10 Original
2-1 thru 2-26 Original
3-1 thru 3-12 Original
4-1 thru 4-6 Original
5-1 thru 5-8 Original
6-1 thru 6-20 Original
7-1 and 7-2 Original
A-1 and A-2 Original
B-1 thru B-4Original
C-1 and C-2 Original
D-1 and D-2 Original
E-1 and E-2 Original
F-1 thru F-10Original
G-1 and G-2 Original

TABLE OF CONTENTS

Section

I GENERAL DESCRIPTION

Title

	Introduction	1-1
	General Characteristics	1 - 1
	810B Computer	1-1
	Computer Options	1 - 1
	Standard Software	1-2
	Peripheral Devices	1-2
	Applications Programming.	1-2
	Computer Organization	1-2
	Memory Unit.	1-2
	Control Unit	1-4
		1-4
	Input/Output Unit	1-5
	SEL 810B Software System.	
	SEL 810B Assembly Program	1-8
	SEL 810B Loader	1-8
	SEL 810B FORTRAN IV	1-8
	SEL 810B Debug	1-8
	SEL 810B Update	1-9
	SEL 810B Library Package	1-9
	SEL 810B Maintenance Routines	1-9
	Power Fail Safe	1-9
II	MACHINE LANGUAGE PROGRAMMING	
	Introduction	2-1
	Memory Reference Instructions	2-1
		2-4
	Augmented Instructions	2-4
	Machine Language Instruction Set	
	Arithmetic Instructions	2-5
	Load/Store Instructions	2-8
	Branch/Skip Instructions	2-9
	Logical Instructions	2-13
	Registers Change Instructions	2-14
	Shift Instructions	2-17
	Control Instructions	2-19
	Input/Output Instruction	2-20
III	ASSEMBLY LANGUAGE PROGRAMMING	
	General Description	3-1
	Location Field.	3-1
	Operation Field	3-1
		3-1 3-1
	Address Field (Variable Field)	
	Comments Field	
	Identification Field	3-3
	Mnemonic Computer Instructions.	3-3
	Absolute Notations for Variable Fields	3-6
	Pseudo-Operation Instructions	3-6
	Summary of Pseudo-Operation Instructions	3-9
	Macro System	3-10
	Macro Prototype	3-11

TABLE OF CONTENTS (Cont'd)

Section	Title	Page			
IV	IN PUT/OUTPUT				
	General Description	4 - 1 4 - 2 4 - 5 4 - 5 4 - 5 4 - 5			
v	PRIORITY INTERRUPT SYSTEM				
	General Description	5 - 1 5 - 3 5 - 3 5 - 3 5 - 4 5 - 4			
VI	PERIPHERAL DEVICES				
	Introduction. Console Typewriter (Model No. 81-711-02A, Device No. 1) ASR-33 Programming. Paper Tape Reader (Model No. 81-510A, 300 cps - Device No. 2) Paper Tape Punch (Model No. 81-520A, 110 cps - Device No. 2) Paper Tape Punch (Model No. 81-520A, 110 cps - Device No. 2) Perforated Paper Tape Spooler (Model No. 80-530A) High Speed Paper Tape Punch/Reader System (Model No. 81-525A, Device No. 2) Magnetic Tape (Model No. 80-615 Series, Device No. 6 and 7) Magnetic Tape Programming High Speed Printer (Model No. 80-700 Series, Device No. 5) High Speed Line Printer Programming High Speed Line Printer Programming Punched Card Reader (Model No. 81-450A 400 cpm - Device No. 4) Punched Card Reader Programming X-Y Incremental Plotter (Model No. 81-810A and 81-812A, Device No. 11) X-Y Plotter Programming Movable Head Disc Storage (Model No. 81-653A, Device No. 13) Movable Head Disc Storage Programming Fixed Head Disc Storage (Model No. 81-654A, Device No. 13) Fixed Head Disc Storage Programming Priority Interrupts for Fixed Head Disc				
VII	OPTIONS				
	Program Protect and Instruction Trap (Model 81-080B) Variable Base Register (Model 81-042B) Stall Alarm (Model 81-043B) Auto Start (Model 81-041B) Table Top (Model 81-057B) Input/Output Parity (Model 81-210B) Index Register (Model 81-006B) 60 Hz Real-Time Clock (Model 81-031B)	7-1 7-1 7-2 7-2 7-2 7-2 7-2 7-2 7-2			
	APPENDIX A. SEL 810B Computer Word Formats	A-1			
	APPENDIX B.SEL Peripheral Device Octal Character CodesAPPENDIX C.SEL 810 Peripheral Device Command and Test Code Formats	B-1 C-1			

TABLE OF CONTENTS (Cont'd)

Section		Title	Page
	APPENDIX D.	SEL 810 Paper Tape Formats	D-1
	APPENDIX E.	SEL 810 Assembler Output Formats	E-1
	APPENDIX F.	Numerical Information	F-1
	APPENDIX G.	SEL 810B Instruction List Summary	G-1

LIST OF ILLUSTRATIONS

Figure

Title

Page

1-1	SEL 810B Block Diagram	1 -3
1-2	810B Computer Basic Data Formats	1-6
2 -1	Typical Memory Reference Instruction Word Format Diagram	2-3
2-2	Input/Output Instruction Word Format Diagram	2-5
2-3	AIP/AOP Instruction Execution Flow Chart	2-21
2-4	I/O Instruction Word Format	2-22
3-1	Example of Assembler Coding	3-1
4-1	Connection of Peripheral Units to the Computer	4-1
4-2	Input/Output Configuration and Computer Interface	4-3
4-3	Peripheral Device Bus Connections	4-4
5-1	Sample Program for Two Typewriters on the Same Standard	
	Output Interrupt Level	5-5
6-1	Paper Tape Data Flow Diagram	6-3
6-2	Magnetic Tape Format 0 Data Word	6-7
6-3	Track and Sector Layout	6-16
6-4	Movable Head Arrangement - Recording Surface	6-16
6-5	Head Position	6-16
6-6	Typical Head Positioning Time Chart	6-16
6-7	Fixed Head Track and Sector Layout	6-19
6-8	Fixed Head Arrangement - Recording Surface	6-19

LIST OF TABLES

Table

Title

2-1	Sample Listing
3-1	Example Address Field Entries
3-2	SEL 810B Mnemonic Instructions
3-3	SEL 810B Absolute Notation Formats
3-4	Summary of SEL 810B Pseudo-Operations
4-1	I/O Control Signals
4-2	Executive Times
5-1	Priority Interrupt Assignments
5-2	BTC Memory Location Assignments
5-3	Standard Interrupt CEU Bit Functions
5-4	Sample Assembler Interrupt Routine
6-1	Model 81-711-02A Console Typewriter Specifications

Page

LIST OF TABLES (Cont'd)

Table	Title	Page
6-2	Bit Coding for the CEU Command	6-2
6-3	Programming Routine for Console Keyboard Input and Console	
	Printer Output	6-3
6-4	Model 81-510A Paper Tape Reader Specifications	6-4
6-5	Model 81-520A Paper Tape Punch Specifications	6-4
6-6	Model 80-530A Paper Tape Spooler Specifications	6-4
6-7	Programming Routine for Copying Paper Tape	6-5
6-8	Model 80-615 Magnetic Tape Transport Specifications	6-6
6-9	CEU, Format 0, Second Word, Bit Functions	6-6
6-10	CEU, Format 1, Second Word, Bit Functions	6-7
6-11	TEU, Second Word, Bit Functions	6-8
6-12	Programming Routine for Magnetic Tape With BTC	6-10
6-13	CEU and TEU Second Word Bit Format for High Speed Printer	6-11
6-14	Model 80-700 Series, High Speed Printer Modifications	6-11
6-15	Programming Routine for High Speed Printer	6-12
6-16	Model No. 81-450A Punched Card Reader Specifications	6-12
6-17	Programming Routine for Punched Card Reader	6-13
6-18	Model No. 81-810A and 81-812A X-Y Plotter Specifications	6-13
6-19	CEU Second Word Bit Format for X-Y Plotter	6-14
6-20	X-Y Plotter Commands and Bit Configuration in CEU Second Word	6-14
6-21	Diagnostic Routine for X-Y Plotter	6-14
6-22	Movable Head Disc Storage, Movement Commands	6-17
6-23	Model 81-654A Fixed Head Disc Storage Specifications	6-18
6-24	Model 81-654-128A Disc Storage Capacity Specifications (One Disc)	6-18
6-25	Fixed Head Disc TEU Second Word Format	6-20
6-26	Fixed Head Disc CEU Second Word Format	6-20

LIST OF RELATED PUBLICATIONS

The following publications contain information not included in this manual but necessary for a complete understanding of the 810B Computer System.

Publication Title	Publication No.
Technical Manual 810B General Purpose Computer	303-095019-000
Reference Manual 810A/810B Assembler	323-095052-001
Technical Manual 810A/810B Loader Program	322-095055-00 1
Technical Manual 810/840 Computer Series Library Subroutines	322-095057-001
Technical Manual 810A/810B Diagnostic Programs	322-095061-001
Technical Manual 810A/810B FORTRAN IV Compiler	322-095062-001
Technical Manual 810A/810B Operator	302-096064-002
Reference Manual 810A/810B Operating	312-095071-000
Technical Manual 810A/810B Assembler Program	322-095094-001
Drawings Manual 810B General Purpose Computer	304-095116-000
Design Manual 810B Input/Output Interface	310-095117-000



SECTION I GENERAL DESCRIPTION

INTRODUCTION

The SEL 810B Computer shown in the frontispiece, is a fast, general-purpose, 16-bit binary computer. The low cost, speed, and highly flexible input/ output structure of this computer make it especially well suited to real-time data collection, processing, and control applications. The 810B computers are designed to meet field requirements such as the following:

Industrial process control (Direct Digital Control)

Factory test automation

Missile and aircraft test data collection

Data logging and display

Real-time and post-test data processing

Telemetry data processing and simulation

Flight simulation

In addition to the basic computer, the SEL 810B system consists of a large variety of standard peripheral devices, data acquisition and display subsystems, and a comprehensive software package.

GENERAL CHARACTERISTICS

810B COMPUTER

All silicon monolithic integrated logic circuits

Sixteen-bit word length plus parity

8192-word memory

750-nanosecond full cycle time

Memory parity bit with parity generator/ checker

Fully parallel operation

Computation time including access and indexing:

Add, Subtract	1.5 microseconds
Multiply	4.5 microseconds
Divide	8.25 microseconds

Double-length Accumulator

Hardware index register (lower B-Accumulator)

I/O structure capable of handling 64 peripheral device controllers (drivers and terminators for 16 controllers supplied with the basic computer)

Two separate levels of priority interrupt

Sixteen sense switches

Switch-addressable program halt

Power fail safe

ASR-33 typewriter with paper tape reader and punch mounted on stand beside the computer

Computer size - 24 inches wide, 62 inches high, 30 inches deep (45 inches deep including optional desk top)

Typewriter size - 22 inches wide, 35 inches high, 18 inches deep

Temperature Environment, Operating:

810B Computer (excluding Teletypewriter) -0° to 55° C (32° to 131° F)

Teletypewriter - 10° to 35° C (50° to 95° F)

COMPUTER OPTIONS

Up to eight automatic block transfer control units capable of transferring up to 1, 333, 000 words per second

Additional hardware index register

Memory expandable to 32K

Program protect and instruction trap feature for guarding blocks of memory against modification and for preventing execution of privileged instructions.

Up to 98 individual levels of priority interrupts

Variable base register-increases direct addressing capability

ASR-35 console typewriter in place of ASR-33

I/O parity checker and generator

Real-time clock

Computer graphics processor

Stall alarm

Auto start

STANDARD SOFTWARE

Full ASA FORTRAN compiler - operates in 8K memory

FORTRAN library

Assembler - relocatable object format, Macro capability, and extensive set of pseudooperations

Compiler/assembler loader

Utility routines - debugging aids, I/O handlers, tape editor

Maintenance routines - complete set for computer and peripheral units

PERIPHERAL DEVICES

Card reader - 200 and 400 card/minute

Card punch - 100 cards/minute

Paper tape reader (photoelectric) - 300 characters/second

Paper tape punch - 110 characters/second

Magnetic tape control unit - handles up to eight tape units

Magnetic tape units - 45, 75, 120, 150 inches/ second; 200, 556, 800 characters/inch; 7 and 9 track

Movable head disc file - 1.5 million words storage, 150 ms maximum track access time (track 00 to track 99)

Fixed head disc files - up to 909K 16-bit word storage, 8.3 ms average access time

Typewriters - ASR-33, KSR-33, ASR-35, KSR-35, RO-33, RO-35, 10 characters/second Line printers - 300, 600, 1000 lines/minute, 120 columns/line

Incremental plotters - 12 inch-chart width (300 steps/second) and 31-inch chart width (200 steps/second)

CRT display - 10 x 10-inch display area including vector generator with the following options:

> Alphanumeric character generator Function switches Light pen Line texture control

Interval timer

Interface subsystem components

Multiplexer - low-level and high-level, solidstate and relay switching

Sample and hold units

Analog/digital converter - up to 15 bits binary. Word rates to 50K words/second

Digital/analog converter - up to 12 bits binary

Customer interfaces

APPLICATIONS PROGRAMMING

The Systems Engineering Laboratories Programming Group has developed both total and basic sets of applications programs for many 810B systems. Capability and experience exists in the areas of:

Real-time executives - monitor systems

Data collection, corrections, recording and logging

Industrial process control

Time-shared operations

Data display

Data analysis and scientific computation

COMPUTER ORGANIZATION

The SEL 810B Computer is formed by four major units: memory, control, arithmetic and input/ output (see figure 1-1). The memory unit stores the instruction words which define the operation of the computer and the data words on which the computer operates. The control unit calls up the instruction words, decodes them and issues commands to operate the computer. The arithmetic unit performs computation with data words supplied by the input/output unit and the memory unit under the direction of the control unit. The input/output unit transmits data words, commands, and status reports between the computer and peripheral equipment. The computer operates on, and from, 16bit binary words which are transferred in parallel between the computer units. Arithmetic operations are performed using two's complement binary arithmetic with negative words stored in the two's complement form. The combined control and arithmetic units are often called the mainframe section.

MEMORY UNIT

The memory unit is composed of one, two, three or four separate modules. Each module has 8192 addressable storage locations. Each location consists of one 16-bit data or instruction word plus a parity bit. The total number of storage locations can range from 8192 provided by the basic 8K module to 32, 768 available with four 8K modules.

Individual modules are composed of these four elements:

- a. 8K x 17-bit Magnetic Core Memory
- b. 13-bit Memory Address Register

- c. 17-bit Data Register
- d. Self-contained Timing and Control

Instruction words and data words are loaded into specific addresses prior to the program execution. Loading may be performed manually through the panel controls or automatically from peripheral units through the use of the supplied loader program. Each input word is transferred to the memory data register and the accompanying storage address is transferred to the memory address register. When both registers have been loaded, a "write" command is issued by the program control unit and the 17 bits in the memory data register are written into the 17 magnetic cores addressed by the memory address register.

When the entire group of instruction words forming a program is loaded and execution is started, addresses selected by the control unit are sent to the memory address register and a "read" command is issued. The state of each core at that address is sensed and transferred to the memory data register. The sensing of the cores sets them all to the same state, so the memory word now in the memory data register is immediately rewritten into its original memory location so as to be available for later use. The word is also transferred to the control unit to be decoded or to

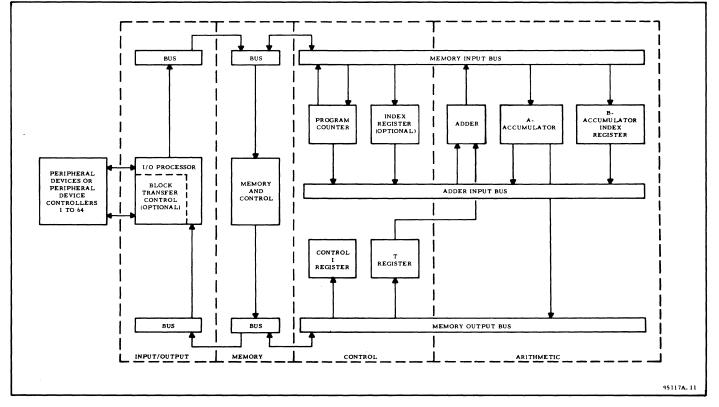


Figure 1-1. SEL 810B Block Diagram

the arithmetic unit for computation. The memory read and write cycles are completely automatic so that only the memory address and source or destination must be supplied by the program.

CONTROL UNIT

The control unit contains a 15-bit binary PROGRAM COUNTER capable of directly addressing 32, 768 memory locations. This counter supplies the addresses of the instruction words from which the computer operates. The counter is initially set to the address minus one of the first instruction of a program when the computer is started. It is then automatically advanced by each instruction until a Halt, Branch or Conditional Skip instruction is read from memory. The Halt instruction stops the computer while the Branch instructions change the contents of the program counter to the operand address contained in the instruction. The Skip instructions cause the program counter to be advanced by either one or two locations, depending on the value of the Skip condition specified by the instructions.

The instruction words are read from memory into the INSTRUCTION REGISTER and automatically restored in memory. The binary digits forming the instruction word are then applied to the OPERA-TION CONTROL circuits. The unique codes assigned to each instruction are then decoded and used to provide timing and gating signals to the remainder of the machine. The signals from switches on the CONTROL CONSOLE are also connected to the OPERATION CONTROL circuits. External PRIORITY INTERRUPTS will cause the control circuits to switch the program counter to programs designed to process the external demand.

The memory cycle during which instruction words are read and decoded is referred to as the "Instruction Cycle". Some instructions, called memory reference instructions, contain a memory address which specifies the location of an "operand" which is to be operated on by the computer. For these instructions, one or more additional memory cycles, called "Execution Cycles", are required. During the instruction cycle, the memory address is supplied in part by the "operand address" contained in the instruction word and by the program counter. The operand is read from memory and operated upon according to signals provided by the operation code. Most memory reference instructions are accessed and executed in a total of two cycles. However, instructions such as multiply and divide require more than one execution cycle.

Many instruction words require no operand from memory and are executed completely within the instruction cycle. Others, while requiring no operand from memory, do require one or more execution cycles for completion. Chief among this latter group are the shift instructions. For these instructions, a group of bits within the instruction word defines the number of shifts to be performed while the operation code of the word defines the type of shifting to be done. Other instructions, notably the input/output control instructions, are composed of two instruction words; one defining the type of operation and the unit and the other defining the actual operand or the operand memory location. The words forming these input/output instructions are automatically unloaded from memory in the proper sequence.

ARITHMETIC UNIT

The arithmetic unit consists of a 16-bit adder and several accessory storage registers. Two of these registers, the A-ACCUMULATOR and the B-ACCUMULATOR, may be loaded and unloaded by program control. The A-ACCUMULATOR is the primary arithmetic register and derives its name from its function of accumulating results of the arithmetic operations. Because only one word may be taken from the memory and input/output units by each instruction, the second operand in add and subtract operations must be loaded in a register prior to the add and subtract instructions. The A-ACCUMULATOR fulfills this function and also provides temporary storage for the result of the arithmetic operation. The B-ACCUMULATOR holds the multiplier during multiply operations and stores the least significant bits of the product. in addition to these strictly arithmetic functions. the two accumulators provide a convenient storage area for rearranging data words through shifting and logical operations.

A third register connected to the adder is the T-REGISTER which holds the operand unloaded from the memory. This 16-bit register plus the 16-bit A and B-ACCUMULATORS supply inputs to the 16bit binary ADDER. When an add instruction is performed, the data words are simply added according to the rules of two's complement binary arithmetic.

The basic data format of the 810B computer is a 16-bit binary single-precision fixed point word. (See figure 1-2). This format contains the sign bit in bit position 0, with bit position 1 holding the most significant data bit and bit position 15 holding the least significant bit. Two's complement representation is used for negative numbers. This format is defined as an integer with an imaginary binary point located to the right of bit position 15. The 810B set of library integer subroutines assumes this representation. The programmer can, of course, scale single-precision words in any desired manner and utilize the extensive shift and test instruction repertoire to maintain the binary point location.

The 810B Computer also accommodates doubleprecision data words (figure 1-2B) of 30 bits plus a sign through the use of the extended B-Accumulator. Each double-precision data word is normally stored in two adjacent memory locations with the most significant half stored in the first (lower) address. The product generated by a singleprecision multiply is located in the A and B-Accumulators in this format. The dividend is assumed to be in this double-precision format prior to the execution of the DIVIDE instruction.

Three floating point data formats are utilized by the 810B Computer library. The single-precision floating point format consists of two words (figure 1-2C). The first word contains the sign and 15 most significant bits of the fractional mantissa; the second word contains the six least significant mantissa bits and the signed 8-bit exponent. The words are stored in adjacent memory locations with the first word located in the lower memory address. Both the mantissa and the exponents carry separate signs so that the mantissa can be positive or negative independent of the sign of the exponent. Two's complement representation is used for negative numbers.

Double-precision floating point format consisting of three memory words is provided for use with the set of double-precision floating point library subroutines (figure 1-2D).

The third floating point data format (complex floating point data) is provided for the set of FORTRAN IV subroutines dealing with complex numbers (figure 1-2E).

The arithmetic unit includes two single-bit registers which are addressable by the program. The first of these is the OVERFLOW latch which can be set during addition, subtraction and division operations. The overflow for an add or subtract occurs when the result exceeds the accumulator capacity. A divide overflow occurs if the divisor is equal to or smaller than the dividend. This latter overflow is due to the fact that the machine treats all divide arguments as double-precision numbers by scaling the single-precision divisor by 2^{15} . If the dividend is larger than the scaled divisor, the quotient will necessarily be a number greater than 2^{15} . Such a number exceeds the capacity of the 15-bit A-Accumulator in which the quotient is to be stored; this produces a false divide. The overflow latch lights the OVERFLOW indicator on the control console and remains set until tests, and reset, by an

SOF (skip no overflow) instruction. Because the latch remains set until tested, such a test should be made immediately following an arithmetic process when an overflow condition could result. This prevents the possibility of a second overflow being undetected by the already set latch. The overflow latch can also be set with an OVS instruction.

The second addressable arithmetic latch is the CARRY latch which connects to the least significant bit of the parallel adder. This latch is set in the regular arithmetic processes to produce a two's complement number (one's complement of the number plus one). The latch is used in the addition and subtraction of double-precision numbers formed in the A and B-Accumulators. The least significant words of the double-precision numbers are processed and stored in the B-Accumulator. If a carry or borrow is generated, it will cause the sign of the B-Accumulator to change. A CSB (copy sign of B) instruction is used to set the carry latch to the state of the B-Accumulator sign bit and then reset the B sign bit to zero (as required in the doubleprecision format). If the operation is addition, the True output of the carry latch is added together with the most significant word; if a subtract operation is in process the False output of the carry latch is added to the most significant word (effectively subtracting the borrow).

The CSB instruction should be followed immediately by the AMA or SMA instruction which operates on the most significant half of the double-precision operand, since the carry latch is cleared at the end of the execution of all instructions except CSB.

INPUT/OUTPUT UNIT

The basic input/output unit contains an input/output processor that communicates with peripheral device control units over 64 parallel direct information channels. Each device control unit is assigned a unit number that corresponds to the number of the direct information channel that is used as a communication path between the computer and the device control unit. Each device control unit can control or communicate with several peripheral devices; therefore, the number of individual peripheral devices that can communicate with the computer, or be under the control of the computer, is virtually unlimited.

Data transfer instructions are provided that enable word transfers directly between the computer memory (or the A-Accumulator) and the peripheral device, through the device control unit. In addition, external unit commands and test instructions are provided.

The I/O instruction set is particularily powerful because each instruction causes several functions

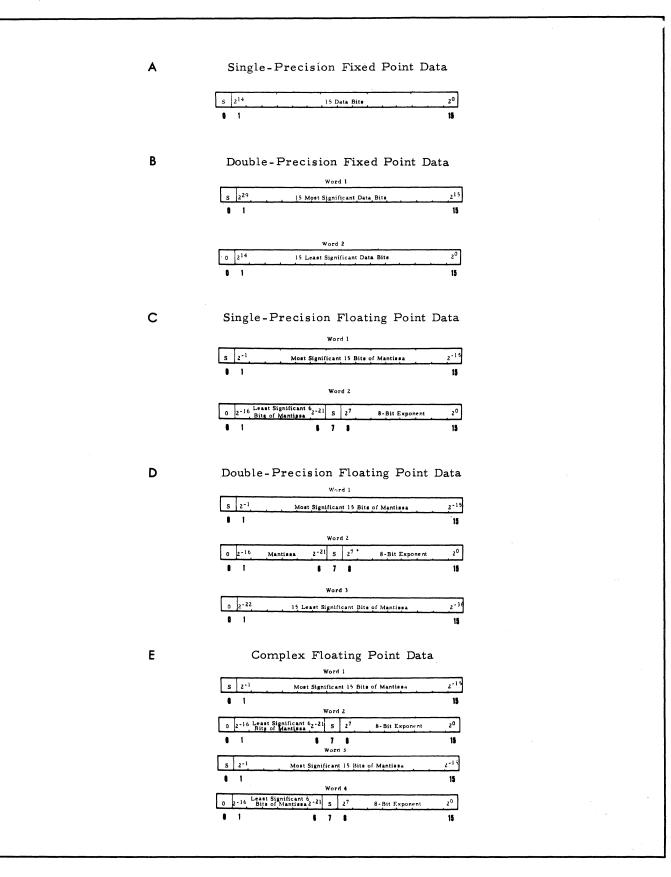


Figure 1-2. 810B Computer Basic Data Formats

to be performed. First, execution of each I/O instruction causes a device to be connected to the computer. The device (unit) number (direct information channel number) is contained in each I/O instruction.

Second, an automatic test is made of the device which determines if the device can execute the instruction. Third, the data or command transfer is made if the device is "ready". Fourth, the device is disconnected. If the device is not ready when tested, the computer will either wait until the device is ready and then transfer, or it will disconnect the device and advance the program counter to a "reject" location. A "Wait Flag" is provided in each I/O instruction, except the test instruction (TEU), to enable the programmer to specify the "Wait" or "Skip" mode of execution. The normal time required to perform the complete connect, test, transfer, and disconnect operation is only four machines cycles plus wait.

In addition to the basic I/O structure, up to eight fully buffered, block transfer units can be added to the computer. These units permit transfer of blocks of data up to 32, 768 words in length between the computer and peripheral devices. Block transfer is made under hardware control at rates up to 1, 333,000 words per second. A single cycle is stolen per word transfer. An automatic reinitialization feature is provided which enables chaining of block transfers. Also up to six Computer Graphics Processors (CGP) may be added to the SEL 810B Computer. The CGP is similar to the BTC with the exception of its specialized operating characteristics and added control functions. Unlike the BTC, the CGP examines each word from memory and either interprets the word as data or as an instruction.

A priority interrupt system is provided which enables the computer to have up to 98 individual levels of priority interrupt. Programmable interrupts can be selectively enabled and disabled under program control. A unique memory location is assigned to each level.

An ASR-33 typewriter, paper tape punch and reader are supplied with the basic computer. The reader operates at 20 characters per second and the punch and printer operates at 10 characters per second. The ASR-33 can be operated either on-line of offline. When operating on-line, the input and output unit operate independently, which enables, for example, a paper tape to be read and a separate set of characters to be printed at the same time. Other console typewriters, such as the ASR-35, can be supplied in place of the ASR-33.

SEL 810B SOFTWARE SYSTEM

A comprehensive, fully-integrated, well-documented and completely checked-out program preparation, library, debugging and utility system is supplied with the SEL 810B Computer system.

Specific features of this standard package are described in detail but a briefing in regard to the philosophy behind the software system package design is mentioned in the following paragraphs.

In determining the optimum software package for the type of equipment under consideration, the following factors were deemed to be of prime importance:

a. The large variety of equipment configuration which will be delivered.

b. The type of application which will be programmed for the equipment.

c. The large amount of programming personnel time which will be involved in developing and debugging operational programs.

d. The need to utilize programs and routines which may already exist on other equipment.

e. The quality and completeness of the documentation supplied with the software and library routines.

In order to satisfy these objectives, two basic types of program preparation systems are provided; a symbolic Macro assembler and a full FORTRAN IV compiling system. Depending upon the specific requirements of a specific portion of an operational package, these two programming systems provide the user with an optimum capability where tradeoffs between coding and checkout time and program running time are involved.

The fact that a specific portion of a program can be coded in either language is most significant to the user. The loader will accept both FORTRAN and assembler generated coding in any sequence.

This feature together with the very comprehensive debug package will significantly reduce the coding and checkout time required to produce operational programs.

The FORTRAN IV language specified for this system is the standard ASA FORTRAN IV language; thus, the FORTRAN IV supplied will provide direct compatibility with the majority of other manufacturer supplied FORTRAN IV systems. In order to satisfy the requirements that all of the supplied software system will operate on a wide variety of computer configurations, especially in the area of peripheral equipment, all of the supplied packages are written in a modular form with a standard program interface specification.

SEL 810B ASSEMBLY PROGRAM

All computer instructions are accepted by the assembler and addresses can be expressed in symbolic, decimal, or octal formats, including address arithmetic with combinations of these.

The following special pseudo-ops are also processed.

BSS	Reserve block of storage name at start
BES	Reserve block of storage name at end
EQU	Define symbolic name
ORG	Set next storage address
ZZZ	Set instruction bits to zero
REL	Set assembly mode to relative
ABS	Set assembly mode to absolute
CALL	Call library subroutine
NAME	Define subroutine name
DATA	Define octal, decimal (fixed or floating) or alphanumeric data
MOR	Pause in assembly process
END	End of program
FORM	Sets bit assignment for ''FDAT'' pseudo-op
FDAT	Same as "Data" with bits assigned by "Form
DAC	Used to generate direct address constant
EAC	Used to generate extended address constants
NOLS	Used to stop program listing
LIST	Used to continue program listing
MACR	Used to name a Macro

EMAC

Used to terminate a Macro.

A symbolic side-by-side listing complete with error messages is output (operator option) along with the object output tape.

SEL 810B LOADER

The SEL 810B object program loader is designed to be compatible with the FORTRAN IV Compiler and the Assembly program.

The program provides for relocatable and absolute instructions. The capability of using pre-compiled subroutine libraries is included in a manner which allows that a given routine will only be loaded once, regardless of the number of times it is referenced in the program.

The system has been designed with the joint aims of (a) minimizing indirect addressing for those program elements which will operate most frequently; (b) establishing uniform subroutine construction and linkage; (c) relieving the user from over-concern with any complexities introduced by the MAP addressing scheme.

SEL 810B FORTRAN IV

Ease of use was a prime consideration in the design of this compiler. As a result, programmers are free of the restrictions often found in other systems. Convenience features include:

a. One-pass Operation - From source language to machine language object code is a standard feature.

b. No Reserved Identifiers - All names are available for use as identifiers.

Optional Tracing - This feature allows с. selective object code tracing for diagnostic purposes.

Optional Mapping - This feature provides d. a listing of the subprograms required for execution and the names or values and relative location assignments of all variable-array names and constant values used by the program.

e. Optional Chaining - This feature provides for sequential loading and execution of segmented programs.

SEL 810B DEBUG

The debug program is a utility program designed to help a programmer debug a program while it is in memory. The following functions are provided:

a. Type the contents of specified memory in octal or command format.

b. Modify the specified memory; input being in octal format.

c. Dump specified memory areas onto paper tape in a format (non-relocatable) that can be loaded using the loader resident in Debug.

d. Enter breakpoints in order to "leap-frog" trace a program.

e. Clear specified areas of memory to zero.

f. Search memory for references to specified areas.

g. Initiate branches (or Halt and Branch) to any part of memory.

h. Load a binary tape that was dumped using Debug.

Each of these functions are initiated by typing a keyword through the console typewriter keyboard.

SEL 810B UPDATE

Correction of errors in card decks is a relatively easy procedure, consisting of pulling out the bad cards and inserting new cards. However, symbolic source programs on paper tapes or magnetic tapes are not so easily corrected or modified.

The UPDATE program is designed to allow the computer operator to easily correct or modify a symbolic source program tape by providing the following functions:

a. Deletion of a specified line or group of lines.

b. Insertion of a new or replacement line or lines.

All references to the symbolic source tape are made by referring to a sequence number. This number is present on all assembly listings.

SEL 810B LIBRARY PACKAGE

The SEL 810B library package includes the complete set of ASA FORTRAN subroutines in the following categories:

Single-Precision Floating Point Functions

Double-Precision Floating Point Functions

Complex Floating Point Functions

Integer Functions

Input/Output Functions

Control Functions

SEL 810B MAINTENANCE ROUTINES

The SEL 810B Checkout Program is a complete package designed to give the operator the ability to exercise the memory, the mainframe logic, the input/output channels and associated peripheral equipment.

The memory exerciser routine generates various types of worst case bit patterns and exercises the memory with these patterns while monitoring for errors. Provisions are made for automatic relocating of the exerciser program to allow the entire memory to be included in all tests. Also included are certain branch/skip instructions which are sequenced and executed through each location in the memory.

The mainframe exerciser routine executes the entire instruction repertoire individually in a large variety of sequence while monitoring the results for errors. Errors are indicated by halts. Pertinent information concerning the instruction that failed and the nature of the failure can be obtained from the A and B-Accumulator displays, the program counter and certain selected memory locations.

The programs for the I/O channels and associated peripheral equipment test the ability of the various I/O units to generate or receive all acceptable characters. A selected input is used and visual monitoring of the control panel or output unit is required by the operator for verification of proper operation. Equipment tested includes standard Teletypewriter output, input, punch and reader as well as optional card punch, card reader, line printer, high-speed paper tape equipment, magnetic tape units and other units as needed for a particular application.

POWER FAIL SAFE

The power fail safe feature provides an "override" interrupt to allow program storage of the contents of all data registers in the event that power drops below 80 volts. This standard feature assures that no information will be destroyed when power is disrupted. The program can be conveniently resumed after power is restored either manually or automatically by means of the optional Auto Start feature.

SECTION II MACHINE LANGUAGE PROGRAMMING

INTRODUCTION

The 810B Computer is operated by a series of instruction words stored in the magnetic core memory. The instruction words are successively read from memory locations addressed by the program counter. Each word specifies one operation; transferring a data word from an input unit to a memory location, adding a memory word to the word in the A-Accumulator, shifting the contents of the A-Accumulator, etc. The program counter is normally advanced one count after each instruction to access the instruction word located in the next sequential memory address. The program counter may be preset to any count by Branch/Skip instructions, which detect certain conditions such as A-Accumulator sign positive, overflow condition, input word ready, etc. The program counter then continues its sequential advance, but starts from the new address until again preset. The branch may be to either a higher or lower count so that portions of a program may be repeated until the branch condition is no longer present.

A list of instructions is provided for the 810B Computer that includes Load/Store instructions which transfer words between the memory and the accumulators, Arithmetic instructions, Shift instructions which allow moving of the bits within words, Logical instructions (AND, OR, NEGATE, etc.), Control instructions (HALT, etc.), Branch/ Skip instructions to provide program modification and Input/Output instructions to command peripheral devices and transfer data into and out of the computer.

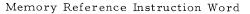
Each instruction word is formed by 16 bits, each of which performs a particular function; defining the operation to be performed, addressing a memory location, defining the number of shifts, etc. The function of a particular bit will vary in different types of instructions. For example, in some words, bit 14 forms part of a memory address; in others, bit 14 forms part of the operation code. The function of the bits depends on the instruction word type defined by the four-bit operation code located in bits 0 - 3 of the first word of each instruction.

There are two types of instruction words used by the SEL 810B; those containing memory addresses within the instruction word and those containing additional code bits in lieu of the address bits.

MEMORY REFERENCE INSTRUCTIONS

The memory reference instructions access the magnetic core memory for an operand. These words contain a four-bit binary operation code, a nine-bit partial memory address and three address modifiers.

Operation Code	1	x	1	м		9 Bit Operand Address
0	3	4	5	6	1	15



The four-bit binary operation codes for the memory reference instructions vary from 0001 (01₈) to 1110 (16₈), not including 1011 (13₈). Codes 00_8^{8} , 13₈ and 17₈ are reserved for augmented instructions (described in later paragraphs). The 13-memory reference instructions contain a nine-bit operand address field (m) that may be coded to obtain 512 unique locations. The memory, whether it is formed by a single memory module or by several modules, is divided into 512-address memory address partitions (MAPs) for addressing purposes. Each MAP extends from memory address XX000 to XX777₈, where XX consists of the six most significant address bits defining the MAP address. In a memory with 8,192 addresses, there are 16 MAPs beginning with MAP 00_{o} and extending through MAP 178. In a maximum memory of 32, 768 addresses, the MAP designations range from 00_{o} to 77_{g} with the addresses arranged in the following sequence:

MAP	008	000008 -	. 00777 ₈
MAP	018	010008 -	. 01777 ₈
MAP	028	020008 -	02777
•		•	•
•		•	•
•			
MAP	778	770008 -	. 77777 ₈

The state of the MAP designator bit (shown as M in the word format diagram) determines the MAP that will contain the operand address. If the MAP designator bit is a ZERO, the operand address will be in MAP 00₈. If the MAP designator bit is a ONE, the operand address will be in the MAP containing the instruction word currently being executed. The MAP address of the instruction is supplied by the program counter which advances sequentially across the imaginary MAP boundaries. The program count ranges from 00000_8 in MAP 00_8 to 777778 in MAP 778 thus including all possible MAP designations. The upper six bits (two octal digits) of the program counter can therefore, add the necessary MAP designation to the nine-bit (three octal digits) operand address to provide a complete 15-bit memory address. The addressing of MAP 00_8 requires only that zeros be put in the upper six bit positions of the 15-bit address to produce addresses 00000_8 to 00777_8 from the basic nine-bit operand address.

The significance of the MAP 00_8 address lies in the fact that these addresses are directly addressable by all instructions irrespective of the MAP location of those instructions. This allows the storage of constants, input/output locations, subroutines, etc., to be stored in this common address area where they can be directly accessed by any portion of the program.

The index flag, shown as X in bit position 4 of the memory reference instruction word format diagram (figure 2-1), is set to one to cause the 15-bit concatenated MAP and operand addresses to be added to the contents of the current index register. The current index register can be either the B-Accumulator, or the optional hardware index register, depending on the condition of the index pointer. The index count can be any 15-bit binary number ranging from 000008 to 777778. The addition of this number to the concatenated address allows the addressing of any memory location within the fullsize 32,768-address memory. If, for example, the instruction being executed is in MAP 128, and the index count is 020228, the 9-bit operand address is 7248 and both the MAP designator and index flag are ones; the resulting effective address is 127248 + 020228 (X) = 147468. If the MAP designator were a zero and the index flag a one, the resulting effective address is 007248 + 020228(X) = 027468.

The B-Accumulator when used as an index register serves another important function in that the register can be incremented by one with an instruction. The optional hardware index register can be incremented by any quantity from zero to fifteen. The incrementing instructions also test the register for negative signs and generate a skip (an extra advance count) to the program counter if the register is not negative. This feature allows the programmer to load a basic negative number into the B-Accumulator or the hardware index register, append the index flag to the instruction and create an iterative subroutine that will access a series of sequential memory addresses. Such a subroutine or "loop" using the B-Accumulator as the index register and written in assembly language is shown in table 2-1. In assembly language a "l" is used to indicate an indexed instruction, an apostrophe (') indicates an octal number and either absolute (220) or symbolic (LOOP, input) addresses can be used. A complete description of the assembly language is presented in Section III.

This series of indexed instructions beginning with location LOOP serves to add 20 pairs of numbers and to store the resulting sums in 20 memory locations. The routine assumes that the index pointer has been set to the B-Accumulator. The first pair of numbers is taken from locations 200

Location	Operation	Address	Comments
INDX	LBA	=-20	Load an index count of -20 in the B-Accumulator.
LOOP	LAA	220,1	Load the A-Accumulator with data word from location (220 + index count).
	AMA	320,1	Add to the contents of the A-Accumulator to the contents of location (320 + index count).
	STA	420,1	Store the sum from the A-Accumulator in loca- tion (420 + index count).
	IBS		Increment B-Accumulator, test for index count of zero, skip next instruction if zero.
	BRU	LOOP	Take next instruction from location LOOP.
PROG	LAA	INPUT	(Next instruction after index loop.)

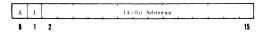
Table 2-1. Sample Listing

added to the index count by the IBS instruction. The resulting -19 count does not equal zero, so the next instruction is executed. This instruction is an unconditional branch instruction which sets the program counter to location LOOP. The next two arguments are taken from locations 201 and 301 and stored in location 401 and the cycle is repeated.

After adding 20 sets of numbers, the final IBS instruction reduces the index count to 00. This causes the next instruction (BRU to LOOP) to be skipped. The program counter now calls a new set of instructions from memory beginning with location PROG.

The third address modification flag contained in all memory reference instructions is the Indirect

Address flag (bit 5.) When present, this bit causes the address, contained in the instruction, as modified by the index and MAP bits, to be interpreted as the location in which the operand address is contained, rather than as the location of the operand itself.



Indirect Address Word

The indirect address word (shown above) contains 14 address bits which are merged with the most significant bit from the program counter. The indirect address may be in the same memory half (of 16K) as the program counter.

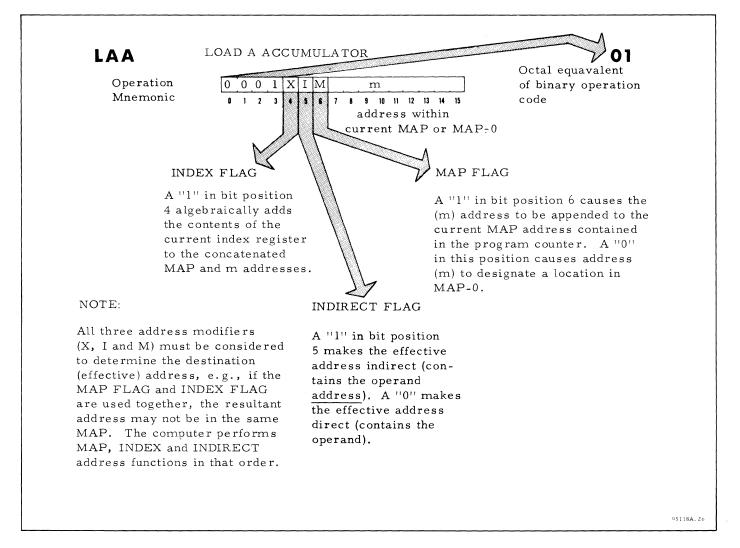


Figure 2-1. Typical Memory Reference Instruction Word Format Diagram

The indirect word format also contains an index flag which, if a one, adds the index count to the indirect address. The index count may be added to the address in the instruction word, and/or the indirect address depending on the presence or absence of an index bit in the instruction and indirect words. The indirect address also includes an indirect flag bit permitting multi-level indirect addressing.

Memory reference descriptions consist of the three-letter mnemonic and a two-character octal operation code. The permissible address modifiers are also shown. An example for a memory reference instruction (LAA) is shown on figure 2-1.

AUGMENTED INSTRUCTIONS

Augmented instructions contain no memory address bits in the first word but do contain additional (augmenting) operation code bits. The augmented instructions have operation codes of 00,13 or 00,17.

0	0_0	0	0	0			<u> </u>		AUGMENT
0		3	4	5	6	9	10	15	CODE

Augmented 00 Instruction Word

The detection of the 00 operation code in the instruction register gates the six augment code bits into a special decoding matrix.

The other augmented operation codes, the 138 and the 17g codes, are also augmented with additional code bits. These instructions have word formats that vary slightly and some include two words to complete the instruction. Two-word instructions are stored in sequential memory locations with the second word called automatically by the machine. If the indirect flag in the first word is a zero, the second word is interpreted as the operand itself. If the indirect flag is a one, the second word is coded in the indirect address word format and is interpreted as the address of the operand. If the MAP bit is a one, the most significant bit of the program counter becomes the 15th bit of the indirect address when the indirect flag is used; if the MAP bit is a zero, the 15th bit of the indirect address is set to zero.

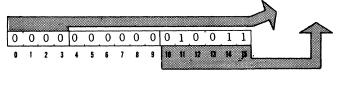
The augmented 138 operation code words are used for disabling and enabling interrupts, testing the condition of the sense switches and testing and commanding of external I/O units. The augmented 178 operation code words are used for Input/Output instructions only.

Augmented instruction octal codes consist of a two-digit operation code 00, 13, or 17 followed by a hyphen and one or two-digits showing the augmenting code. For example:

SAN

00-11

SKIP IF A - ACCUMULATOR IS NEGATIVE

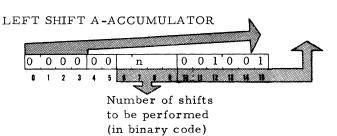


NOTE

All Augmented 00 instructions contain the augment code in bit positions 10-15.

Shift instruction words use bit position 6 through 9 to hold the number of shifts to be performed by the instruction. For example:

LSA -1.



Input/Output Instructions contain both 13g and 17g operation codes. The augmenting code bits for these instructions appear as shown in figure 2-2. Bits 10 through 15 always contain the peripheral unit number in binary code.

In the <u>IMMEDIATE MODE</u>, the second instruction word is treated as the operand. In executing MOP, CEU and TEU instructions, the contents of the instruction's second word are transferred to the specified unit. MIP execution consists of transferring a word or character from a specified unit into the instruction's second word location.

In the <u>ADDRESS MODE</u>, the instruction's second word is interpreted as the operand address. The indirect address format is used in the instruction's second word. Therefore, indexing and indirect chaining may be used in addressing the operand.

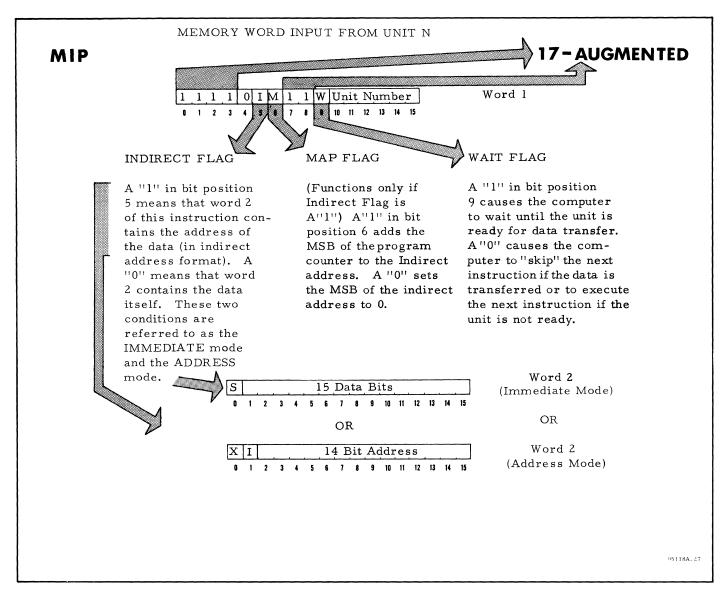


Figure 2-2. Input/Output Instruction Word Format Diagram

The addressing mode is specified in the instructions first word by the value of the Indirect Address Flag (I). If I is a ONE, the Address Mode is executed. In this mode the most significant program counter bit is appended to the most significant end of the 14-bit indirect address if the MAP Flag is a ONE, and ZERO is appended to the most significant end of the indirect address if the MAP Flag is a ZERO.

MACHINE LANGUAGE INSTRUCTION SET

The instruction words causing the various SEL 810B machine operations are described in detail on the following pages. The descriptions include the operation-mnemonic and octal machine code in bold type. The binary word format shows bit assignments for operation code, augment code, operand address and flags (MAP, INDEX, INDI-RECT, WAIT, etc.). A brief explanation of the functions, register(s) affected, memory cycles required, indicators (if any) and special notes complete the description.

ARITHMETIC INSTRUCTIONS

All arithmetic functions of the computer are performed by this group of seven instructions. The AMA (add) instruction calls a word from memory and adds it to the word previously loaded into the A-Accumulator. The memory word called in the SMA (subtract) instruction is two's complemented and added to the A, Accumulator word. The MPY (multiply) instruction repeatedly adds the memory and the A-Accumulator words according to the value of the word in the B-Accumulator. The DIV

(divide) instruction repeatedly subtracts (adds the complement of) the memory word from the doublelength word in the A- and B-Accumulators. The results of these operations are stored in the A-Accumulator (sum, difference, quotient and most significant half of the product) and the B-Accumulator (remainder and least significant half of the product). The augmented 00g RNA (Round A-Accumulator) instruction is used to round-off the most significant half of the product in the A-Accumulator according to the value of the least significant half of the product in the B-Accumulator. The AMB (add) instruction calls a word from memory and adds it to the word previously loaded into the B-Accumulator. The OVS (set overflow) instruction causes the overflow latch to be set.

AMA 05

ADD MEMORY TO A-ACCUMULATOR

0 1	0	1	Х	I	M	m	
0		3	4	5	6	7	15

The contents of the effective memory address (addend) are algebraically added to the contents of the A-Accumulator (augend). The sum replaces the previous contents of the A-Accumulator with the sign of the A-Accumulator set to the algebraic sign of the sum.

NOTE

The augment must be located in the A-Accumulator prior to the AMA instruction. This may be accomplished through a preceding LAA instruction or the augend may already be properly located as a result of a prior operation.

Timing:	2 cycles
Indicators:	OVERFLOW if the sum
	exceeds 15 bits plus sign
Registers Affected:	A-Accumulator

AMB 16

ADD MEMORY TO B-ACCUMULATOR

1 1	1	0	Х	Ι	М	m
0		3	4	5	6	7 15

The contents of the effective memory address (addend) are algebraically added to the contents of the B-Accumulator (augend). The sum replaces the previous contents of the B-Accumulator with the sign of the B-Accumulator set to the algebraic sign of the sum.

NOTE

The augend must be located in the B-Accumulator prior to the AMB instruction. This may be accomplished through preceding LBA instruction or the augend may already be properly located as a result of a prior operation.

Timing: Indicators:

Registers Affected:

2 cycles OVERFLOW if the sum exceeds 15-bits plus sign B-Accumulator

SMA 06

SUBTRACT MEMORY FROM A -ACCUMULATOR

0	1	1	0	Х	I	М		m	
0			3	4	5	6	1		15

The contents of the effective memory address (subtrahend) are algebraically subtracted from the contents of the A-Accumulator (minuend). The difference replaces the previous contents of the A-Accumulator and the sign of the A-Accumulator is set to the sign of the algebraic difference.

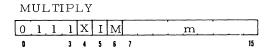
NOTE

The minuend must be located in the A-Accumulator prior to the SMA instruction. This may already be properly located as a result of a prior operation.

Timing:	2 cycles
Indicators:	OVERFLO
	difference
	plus sign
Registers Affected:	A-Accumu

Cycles OVERFLOW if the algebraic ifference exceeds 15-bits lus sign -Accumulator

MPY 07



The contents of the effective memory address (multiplicand) are multiplied by the contents of the B-Accumulator (multiplier). The most significant half of the product replaces the previous contents of the A-Accumulator. The least significant half of the product replaces the previous contents of the B-Accumulator. The sign of the A-Accumulator is determined by the algebraic sign of the product; the sign of the B-Accumulator is set to plus. The contents of the memory are unchanged.

NOTE

If the multiplier and the multiplicand are considered to be integers (binary point to the right of bit 15), the product is a doubleprecision integer (binary point to the right of bit 15 in the B-Accumulator). If the multiplier is scaled left by 2^a and the multiplicand is scaled left by 2^b, then the product is scaled left by 2 a+b.

Timing:	6 cycles
Indicators:	OVERFLOW if both
	multiplier and multi-
	plicand are equal to
	minus full scale.
Registers Affected:	A-Accumulator,
	B-Accumulator

DIV 10

DIVIDE

1	Σ,	0	0	Х	Ι	M	m	
0			3	4	5	6	7	15

The contents of the A- and B-Accumulators (double length dividend) are divided by the contents of the effective memory address (single length divisor). The quotient is stored in the A-Accumulator and the remainder is stored in the B-Accumulator. The sign of the quotient is set to the algebraic sum of the divisor and dividend signs. The sign of the remainder is set to the sign of the original dividend. The contents of the memory are unchanged.

NOTE

The dividend is assumed to be a double-precision quantity (30 bits and sign) which is to be divided by a single-precision quantity (15-bits and sign). The result is two singleprecision quantities, the quotient and the remainder. If the part of the dividend contained in the A-Accumulator is greater than or equal to the divisor, an "overflow" will result. The quotient of two single-precision quantities is obtained by the instructions shown below.

CLA		Clear A-Accumulator
LBA	DVND	Load Dividend into
		B-Accumulator
DIV	DVSR	

NOTE (Cont'd)

After the divide instruction is executed, the quotient and remainder are stored in the A- and B-Accumulators, respectively. No "overflow" can occur except when the divisor is equal to zero.

Divide scaling is performed by the simple algorithm given below. Considering the binary point of the operands to be located between the sign bit and the most significant bit, that is at B_0 in single-precision and C_0 in double-precision quantities, the binary point in the quotient is determined by the relationship:

C(M) dividend - B(N) divisor = B(M-N) quotient

where - $30 \le M \le 30$ and - $15 \le N \le 15$

If the scale factor is greater than B_{15} an overflow will result.

Timing: Indicators: 11 cycles OVERFLOW if the divisor is the portion of the dividend contained in the A-Accumulator A-Accumulator, B-Accumulator

Registers Affected:

RNA 00-01

ROUND A-ACCUMULATOR

0	0	0	0	0	0_0	<u></u> 0	0	0	0	0_0	0	0 1	
0			3	4				9	10			15	

The contents of the A-Accumulator are increased by one if the second most significant bit of the B-Accumulator (B_1) is a one.

Timing:	l cycle
Indicators:	OVERFLOW if the result
	in the A-Accumulator
	exceeds 15-bits.
Registers Affected:	A-Accumulator

OVS 00-37

SET OVERFLOW LATCH

0 0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
0		3	4					9	10					15

The overflow latch is set by the execution of this instruction.

NOTE

This instruction is used at the exit of interrupt routines to set the overflow latch if it was set when the interrupt occurred.

Timing:	l cycle
Indicators:	OVERFLOW is set
Registers Affected:	None

LOAD/STORE INSTRUCTIONS

This group of five standard and two optional instructions handles the transfer of data words within the computer. One pair of instructions - LAA (Load A-Accumulator) and STA (Store A-Accumulator) transfers data between the memory and the A-Accumulator. A second pair - LBA (Load B-Accumulator) and STB (Store B-Accumulator) communicates between memory and the B-Accumulator. All four words are memory address instructions and, as such, contain MAP, index and indirect address modifiers. The instructions are used primarily to transfer data to the accumulators for use in arithmetic operations and then to store the results of those operations. The B-Accumulator, however, also functions as a hardware index register/counter so that the LBA and STB instructions serve to load and store the index count.

The LCS (Load Control Switches) instruction is an augmented 00g word. This instruction is used to transfer the information set into the front panel control switches by the operator to the A-Accumulator. The switches can be used to modify the program in response to external requirements by using the data brought to the accumulator by the LCS instruction to change branch destinations, etc.

The two optional load/store instructions are included as part of the hardware index register option. This pair of instructions is mnemonically labeled LIX and STX. The LIX (Load Index Register) instruction loads data from memory into the hardware index register. The STX (Store Index Register) instruction stores data from the hardware index register into memory. Both of these instructions are two word instructions, with the first word of each an augmented 00g word. If the indirect bit (bit 5) of the first word is a zero the second word is the instruction operand (Immediate Mode). If the indirect bit is a one, the second word is the address, in indirect address format, of the operand (Address Mode).

LAA 01

LOAD A -ACCUMULATOR	
---------------------	--

0	0	0	1	Х	Ι	Μ		m	
0			3	4	5	6	1		15

The contents of the effective memory address replace the previous contents of the A-Accumulator. The contents of the memory are unchanged.

NOTE

The A-Accumulator must be loaded with the augend, minuend and most significant bits of the dividend prior to add, subtract and divide instructions.

Timing:	2 cycles
Indicators:	None
Registers Affected:	A-Accumulator

LBA 02

LOAD B-ACCUMULATOR	LOAD	B-AC	CUMU	JLA	TOR
--------------------	------	------	------	-----	-----

0	0	1	0	Х	Ι	M		m
0			3	4	5	6	7	15

The contents of the effective memory address replace the previous contents of the B-Accumulator. The contents of the memory are unchanged.

NOTE

This instruction is used to load the index count when the B-Accumulator is to function as the hardware index register. The B-Accumulator must also be loaded with the least significant half of a dividend and the multiplier prior to divide and multiply instructions.

Timing:	2 cycles
Indicators:	None
Registers Affected:	B-Accumulator

LIX 00-45

LOAD INDEX (OPTIONAL)

										W	or	d 1	
0	0	0	0	0	I	M	0,0	0	1,0	0	1	0	1
0			3		5	6			10				15
					(A	dd	ress	Μ	ode)	w	or	d 2	
х	1				14-1	BIT A	DDRES	s	L I _	1			
0	1	2											15

The contents of the effective memory address replace the previous contents of the index register. The contents of the memory are unchanged.

Timing:	2 cycle s (I mmediate Mode)
Indicators:	None
Registers Affected:	Index Register
Operand Address Mode:	Immediate -I=0, Address- i=1 (in Word 1)

LCS 00-31

LOAD CONTROL SWITCHES

0 0	0	0	0	0	0	0	0	0	0	1	1	0	0	1
0		3	4					9	10					15

The positions of control panel switches 0-15 are sensed as bits (up-1, center-0) and transferred to the A-Accumulator.

Timing:	l cycle
Indicators:	None
Registers Affected:	A-Accumulator

STA 03

STORE A -ACCUMULATOR

0.0	1	1	х	I	М		, m
		3	4	5	6	1	15

The contents of the A-Accumulator replace the previous contents of the effective memory address. The contents of the A-Accumulator are unchanged.

Timing:	2 cycles
Indicators:	None
Registers Affected:	None

STB 04

STORE B-ACCUMULATOR

0	1	0	0	x	I	M		m
0			3	4	5	6	7	15

The contents of the B-Accumulator replace the previous contents of the effective memory address. The contents of the B-Accumulator are unchanged.

Timing:2 cyclesIndicators:NoneRegisters Affected:None

STX 00-44

STORE INDEX (OPTIONAL)

													W	or	d 1	
0	0	0	0	0	I	М	0	0	0	1	0	0	1	0	0	

					(Address Mode)	Word 2
x	I			1	14-BIT ADDRESS	
0	1	2				15

The contents of the index register replace the contents of the effective memory address. The contents of the index register are unchanged.

Timing:	2 cycles (Immediate Mode)
Indicators:	None
Registers Affected:	None
Operand Address	Immediate -I=0, Address-
Mode:	I=1 (Word 1)

BRANCH/SKIP INSTRUCTIONS

This group of thirteen standard and two optional instructions provides the decision-making capability of the computer. Only four of these instructions the BRU (Unconditional Branch), SPB (Store Place and Branch), IMS (Increment Memory and Skip) and CMA (Compare Memory to A-Accumulator) are memory reference instructions. The remainder, all skip instructions, are augmented 00g instruction words with the single exception of the SNS instruction which is an augmented 13g code word.

The three branch instructions, BRU, LOB and SPB, specify a new address that will be transferred to the program counter to move the program to a new address in the core memory. The ten skip instructions are all dependent on the presence or absence of a specific condition such as the sign of the A-Accumulator, set overflow latch or memory word sign. If the specific condition is present (or absent depending on the instruction as presented below) the next instruction (NI), is skipped and the second successive instruction (NI+1) is executed. The NI is usually a BRU or SPB which branches the program to a new section of the memory. Thus, an SOF instruction would skip a BRU that enters a corrective subroutine, if the overflow latch were not set. If the latch were set (indicating an invalid arithmetic operation), the NI would not be skipped and the corrective subroutine would be entered.

The LOB (Long Branch) instruction consists of two memory words. The first word is an augmented 00g operation code word; the second contains a memory address. The second word is automatically read from memory as part of the normal execution cycle.

The two optional skip instructions (SXB and IXS) are included as part of the index register option. They are used to test the index register and index pointer, respectively.

BRU 11

UN	1C0	ΟN	DI	TI	ONAI	BRANCH
1	0	0	1	Х	ΙM	m

0	3	4	5	6	1	15

The effective address replaces the contents of the program counter.

NOTE

If the Program Protect and Instruction Trap option is included (and the Protect Mode switch is ON), when the BRU indirect instruction is executed following a TOI instruction to exit from a priority interrupt routine, bits 2 through 15 of the effective address replace the contents of program counter, and the Protect Latch is set to the state of bit "0" of the effective address.

Timing:	l cycle
Indicators:	None
Registers Affected:	Program Counter

SPB 12

STORE PLACE AND BRANCH

1 0	1_0	X	I	Μ	m]
0	3	4	5	6	7 15	

The contents of the program counter plus one replace the previous contents of the effective memory address and the effective memory address plus one replaces the previous contents of the program counter.

NOTE

Execution of this instruction is modified when caused by a priority interrupt in that the contents of the program counter are unchanged when transferred to the effective memory address. If the Program Protect and Instruction Trap option is included (and the Protect Mode switch is ON). when the SPB indirect instruction is caused by a priority interrupt, the status of the Protect Latch at the time of the interrupt is stored in bit 0 of the effective memory address.

Timing:	2 cycles
Indicators:	None
Registers Affected:	Program Counter

IMS 14

INCREMENT	MEMORY	AND	SKIP
-----------	--------	-----	------

1.1.0	0	X	I	М		
•	3	4	5	6	7	15

The contents of the effective memory address are increased by one. If the contents of that address then equal zero, the next instruction is skipped.

NOTE

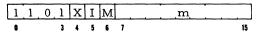
This instruction allows any memory cell to be used as an auxiliary index register.

Timing: Indicators: Registers Affected:

3 cycles None Program Counter

CMA 15

COMPARE MEMORY AND A-ACCUMULATOR (3-WAY)



The contents of the effective memory address are algebraically compared to the contents of the A-Accumulator.

If A M, the program proceeds to the next succes---sive instruction.

If A = M, the next instruction is skipped.

If A M, the next two instructions are skipped.

The contents of the memory and of the A-Accumulator are unchanged.

Timing:	3 cycles
Indicators:	None
Registers Affected:	Program Counter.

SNS 1304

SENSE	NU	JMB	ER	ΕI) SWI	[TC	СН			Binary
1 0 1	1	0_0	0	1	0.0	0	0			Switch
0	3	4	6	1	9	10	11	12	15	No.

Tests to see if a specific control panel switch (0-15) is set; if switch is NOT set, the next instruction is skipped.

Timing:	l cycle
Indicators:	None
Registers Affected:	Program Counter

IBS 00-26

INCREMENT B-ACCUMULATOR AND SKIP

0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	
0			3	4											15	

The contents of the B-Accumulator are increased by one. If the contents of the B-Accumulator are then zero or positive, the next instruction is skipped.

NOTE

This instruction can be used as part of an indexing loop in which the contents of the B-Accumulator NOTE (Cont'd)

(index register) are used to modify the operand address of an instruction, then IBS and branch back to repeat the loop. If the index number has a negative sign, the IBS instruction will eventually decrease the absolute value to zero (positive sign) and will then skip the branch instruction and proceed with the remainder of the program.

Timing:	l cycle
Indicators:	None
Registers Affected:	B-Accumulator, Program
	Counter

SAS 00-21

	SF	ΠP	Р С	N	A	-A	CC	UI	МÜ	L	ΑT	OF	ιs	IG	N	(3-WAY)
Γ	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
_					_					9						15

If the sign of the A-Accumulator is negative, the next successive instruction is executed.

If the contents of the A-Accumulator are zero, the next instruction is skipped.

If the sign of the A-Accumulator is positive and the contents are greater than zero, the next two instructions are skipped.

Timing:	l cycle
Indicators:	None
Registers Affected:	Program Counter

SAZ 00-22

	SKIP IF	A-ACCUMUL	LATOR IS	ZERO
--	---------	-----------	----------	------

0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0
0			3	4					9	10					15

If the contents of the A-Accumulator are zero, the next instruction is skipped.

Timing:	l cycle
Indicators:	None
Registers Affected:	Program Counter

SAN 00-23

SKIP IF A -ACCUMULATOR IS NEGATIVE

0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1
0			3	4					9	10					15

If the sign of the A-Accumulator is negative, the next instruction is skipped.

Timing:	l cycle
Indicators:	None
Registers Affected:	Program Counter

SAP 00-24

SKIP IF A -ACCUMULATOR IS POSITIVE

0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0
0			3	4					9	10					15

0 :	34	l de la companya de la	9	10	15

If the sign of the A-Accumulator is positive, the next instruction is skipped.

Timing:	l cycle
Indicators:	None
Registers Affected:	Program Counter

SOF 00-25

SKIP NO OVERFLOW

0.0	0.0	0	0	0	0	0	0	0	1	0	1	0	1
0	3	4					9	10					15

If the arithmetic overflow latch is set, it is reset and the next instruction is executed; if the latch is reset, the next instruction is skipped.

NOTE

This instruction is used as a program check on the magnitude of the results of arithmetic operations. The next instruction (NI), executed in the case of an overflow, is usually a BRU to a corrective subroutine. The second sequential instruction (NI+1) is the next instruction of the normal program.

Timing:	l cycle						
Indicators:	OVERFLOW is reset						
Registers Affected:	Program Counter						

SNO 00-32

SKIP NORMALIZED A -ACCUMULATOR

0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	0
0			3	4					9	10					15

If bit A_1 does not equal bit A_0 of the A-Accumulator, the next instruction is skipped.

NOTE

This instruction is used in conjunction with the left arithmetic shift instruction to normalize the contents of the A-Accumulator.

Example:

Loc.	Oper.	Address	Comments
NORM	LSA SNO	1	left shift l test for A ₀ = A ₁
	BRU	NORM	shift again if A ₀ = A ₁
	STA		store normalized word
PROG	LAA		remainder of program

Timing:	l cycle
Indicators:	None
Registers Affected:	Program Counter

LOB 00-36

LONG BRANCH

							Wo	ord l	
0.0.	0.0	0.0	.0.0	. 0	0	0.	1,1	1.1	0
0	3	4			9	10 .			15
							Wo	ord 2	2
0	.]	5-B	it Ad	dre	ss				
0 1									15

Bits 1 through 15 of the second word replace the contents of the program counter.

NOTE

This instructions allows a branch to any of the 32,768 memory locations available with the full complement of four memory modules. This instruction is NOTE (Cont'd)

extremely useful as a "return" branch from a subroutine intiated by a Store Place and Branch instruction when the stored program count is in the upper 16,384 memory addresses and the subroutine is operating in the lower 16,384 memory addresses. If the Program Protect and Instruction Trap option is included (and the Protect Mode switch is ON), when the LOB instruction is used following a TOI instruction to exit from a priority interrupt routine, the Protect Latch is set to the state of bit "0" of the effective address.

Timing:	2 cycles
Indicators:	None
Registers Affected:	Program Counter

SXB 00-50

SKIP IF INDEX POINTER IS SET TO B-ACCUMULATOR (OPTIONAL)

0_0	0,0	0	0	0	,0	0	0	1	, 0	1	0	0	0
0	3	4					9	10					15

The next instruction is skipped if the index pointer is set to the B-Accumulator.

Timing:	l cycle
Indicators:	None
Registers Affected:	Program Counter

IXS 00-N-51

INCREMENT INDEX BY N AND SKIP IF POSITIVE (OPTIONAL)

The value of N (0-15), contained in bits (6-9) is added to bits 12-15 of the index register to increase the index register contents by the positive value of N. If the contents of the index register are equal to zero or positive, after the value of N is added, the next instruction is skipped.

NOTE

The option of the IXS instruction is identical to that of the IBS NOTE (Cont'd)

instruction for the special case where N = 1. The IXS instruction contains the added flexibility of the variable N field. If an N value of zero is specified, only a test of the index register is accomplished.

Timing:	l cycle if no skip
	2 cycles if skip
Indicators:	None
Registers Affected:	Index Register, Program
	Counter

LOGICAL INSTRUCTIONS

The five logical instructions, all augmented 008 words, affect only the A- and B-Accumulators. These instructions are provided to allow the logical modification of instruction and data words.

The ABA and OBA instructions are used to mask (logically remove) portions of a single word and merge (logically combine) portions of two words. The contents of the A-Accumulator can be two's complemented through the NEG instruction, while the sign bit of the A-Accumulator can be complemented by the ASC instruction. Sign magnitude form numbers (true binary form with either + or sign to show polarity) can be converted to two's complement form and the reverse operation can be performed by use of the CNS instruction (Refer to page 2-14). The NEG, ASC, and CNS are used primarily in the arranging of data formats to satisfy input/output requirements, but also find use in creating special flag, indicator and constant words.

ABA 00-27

AND A - AND B-ACCUMULATORS

0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	1
0			3	4					9	10					15

The contents of the B-Accumulator form a logical product with the contents of the A-Accumulator. The product is stored in the A-Accumulator and the contents of the B-Accumulator are unchanged.

NOTE

This instruction is used as a masking instruction as follows:

A-Acc. (000000001111111) MASK B-Acc. (10101010101010) DATA A-Acc. (00000000101010) LOG. PROD. Timing:1 cycleIndicators:NoneRegisters Affected:A-Accumulator

OBA 00-30

OR A- AND B-ACCUMULATORS

0 0	0	0	0	0	0	0	0	0	0	1	1	0	0	0
0		3	4					9	10					15

The contents of the B-Accumulator form a logical sum with the contents of the A-Accumulator. The sum is stored in the A-Accumulator and the contents of the B-Accumulator are unchanged.

NOTE

This instruction is used as a merging instruction as follows:

A-Acc	(0000000010101010)	DATA
B-Acc	(101010100000000)	DATA
A-Acc	(1010101010101010)	LOG. SUM

Timing:	l cycle
Indicators:	None
Registers Affected:	A-Accumulator

NEG 00-02

NEGATE THE A-ACCUMULATOR

0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	
0			3	4					9	10					15	

The contents of the A-Accumulator are two's complemented.

Timing:	l cycle
Indicators:	OVERFLOW if operand is
	minus full scale
Registers Affected:	A-Accumulator

ASC 00-20

COMPLEMENT SIGN OF A -ACCUMULATOR

0_0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
0		3	4					9	10					15

The sign bit of the A-Accumulator is complemented.

Timing:	l cycle
Indicators:	None
Registers Affected:	A-Accumulator

CNS 00-34

CONVERT NUMBER SYSTEMS

0 0	0.0	0	0	0	0	0	0	0	1	1	1	0	0	
0	3	4					9	10					15	

The least significant bits of a negative-signed number in the A-Accumulator are two's complemented while the sign bit remains unchanged. Positive-signed numbers are not affected.

NOTE

This instruction is used to convert sign-magnitude numbers to two's complement numbers and two's complement numbers to sign-magnitude numbers. Positive-signed numbers are not affected because the form is the same for both number systems.

Timing:	l cycle
Indicators:	OVERFLOW if operand
	is minus full scale
Registers Affected:	A-Accumulator

REGISTERS CHANGE INSTRUCTIONS

The five standard instructions are used primarily to manipulate data, create specific formats and perform routine operations connected with doubleprecision arithmetic. These instructions are also extremely useful in rearranging data in conjunction with shift instructions to achieve necessary word formats.

There are eight optional register change instructions (TBP, TPB, TAX, TXA, TBV, TVB, XPX and XPB). The TBP and TPB instructions are included as part of the memory protect option. The TAX, TXA, XPX and XPB instructions are included as part of the hardware index register option. The TBV and TVB are included as part of the variable base register option.

CLA 00-03

CLEAR A-ACCUMULATOR

0 0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
0		3	4					9	10					15

The contents of the A-Accumulator are replaced with all zeros.

Timing:	l cycle
Indicators:	None
Registers Affected:	A-Accumulator

TBA 00-04

TRANSFER B-ACCUMULATOR TO A-ACCUMULATOR

0 0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
0		3	4					9	10					15

The contents of the A-Accumulator are replaced by the contents of the B-Accumulator. The contents of the B-Accumulator are unchanged.

Timing:	l cycle
Indicators:	None
Registers Affected:	A-Accumulator

TAB 00-05

TRANSFER A-ACCUMULATOR TO B-ACCUMULATOR

0.0	0,0	0	0,0	,0	0	0	0	0	0	1	0,	1
0	3	4				9	10					15

The contents of the B-Accumulator are replaced by the contents of the A-Accumulator. The contents of the A-Accumulator are unchanged.

Timing:

l cycle

None

Indicators:

Registers Affected: B-Accumulator

IAB 00-06

INTERCHANGE A - AND B-ACCUMULATORS

0.0.	0,0	0,	0.0.	0,0	10	0,0	1011	1,0	
0	3	4			9	10		15	

The contents of the A-Accumulator are replaced by the contents of the B-Accumulator and the contents of the B-Accumulator are replaced by the contents of the A-Accumulator.

Timing:	l cycle
Indicators:	None
Registers Affected:	A -Accumulator B-Accumulator

CSB 00-07

COPY SIGN OF B-ACCUMULATOR

0,0	0,0	0	0	0	0	0	,0	0 [.]	0	0	1	1	_ 1
0	3	4					9	10					15

The CARRY latch is set to the sign of the B-Accumulator and the B-Accumulator sign bit is then set to 0 (plus).

NOTE

This instruction is used to store the carry generated during doubleprecision addition. If the sign of the B-Accumulator is a one following execution of an AMB instruction, the execution of the CSB causes the one in B_0 to be transferred to the carry latch. An AMA, SMA or NEG instruction must be executed next to insure that the carry bit is added to the contents of A15. No instruction may be executed between the CSB and AMA, SMA or NEG because (1) the carry latch is reset at the end of the execution of all instructions except CSB and (2) the contents of the carry latch are added to the contents of A15 as part of the execution of many instructions.

Timing:	l cycle
Indicators:	None
Registers Affected:	B-Accumulator

TBP 00-40

TRANSFER B-ACCUMULATOR TO PROTECT REGISTER (OPTIONAL)

0 0	0,0	0	0,0	ļ, ļ	0,0	1	, 0	0	0	0,0
0	-3	4			9	10				15

The contents of the Program Protect register are replaced by the contents of the B-Accumulator. The contents of the B-Accumulator are unchanged.

NOTE

See "Program Protect and Instruction Trap (Model 81-080B)" in SECTION VII, OPTIONS, for program protect description.

Timing:	l cycle
Indicators:	None
Registers Affected:	Program Protect Register

TPB 00-41

TRANSFER PROTECT REGISTER TO B-ACCUMULATOR (OPTIONAL)

0 0	, 0	0	0	0	0	0	0	0	1	10	0	0	0	, 1	
0		3	4					9	10					15	

The contents of the B-Accumulator are replaced by the contents of the Program Protect register. The contents of the Program Protect register are unchanged.

NOTE

See "Program Protect and Instruction Trap (Model 81-080B)" in SECTION VII, OPTIONS, for program protect description.

Timing:	l cycle
Indicators:	None
Registers Affected:	B-Accumulator

TAX 00-52

TRANSFER A-ACCUMULATOR TO INDEX REGISTER

0,0	0	0	0	0	0	0	0	0	1	0	1	0	1	0
0		3	4					9	10					15

The contents of the index register are replaced by the contents of the A-Accumulator. The contents of the A-Accumulator are unchanged.

Timing:	l cycle
Indicators:	None
Registers Affected:	Index Register

TXA 00-53

TRANSFER INDEX REGISTER TO A-ACCUMULATOR (OPTIONAL)

0,0	0,0	0	0,0	0	0	0	1	0	1	0	1	1
0	3	4				9.	10					15

The contents of A-Accumulator are replaced by the contents of the index register. The contents of the index register are unchanged.

Timing:	lcycle
Indicators:	None
Registers Affected:	A-Accumulator

TBV 00-42

TRANSFER B-ACCUMULATOR TO VARIABLE BASE REGISTER (OPTIONAL)

0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0
0			3	4					9	10					15

This instruction transfers bits 1 through 6 of the B-Accumulator to bits 1 through 6 of the variable base register. The contents of the B-Accumulator are unchanged.

Timing:	l cycle
Indicators:	None
Registers Affected:	Variable base register

TVB 00-43

TRANSFER VARIABLE BASE REGISTER TO B-ACCUMULATOR (OPTIONAL)

0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1
			3	4					9	10					15

This instruction transfers the 6-bit contents of the variable base register to bit positions 1 through 6 of the B-Accumulator. B-Accumulator bits 0 and 7 through 15, are set to zero. The contents of the variable base register are unchanged.

Timing:	l cycle
Indicators:	None
Registers Affected:	B-Accumulator

XPX 00-46

SET INDEX POINTER TO INDEX REGISTER (OPTIONAL)

0,	0,0	0	0	0	0	0	0	0	1	0	0	1	1	0	
0		3	4					9	10					15 ·	

The index pointer flip-flop is set by the execution of this instruction. When this flip-flop is set, the presence of an index flag bit (X=1) in an instruction or indirect address word causes the contents of the index register to be added to the operand address.

Timing:	l cycle
Indicators:	Index Pointer
	Light ON
Registers Affected:	None

XPB 00 47

SET INDEX POINTER TO B-ACCUMULATOR (OF C NAL)

0,0	Ō	0	0	0	0	0	0	0	1	0	0	1	1	1
0		3	4					9	10					15

The index pointer flip-flop is reset by the execution of this instruction. When this flip flop is reset, the presence of an index flag bit (X=1) in an instruction or indirect address word causes the contents of the B-Accumulator to be added to the operand address.

NOTE

Operation of the MASTER CLEAR switch resets the index pointer flip-flop.

Timing:	l cycle							
Indicators:	Index Pointer Light OFF							
Registers Affected:	None							

SHIFT INSTRUCTIONS

The eight instructions forming the shift group are augmented 008 instructions with bits 6 through 9 containing the binary shift count. While the actual count is in binary code, the number of shifts to be performed by the instruction is usually specified in decimal in symbolic coding. Up to 15 (178) shifts can be specified for each instruction.

There are two types of shift instructions; arithmetic shifts which bypass the sign bit and logical shifts which move all 16 bits. Right arithmetic shifts move bits from position 1 to 2, 2 to 3, 3 to 4, etc., with bit 1 always set to the same state as the sign bit. The bit originally located in position 15 is shifted off. In left arithmetic shifts, the bits are moved from 15 to 14, 14 to 13, 13 to 12, etc., with zeros being loaded into position 1. The sign bit remains intact.

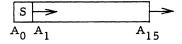
In right logical shifts, the sign is shifted to position 1, 1 to 2, 2 to 3, etc., and zeros are loaded into the sign bit position. If left logical shifts, final zeros are loaded into position 15 and bit 1 shifted to the sign position with the sign bit shifted off.

RSA 00-10

RIGHT SHIFT A -ACCUMULATOR

0.0.0	0.0	0	0		n	0	0	1	0	0	0
1	3	4	5	6		10					15

Bits A_1 through A_{15} are shifted right n number of places. The sign bit is unchanged, but supplies the bits (l's if negative, 0's if positive) shifted into A_1 as the original bits are shifted from A_{15} .



Timing:	If $n = 1 - 4 2$ cycles								
	n = 5 - 8 3 cycles								
	n = 9 - 12 4 cycles								
	n = 13 - 15 5 cycles								
Indicators:	None								
Registers Affected:	A-Accumulator								

FRA 00-12

FULL RIGHT ARITHMETIC SHIFT

0	0	0	0	0	0		n		0	0	1	0	1	0
1			3	4	5	6		9	10					15

Bits A₁ through A₁₅ and B₁ through B₁₅ are shifted right n places. Both sign bits are unchanged. A sign supplies bits to A₁ (1's if negative, 0's if positive) while A₁₅ supplies bits to B₁. The original B-Accumulator bits are shifted off from B₁₅.

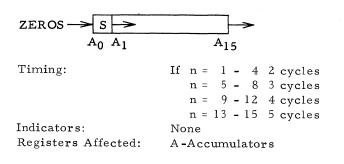
	¥
s ->	
A ₀ A ₁ A ₁₅	B ₀ B ₁ B ₁₅
Timing:	If n = 1 - 4 2 cycles n = 5 - 8 3 cycles n = 9 - 12 4 cycles n = 13 - 15 5 cycles
Indicators: Registers Affected:	None A-Accumulator, B-Accumulator

RSL 00-15

RIGHT SHIFT LOGICAL A -ACCUMULATOR

0 0	0 0	0	0		ņ		0	0	1	1	0	1
•	3	4	5	6		1	10					15

Bits A₀ through A_{15} are shifted right n places. Zeros enter A₀ as the original bits are shifted off A_{15} .



LSA 00-11

LEFT SHIFT A-ACCUMULATOR

0	0	0	0	0	0		ņ	0	0	1	0	0	1
			3	4	5	6	5	10					15

Bits A₁ through A₁₅ are shifted left n number of places. The sign bit is unchanged. Zeros are shifted into bit A₁₅ as the original bits are shifted from A₁.

S	ZEROS
	A ₁₅
Timing:	If $n = 1 - 4$ 2 cycles
	n = 5 - 8 3 cycles
	n = 9 - 12 4 cycles
	n = 13 - 15 5 cycles
Indicators:	None
Registers Affected:	A-Accumulator

FLA 00-17

FULL	LEFT	ARITHMETIC	SHIFT

0 0	0	0	0	0		ņ		0	0	1	1	1	1
0		3	4	5	6		9	10					15

Bits A₁ through A₁₅ and B₁ through B₁₅ are shifted left n places. The signs of the A- and B-Accumulators are unchanged. Zeros are supplied to B₁₅. The output of B₁ is applied to A₁₅, and the original A-Accumulator bits are shifted off from A₁.

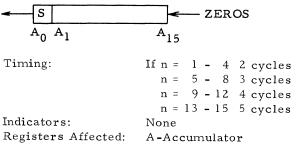
<		
S	S	ZEROS
A ₀ A ₁ A ₁₅	B ₀ B ₁	B ₁₅
Timing:	If $n = 1 - 4$ n = 5 - 8 n = 9 - 12 n = 13 - 15	3 cycles 4 cycles
Indicators: Registers Affected:	None A-Accumulator B-Accumulator	,

LSL 00-16

LEFT SHIFT LOGICAL A-ACCUMULATOR

0	0	0	0	0	0		n		0	0	1	1	1	0
8				4	5	6		9	10					15

Bits A_0 through A_{15} are shifted left n places. Zeros enter A_{15} as the original bits are shifted off A_0 .

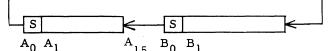


FRL 00-14

FULL ROTATE LOGICALLY

0.0	0	0	0	0		ņ		0	0	1	1	0	0
•			4	5	6		9	10					15

Bits A_0 through A_{15} and B_0 through B_{15} are rotated to the left n number of places. The bits from A_0 enter B_{15} and the bits from B_0 enter A_{15} .



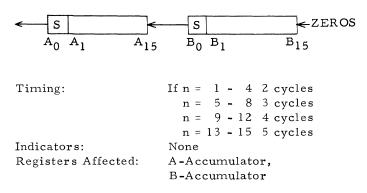
If $n = 1 - 4$ 2 cycles
n = 5 - 8 3 cycles
n = 9 - 12 4 cycles
n = 13 - 15 5 cycles
None
A-Accumulator,
B-Accumulator

FLL 00-13

FULL LEFT LOGICAL SHIFT

0	0	0	0	0	0		ņ		0	0	1	0	1	1
1			3	4	5	6		1	10					15

Bits A₀ through A₁₅ and B₀ through B₁₅ are shifted left n places. The bits from B₀ enter A₁₅, the original A-Accumulator bits are shifted off from A₀ and zeros enter B₁₅.



CONTROL INSTRUCTIONS

The five instructions in this group are used for general "housekeeping" functions required by the program. The HLT (Halt) instruction stops the computer after loading the next instruction into the instruction register. the NOP (No Operation) instruction performs no function other than to reserve a program slot for a future addition or to delay the program to match real-time input. The TOI (Turn Off Interrupt), PIE (Priority Interrupt Enable) and PID (Priority Interrupt Disable) allow program control of the priority interrupt circuits as described in the following paragraphs.

The HLT, NOP, and TOI instructions are singleword augmented 00_8 operation code words. The PIE and PID instructions are augmented 13_8 words and both include a second word containing the priority interrupt channel numbers. The second words are automatically read from memory as part of the normal execution cycle of the instruction.

HLT 00-00

HALT

0 0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8		3	4					9	10					15

Halts the operation of the computer.

NOTE

The computer stops with the address of the next instruction in the program counter. The START switch is closed to initiate the I cycle of the addressed instruction. The program counter may be manually reset and set to a new starting address prior to closing the START switch.

Timing:	n.a.
Indicators:	HALT
Registers Affected:	None

NOP 00-33

NO OPERATION

0	0,	0	0	0	0	0	0	0	0	0	1	1	0	1	1
				4	_				9	10					15

No operation is performed.

NOTE

This instruction is used to reserve memory locations for instructions to be added within the program encompassing that memory location. It may also be used to delay a program to match a real-time input or output transfer rate.

Timing:	l cycle
Indicators:	None
Registers Affected:	None

TOI 00-35

TURN OFF INTERRUPT

0	0.	0	0	0	0	0	0	0.	0	0.	1	1	1	. 0	1
			3	4					1	10					15

Sets a control latch associated with the highest active priority interrupt so that the interrupt will be reset by the next Long Branch or indirectly addressed Unconditional Branch instruction.

Timing:	l cycle
Indicators:	None
Registers Affected:	None

PIE 1306

PRIORITY INTERRUPT ENABLE

											V	٧o	rd	1	
1	0	1	1	0	0	0	1	1	0	0	0	0	0	.0	.0
1			3	4		6	7								15
											V	۷o	$\mathbf{r}\mathbf{d}$	2	
0	Bina Group	ry No.		12	11	Jnitar 10	у _, С	hanne 8	1 No. 7	6	V 5	V o	rd	2	1

Enables any combination of the 12 priority interrupt levels belonging to the selected priority interrupt group. Bits 15 through 4 of the second word are set to ONES to enable interrupt levels 1 through 12.

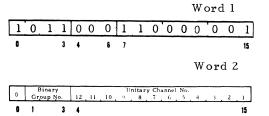
NOTE

This instruction allows a selected number of 96 priority interrupt levels (12 levels/group x 8 groups) to be enabled for operation.

Timing:	2 cycles
Indicators:	None
Registers Affected:	None

PID 1306-01

PRIORITY INTERRUPT DISABLE



Disables any combination of the 12 priority interrupt levels belonging to the selected priority interrupt group. Bits 15 through 4 of the second word are set to ones to disable levels 1 through 12.

NOTE

This instruction allows a selected number of 96 priority interrupt levels (12 levels/group x 8 groups) to be disabled.

Timing:	2 cycles
Indicators:	None
Registers Affected:	None

INPUT/OUTPUT INSTRUCTION

Input/output instructions contain 138 or 178 operation codes. Several of the input/output instructions are two word instructions. The two instruction words are stored in sequential memory locations with the second word called automatically by the 810B computer. The six I/O instructions are:

COMMAND EXTERNAL UNIT (CEU)

TEST EXTERNAL UNIT (TEU)

ACCUMULATOR WORD OUTPUT TO PERIPHERAL (AOP)

MEMORY WORD OUTPUT TO PERIPHERAL (MOP)

ACCUMULATOR WORD INPUT FROM PERIPHERAL (AIP)

MEMORY WORD INPUT FROM PERIPHERAL (MIP)

Two instructions, A Input (AIP) and A Output (AOP) are provided to enable words or characters to be transferred between the A-Accumulator and peripheral units. These instructions provide a convenient character assembly/disassembly capability. Each of these instructions occupies a single memory location. The two instructions, Memory Input (MIP) and Memory Output (MOP), enable words or characters to be transferred directly between specified memory locations and peripheral units. The instruction Command External Unit (CEU) enables all system devices connected to the computer to be controlled by the program. The CEU instruction is used to initiate Block Transfer Control units as well as to control computer peripheral devices and special system units. The Test External Unit (TEU) instruction is provided to enable system devices to be tested by the computer. The test result causes the instruction following the TEU to be either executed or skipped. Two memory locations are required to store the MIP, MOP, CEU and TEU instructions.

Data or command word transfer instructions can be executed in either of two modes - Wait Mode or Skip Mode, as defined in the following paragraphs.

a. <u>Wait Mode</u> - In this mode, the transfer is not made until the unit sends a "Ready" signal to the computer. The computer continues to test for the Ready signal each machine cycle and then executes the transfer during the first cycle following the recognition of the Ready signal. After the transfer, the device is disconnected and the next instruction in sequence is executed. The specific meaning of the Ready signal is defined in each I/O instruction description.

b. <u>Skip Mode</u> - In this mode, the Ready signal is tested only once. If the Ready signal is present, the transfer is executed. The Program Counter is then advanced to cause the next instruction to be skipped. If the device indicates "Not Ready", the device is disconnected and the Program Counter is advanced to cause the next instruction to be executed. This conditional skip features enables all I/O instructions (except TEU) to perform the total function of "Connect Unit, Test for Ready, Transfer if Ready, and Disconnect Unit".

The flow chart showing the execution of the AIP and AOP instructions is shown in figure 2-3. As shown in the flow chart, the value of the Wait Flag determines whether the instruction is executed in the Wait or Skip Mode. The MIP, MOP and CEU instructions are executed in the same manner, except that the Program Counter is advanced by one before the transfer is made in order to obtain the operand address.

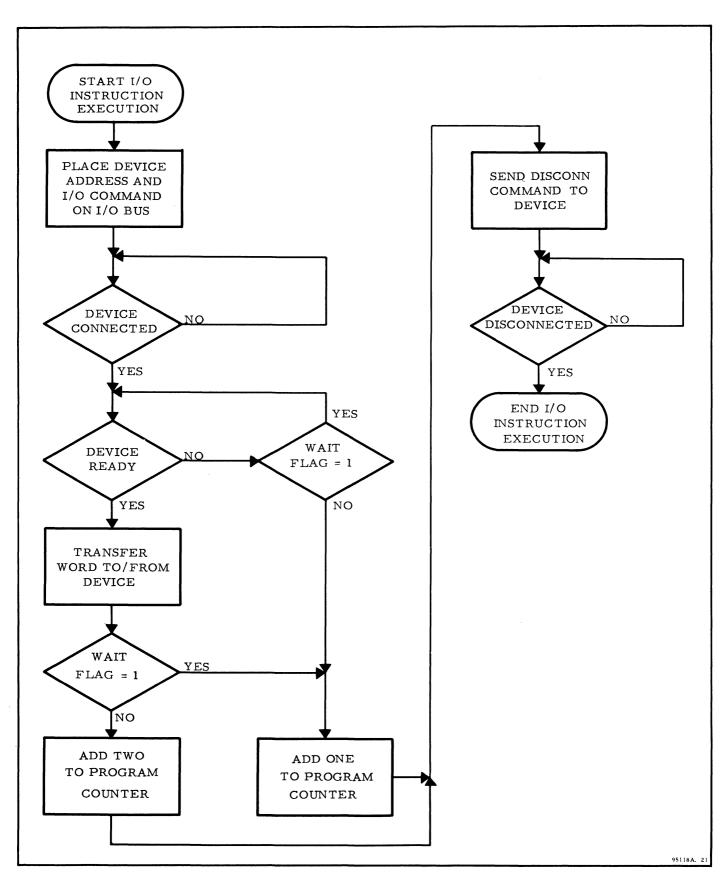


Figure 2-3. AIP/AOP Instruction Execution Flow Chart

Execution of the TEU instruction requires no Ready Test. An on-line unit is always "Ready" to be tested. The Test word is always transferred to the device and a Test Return signal is tested. The result of the test is a conditional skip of the next instruction.

In addition to providing selectable execution modes, the two-word I/O instructions (MIP, MOP, CEU, and TEU) provide two selectable operand addressing modes, Immediate Mode and Address Mode as defined in the following paragraphs.

a. <u>Immediate Mode</u> - In this mode, the second instruction word is treated as the operand. In executing MOP, CEU, and TEU instructions, the contents of the second instruction location are transferred to the specific unit. MIP execution consists of transferring a word or character from a specified device into the second instruction location.

b. <u>Address Mode</u> - In this mode, the second instruction word is interpreted as the operand address. The indirect address format is used in the second instruction word. Therefore, indexing and indirect chaining may be used in addressing the operand. The addressing mode is specified in the first instruction word by the value of the Indirect Address Flag (I). If I is a ONE, the Address Mode is executed.

The format of the I/O instruction words is in figure 2-4. The specific coding used in each instruction is defined in the individual instruction descriptions. Bit definitions are given in table 2-2.

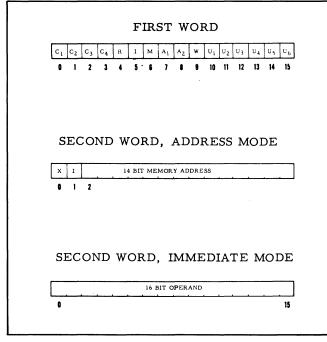


Figure 2-4. I/O Instruction Word Format

Symb.	Definition	Contained In
С	Command Code	All I/O Instructions
R	Character Merge Flag	AIP
Ι	Indirect Address Flag	MIP, MOP, CEU, TEU
М	Map Bit	MIP, MOP, CEU, TEU
А	Augmented Command Code	A11
W	Wait Flag	All except TEU
U	Unit Number (00 ₈ - 77 ₈)	A11
Х	Index Flag	MIP, MOP, CEU, TEU

Table 2-2. Bit Definitions

CEU	1300	Immediate – Skip Mode
	1301	Immediate – Wait Mode
	1320	Address - Skip Mode
	1321	Address - Wait Mode
	1330	Address - Map - Skip Mode
	1331	Address - Map - Wait Mode
COM		

COMMAND EXTERNAL UNIT

								Word 1	
1 0 1	1	0 1	M	0	0	W	DEVICE	(UNIT) NUMBE	R
0	3	4	6	1	8	9	10	1	5
		(4	Add	re	ss	M	ode)	Word 2	
XI		14.	Bi	t A	,dd	re	ss.		
8 1 2								1	15

Transfer the command (up to 16 bits) contained in the specified memory location to unit n.

Operand Address	Immediate -		I	=	0	
Modes:	Address - I =					
	(First Word)				

NOTE

M functions only if the Indirect Flag (bit 5) is a "1". If bit 5 and bit 6 are both "1" bits the MSB

	N	OTE (Cont'd)	Operand Add Modes:	lress	Immediate - I = 0 Address - I = 1			
	of the prog	gram counter is merged			(First Word)			
	with the Ir	direct Address. If						
	bit 5 is a	'l'' and bit 6 is a ''0''						
	the MSB o	f the Indirect Address			NOTE			
	is set to a	"0". This feature						
	allows the	program to be exe-						
	cuted in u	oper memory (MAP 40	M fr	M functions only if the Indirect Flag				
	or greater) in the same manner			a "l". If bit 5 and bit 6			
	as it is ex	ecuted in lower memory.	are	both	"l" bits the MSB of the counter is merged with the			
Execut	ion Modes:	Skip (W = 0), Wait (W = 1)	Indi	irect 1	Address. If bit 5 is a "1" is a "0" the MSB of the			
Transf	er Criterion	: A unit answers "Ready" to			Address is set to a "0".			
		a CEU test if the unit can			ture allows the program to			
		immediately start execu-			ted in upper memory			
		tion of any new function			or greater) in the same			
		command.	mar	nner a	as it is executed in lower			
			mer	mory.				
		NOTE		-				
	•		Execution M	odes:	This instruction is always			
	The bits in	n most unit command			executed in the same			
	codes are	micro-programmed.			mode. An on-line unit			
		ther one or several			is always "Ready" to			
	function co	ommands may be trans-			accept a test code.			
	ferred to a	a unit by execution of a			Therefore, the code			
	0	U instruction. Refer to			is always transferred			
	Section VI	for the definition of the			and the return is always			
	standard u	unit command codes.			tested. The Wait Flag is not used.			
Timing	:	4 cycles + wait						
Indicat	ors:	I/O WAIT	Timing:		4 cycles + wait			
Registe	ers Affected:	None	Indicators:		None			
			Registers Af	ffected	d: None			
TEU	1302	Immediate Mode		~~				
	1322		AOP 170	50	Skip Mode			
		Address Mode	170	D1	Wait Mode			
	1332	Address, Map Mode						
TEST	EXTERNAL	TINITT	ACCUMULA	TOR	WORD OUTPUT TO PERIPHERAL			
נדטבי		OUTT	1 1 1 1 0 0		0 W DEVICE (UNIT) NUMBER			
		Word 1	1 1 1 1 1 0 0 3 4		8 9 10 15			
1 0 1	1 0 I M 0	1 0 DEVICE (UNIT) NUMBER		-				

3 4

XII

0 1 2

67

14-Bit Address

8 9 10

(Address Mode) Word 2

Transfers the test code (up to 16 bits) contained in the specified memory location to unit n. A return signal from the unit is then tested, and the program counter is advanced accordingly. If the return signal indicates a "Ready" or "Go" condition, the next instruction in sequence is skipped. A return signal indicating a "Not Ready" or abnormal condition causes the next instruction to be executed.

15

15

Transfers a word from the A-Accumulator to unit n. Character oriented units accept only bits $A_0 - A_7$.

Execution Modes:	Skip (W = 0), Wait (W = 1)
Transfer Criterion:	A unit answers "Ready" to an AOP test if the unit can immediately receive a new word or character.
Timing: Indicators: Registers Affected:	4 cycles + wait I/O WAIT None

MOP 1704	Immediate, Skip Mode
1705	Immediate, Wait Mode
1724	Address, Skip Mode
1725	Address, Wait Mode
1734	Address, Map, Skip Mode
1735	Address, Map, Wait Mode

MEMORY WORD OUTPUT TO PERIPHERAL

											Word 1
Γ	1	1	1	0	I	Μ	1	0	W	DEVICE	(UNIT) NUMBER
0			ė	1		6	7	ÿ	э	10	15

(Address Mode) Word 2

Χ	Ι		14-E	3iț	Address		
0	1	2					

Transfers a word from the specified memory location to unit n. Character oriented units accept only bits $m_0 - m_7$ from the specified memory location.

Operand Address:	Immediate – $I = 0$
Modes:	Address - I = 1
	(First Word)

NOTE

M functions only if the Indirect Flag (bit 5) is a "1". If bit 5 and bit 6 are both "1" bits the MSB of the program counter is merged with the Indirect Address. If bit 5 is a "1" and bit 6 is a "0" the MSB of the Indirect Address is set to a "0". This feature allows the program to be executed in upper memory (MAP 40 or greater) in the same manner as it is executed in lower memory.

AIP	1702	Skip Mode
	1703	Wait Mode
	1742	Merge, Skip Mode
	1743	Merge, Wait Mode

ACCUMULATOR WORD INPUT FROM PERIPHERAL

1	1	1	1	R	0	0	0	1	W	DEVIC	CE (UNIT) NUMBER
8			3	4	5	6	1	8	9	10	15

Transfers a word or character from unit n into the A-Accumulator. Character oriented units transfer characters into bits $A_8 - A_{15}$.

NOTE

This instruction contains a convenient provision for character assembly in the A-Accumulator. If the optional Merge Flag (R) is a ONE, the input character is added to the contents of the A-Accumulator. If (R) is zero, the A-Accumulator is cleared prior to the input of a character or word. Therefore, an 8-bit character can be read, with R = 0, shifted left 8 bit positions then merged with the next character read with an R = 1.

E xecution Modes:

Skip (W = 0), Wait (W = 1)

Execution Modes:	Skip (W = 0), Wait (W = 1)	Transfer Criterion:	A unit answers "Ready" to an AIP test if the unit has a word or character ready for
Transfer Criterion:	A unit answers "Ready" to an MOP test if the unit can immediate receive a		immediate transfer.
	new word or charac- ter.	Timing:	4 cycles + wait
Timing: Indicators:	4 cycles + wait I/O WAIT	Indicators:	I/O WAIT
Registers Affected:	None	Registers Affected:	A-Accumulator

MIP 1706	Immediate, Skip Mode
1707	Immediate, Wait Mode
1726	Address, Skip Mode
1727	Address, Wait Mode
1736	Address, Map, Skip Mode
1737	Address, Map, Wait Mode

MEMORY WORD INPUT FROM PERIPHERAL

										Word	1
1 1	1	1	0	Ι	M	1	1	W	DEVICE (UNIT) NUMB	ER
0		3	4		ó	7	8	ý	10		15
(Address Mode) Word 2											
XI		14	- B	it.	Ad	dr	ess	3			
0 ;											15

Transfers a word or character from unit n to the specified memory location. Character oriented units transfer characters into bits mg - m_{15} of the specified memory location.

Operand	Address
Modes:	

Immediate	-	I	~	0
Address	-	I	=	1
(First Wor	d)			

NOTE

M functions only if the Indirect Flag (bit 5) is a "1". If bit 5 and bit 6 are both "1" bits the MSB of the program counter is merged with the Indirect Address. If bit 5 is a "1" and bit 6 is a "0" the MSB of the Indirect Address is set to a "0". This feature allows the program to be executed in upper memory (MAP 40 or greater) in the same manner as it is executed in lower memory.

Execution Modes:	Skip (W = 0), Wait (W = 1)
Transfer Criterion:	A unit answers "Ready" to an MIP test if the unit has

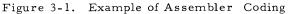
	a word or character ready for immediate transfer.
Timing:	4 cycles + wait
Indicators:	I/O WAIT
Registers Affected:	None

SECTION III ASSEMBLY LANGUAGE PROGRAMMING

GENERAL DESCRIPTION

The general format of the SEL 810B Assembly symbolic instruction input (source input) consists of five major fields. These fields are the Location, Operation, Address and Comments - Identification fields. Figure 3-1 shows a sample coding format that may be used for writing source programs in symbolic assembly language. The following paragraphs describe the coding format.

		IDENTIFICATION
LOC.	OPER.	ADDRESS, INDEX
1	6	11 25 72
*+++	+ ^B + ^I + ^N	ARY-PCTAL CONT- KS+
*+++	VER	SION AND TYPEOUT K/
++++	REL	
++++	NAME	^B t ^I t ^N t ^O t ^C t ^T t·t ^B t + t + t + t + t/t/t
B	DIAC	<mark>╶<mark>┼[╬]┾┊┊┊┊┊┊┊┊┊┊┊┊┊</mark></mark>
+++	TIAB	┃
++++	LIX	
-+++-	DATA	
+++	CLA	+++++++++++++++++++++++++++++++++++++++
+++	FLL	
-+ + +-	BRU	*++2
LOOP	FLL	4 3 + + + + + + + + + + + + + + + + + +
++++	A _M A	
+++	LSL	
-+++-	AOP	
	I X S	
+ + +	BIRU	
	B,R,U,*	B + + + + + + + + + + + + + + + + + + +
	E N D	
		95118A. :



LOCATION FIELD

The Location field (columns 1 - 4, in figure 3-1) may consist of a symbolic label for the instruction

line when it becomes necessary to refer to this location elsewhere in the program. The symbolic label consists of 1 to 4 characters; the first character <u>must</u> be a letter and the remaining characters may be either letters or digits. If no reference to the instruction line is necessary, the Location field may be left blank.

OPERATION FIELD

The Operation field (columns 6 - 9, in figure 3-1) consists of a mnemonic computer instruction or pseudo-operation. A list of mnemonic instructions and pseudo-operations is given in tables 3-2 and 3-4.

Mnemonic computer instructions consist of 3 letters. The mnemonic instruction must be "left-justified" in the Operation field, i.e., written in columns 6, 7 and 8. If the instruction address is to be made indirect, the 3-letter mnemonic is followed by an asterisk (*) written in column 9.

Pseudo-operations consist of 3 or 4 letters and represent either data definition or instructions to the assembly program.

ADDRESS FIELD (VARIABLE FIELD)

Memory reference instructions use the variable field to define the operand address which may be followed by a comma and a one (, 1), to signify indexing. Some other instructions, such as shift and I/O machine operations and some pseudooperations, have special formats for the variable field which are defined in tables 3-1, 3-2, 3-3 and 3-4. If no address field definition is required the address field is left blank.

Operand Address Formats

<u>No Address</u>. The address field may be left blank if no operand address is required.

Symbolic Address. Consists of 1 to 4 characters starting with a letter.

External Symbolic Address. An external symbolic address consists of a dollar sign (\$) followed by 1 to 6 characters, the first of which is a letter. This external variable is presumed not defined within the program in which it is contained but refers to a subroutine or item located in a different subprogram or located on the library tape. No address arithmetic may be performed on External Symbolic Addresses.

Absolute Address. When reference to a fixed memory location is required or when the address represents a count (such as in a shift instruction). The address consists of digits only and is presumed decimal. Octal addresses are preceded by an apostrophe (').

<u>Current Location</u>. The location of this instruction is used as the instruction's address if a single asterisk (*) appears in the address sub-field. This allows for reference to this or nearby instructions without having to assign a symbolic name to that instruction.

Address Arithmetic. Any current location (*), symbolic (NAME), or absolute ('1234) address may be joined with a constant, current locations (*), symbolic (NAME) or absolute (1234) address by an intervening plus (+) or minus (-) operator to define an effective address (NAME + 4). The above may be extended to more than two operands (A - B + 2).

Literal Address. Literal addresses allow a constant to be defined, assigned to a memory cell and that assignment location used as the address for this instruction. All constants defined in literal addresses will optimize storage so that all identical constants (regardless of their format) will be assigned only once. A literal address consists of an equal sign (=) followed by the constant. Any decimal integer, octal number, single asterisk (current location), previously defined symbolic name or combination of these formats joined by a + or - may follow the equal sign in a literal address.

Location To Be Filled. A double asterisk (**) indicates the address portion of this instruction is to be filled in by the object program at run time and is identical to an absolute address of 000.

The assembly program presumes the computer has a 15-bit address and, therefore, does not attempt to reduce the argument address to a 9-bit address. When the resulting object tape is loaded by the loader into memory starting at a location determined by the operator, these 15-bit addresses are modified as follows:

(a) If the argument address is located in MAP zero, the address is truncated to 9 bits and the MAP bit is set to zero.

(b) If the argument address is located in the same MAP as the instruction in which it i ${f s}$

contained, the address is truncated to 9 bits and the MAP bit is set to one.

(c) Otherwise, truncate the 15-bit address to 14 bits and store the 14-bit address and its indirect and index bits automatically into a cell in MAP zero, set the 9-bit address of this cell in the instruction being loaded, and set the MAP bit to zero and the indirect bit to one.

Table 3-1 lists examples of address field entries.

Table 3-1. Example Address Field Entries

Field Entry	Description
	No Address
0, 1	Absolute Zero Address, Indexed
ALPH	Symbolic Address
ALPH, 1	Symbolic Address and Index
519	Absolute Decimal Address
'1067,1	Absolute Octal Address, Indexed
NAME+4	Address Arithmetic
COMN-2, 1	Address Arithmetic, Indexed
ALPH-PHPA+2,1	Symbolic Address Arith- metic Indexed
= 100	Literal Decimal Constant
='41237	Literal Octal Constant
*	Current Location
*-3, 1	Near-by Address, Indexed
**	Address to be Filled
**, 1	Indexed Address to be Filled
\$SQRT	External Symbolic Address

COMMENTS FIELD

The comments field starts immediately after the first space in the variable address field. This field has no effect on the assembler but is printed out on the symbolic listing if a listing is requested. Any line which has an asterisk (*) in the first character position of that line will be considered a line of comments. (See line 1 in figure 3-1.)

Because of width limitations on the typewriter, comments appearing after column 50 will be output only on the line printer.

IDENTIFICATION FIELD

This field is not checked and is considered as part of the comments. It is provided as a programmers aid. For example, it may be used to identify a card or cards in a card deck or for sequencing the card deck. It is located in columns 73 thru 80.

MNEMONIC COMPUTER INSTRUCTIONS

The computer instructions listed in table 3-2 will be accepted by the SEL 810B assembly program. All permissible fields are shown in the Allowable Fields column with all required fields underlined. Any of the described symbolic notations in this manual may be used in the variable fields, providing they are defined.

Absolute notations for the variable fields are shown in table 3-3.

MNEMONIC Instruction	Allowable Fields	Description
AMA	AMA* <u>Addr</u> , l	Add Memory to A-Accumulator
AMB	AMB* <u>Addr</u> , l	Add Memory to B-Accumulator
SMA	SMA* <u>Addr</u> , l	Subtract Memory from A-Accumulator
МРҮ	MPY* <u>Addr</u> , l	Multiply B-Accumulator times Memory
DIV	DIV* <u>Addr</u> , l	Divide A and B-Accumulator by Memory
RNA	RNA	Round A-Accumulator by MSB in B-Accumulator
ovs	ovs	Overflow Set
LAA	LAA* <u>Addr</u> , l	Load A-Accumulator from Memory
LBA	LBA* <u>Addr</u> , l	Load B-Accumulator from Memory
STA	STA* <u>Addr</u> , l	Store Memory from A-Accumulator
STB	STB* <u>Addr</u> , l	Store Memory from B-Accumulator
LIX	LIX DATA	Load Index (Immediate Mode)
STX	LIX* DAC* <u>Addr</u> , l STX	or (Address Mode) Store Index
	DATA	(Immediate Mode)
	STX* DAC* <u>Addr</u> , l	or (Address Mode)
LCS	LCS	Load Control Switches in A-Accumulator
BRU	BRU* <u>Addr</u> , 1	Unconditional Branch
SPB	SPB* <u>Addr</u> , l	Store Place and Branch

Table 3-2. SEL 810B Mnemonic Instructions

MNEMONIC Instruction	Allowable Fields	Description			
SNS	SNS Switch no.	Skip if Console Switch Not Set			
IMS	IMS* <u>Addr</u> , l	Increment Memory and Skip			
СМА	CMA*Addr, l	Compare Memory and A-Accumulator (3 way $n+1(-)$, $n+2$ (0), $n+3$ (+)			
IBS	IBS	Increment B-Accumulator (Index) and Skip			
SAZ	SAZ	Skip if A-Accumulator is Zero			
SAP	SAP	Skip if A-Accumulator is Positive			
SAN	SAN	Skip if A-Accumulator is Negative			
SOF	SOF	Skip NO Overflow			
SAS	SAS	Skip on A-Accumulator sign (3 way) n+1 (-), n+2 (0), n+3 (+)			
SNO	SNO	Skip if A-Accumulator is not Normalized			
LOB	LOB	Long Branch			
	EAC Addr, 1				
SXB	SXB	Skip if Index Pointer is Set to B-Accumulator			
IXS	IXS	Increment Index and Skip if Positive			
ABA	ABA	AND A-Accumulator and B-Accumulator			
OBA	OBA	OR A-Accumulator and B-Accumulator			
NEG	NEG	Negate A-Accumulator			
ASC	ASC	Complement A-Accumulator Sign			
CNS	CNS	Convert Number System			
CLA	CLA	Clear A-Accumulator			
TAB	TAB	Transfer A-Accumulator to B-Accumulator			
IAB	IAB	Interchange A-Accumulator and B-Accumulator			
CSB	CSB	Transfer B-Accumulator Sign to Carry and Clean B-Accumulator Sign to Positive			
TBA	TBA	Transfer B-Accumulator to A-Accumulator			
TAX	TAX	Transfer A-Accumulator to Hardware Index Register			

Table 3-2. SEL 810B Mnemonic Instructions (Cont'd)

MNEMONIC Instruction	Allowable Fields	Description					
TXA	TXA	Transfer Index Register to A-Accumula- tor					
TAP	TAP	Transfer B-Accumulator to Protect Register					
TPA	TPA	Transfer Protect Register to B-Accumulator					
TBV	TBV	Transfer B-Accumulator to Variable Base Register (VBR)					
TVB	TVB	Transfer Variable Base Register (VBR) to B- Accumulator					
ХРХ	ХРХ	Set Index Pointer to X Index Regis- ter					
ХРВ	ХРВ	Set Index Pointer to B-Accumulator					
RSA	RSA <u>Count</u>	Right Shift A-Accumulator					
LSA	LSA <u>Count</u>	Left Shift A-Accumulator					
FRA	FRA <u>Count</u>	Right Shift A-Accumulator and B-Accumulator					
FLA	FLA <u>Count</u>	Left Shift A-Accumulator and B-Accumulator					
RSL	RSL <u>Count</u>	Right Logical Shift A-Accumulator					
FRL	FRL <u>Count</u>	Logical Rotate A-Accumulator and B-Accumula					
LSL	LSL <u>Count</u>	Left Logical Shift A-Accumulator					
FLL	FLL <u>Count</u>	Logical Left Shift A-Accumulator and B-					
HLT	HLT	Accumulator Halt					
NOP	NOP	No Operation					
TOI	TOI	Turn Off Interrupt					
PIE	PIE DATA <u>Group & Level</u> PIE* DAC* <u>Addr</u> , l	Priority Interrupt Enable					
PID	PID DATA <u>Group & Level</u> PID* DAC* <u>Add</u> r, l	Priority Interrupt Disable					
AOP	AOP <u>Unit</u> , Wait	A-Accumulator Out to Unit (n) (Without wait, skip on ready)					
AIP	AIP <u>Unit</u> , Wait, Merge	A-Accumulator Input from Unit (n) (Without wait, skip on ready)					

Table 3-2. SEL 810B Mnemonic Instructions (Cont'd)

MNEMONIC Instruction	Allowable Field	Description
МОР	MOP <u>Unit</u> , Wait	Memory Out to Unit (n)
	DATA OR MOP* <u>Unit</u> , Wait, MAP DAC* <u>Addr</u> , 1	(Immediate Mode) OR (Address Mode)
MIP	MIP <u>Unit</u> , Wait, DATA OR MIP* <u>Unit</u> , Wait, MAP DAC* <u>Addr</u> , 1	Memory Input from Unit (n) (Without wait, skip on ready) (Immediate Mode) OR (Address Mode)
CEU	CEU <u>Unit</u> , Wait, DATA OR CEU* <u>Unit</u> , Wait MAP DAC* <u>Addr</u> , 1	Command External Unit (n) (Without wait, skip on ready) (Immediate Mode) OR (Address Mode)
TEU	TEU <u>Unit</u> , DATA OR TEU* <u>Unit</u> , MAP DAC* <u>Addr</u> , 1	Test External Unit (Immediate Mode) OR (Address Mode)

Table 3-2. SEL 810B Mnemonic Instructions (Cont'd)

ABSOLUTE NOTATIONS FOR VARIABLE FIELDS

Table 3-3 lists the absolute notations for the variable fields.

Variable Field	Absolute Notation
Addr	5 octal digits ('00000-'77777) 5 decimal digits (00000- 32767)
Count	2 octal digits ('00-'17) 2 decimal digits (00-15)
Switch No.	2 octal digits ('00-'17) 2 decimal digits (00-15)
Group Level	l octal digit (0-7 = bits 1-3) 4 octal digits (0001-7777=
Unit	bits 4-15) 2 octal digits ('01-'77 rep- resenting units 1 to 63) or 2 decimal digits (1 to 63)

Table 3-3. SEL 810B Absolute Notation Formats (Cont'd)

Variable Field	Absolute Notation
Wait	1 binary bit (0, No Wait or 1, Wait = bit 9)
Merge	l binary bit (0, No Merge or 1, Merge = bit 4)

PSEUDO-OPERATION INSTRUCTIONS

This group of instructions is used to instruct the SEL 810B Assembly Program and are not executed by the computer. A description of each pseudo-operation is given in the following paragraphs.

- ABS Set the mode of the assembly program to ABSolute. When in this mode, all symbolic addresses will be assigned relative to location 00000 and output in a non-relocatable format.
- REL Set the mode of the assembly program to

RELative. When in this mode, all symbolic addresses will be assigned relative to the start load address (assigned when loading the program into memory) and output in a relocatable format compatible with the loader. The assembly program is initialized to the absolute mode and will remain in this mode until changed by an REL pseudo-op).

- ORG The variable field specifies an address. When the assembly is in absolute mode, this address specifies the location of the next instruction. When the assembly is in relative mode, this address will be added to the start load address (assigned when loading the program into memory) in order to specify the location of the next instruction. In either case, all following instructions will be stored sequentially in memory until another ORG pseudo-op is given.
- EQU The symbol in the location field will be assigned the address or value specified in the variable field. Constant values may not exceed 15 bits.
- DAC This pseudo-op is used to generate Direct Address Constants used as argument addresses for subroutine calling sequences, or referred to by indirect instructions. The address in the variable field may be in any of the formats shown previously (variable field formats for instructions). The address will be truncated to 14 bits and will occupy bits 2 to 15 of the resulting word. The address may be indexed and the pseudo-op may be tagged indirect if required. (Setting bits 0 and 1.)
- EAC This pseudo-op is used to generate 15bit Extended Address Constants used as arguments of Long Branch instructions. Any of the Formats shown previously are acceptable in the variable field, except that the instruction may not be indexed nor made indirect.
- DATA The variable field of this pseudo-op may contain any number and any mixture of the following data item formats. If the location field contains a symbol, it will be assigned the location of the first data item. If more than one data item is present (separated by commas), they will be assigned sequential storage locations.

- a. Octal Data Item Format: An optional sign (+ or -), followed by an apostrophe character ('), followed by 0 to 6 octal digits (0 through 7). If less than 6 digits are present, the number will be right-justified with leading zeros added. If a minus sign is present, the number will be 2's complemented; a plus sign is ignored.
- b. Decimal Integer Format: An optional sign (+ or -) followed by 0 to 5 decimal digits (0 through 9). The number will be converted to binary and stored at a scale of B15. The number will be stored positively unless a minus sign is present. A minus sign will cause the 2's complement of the number to be stored.
- c. Fixed-Point Single Precision Decimal Data -Format: An optional sign (+ or -), 0 to 5 decimal digits (0 through 9), mixed with an optional decimal point, the letter B, followed by a decimal number between +15 and -15. Example: -3.14157B6. A minus sign will cause the 2's complement of the number to be stored. One word will be generated.
- d. Fixed-Point Double Precision Data Format: An optional sign (+ or -), 0 to 10 digits mixed with an optional decimal point, the letter C, followed by a decimal number between +30 and -30. Example: 103.637942C10. A minus sign will cause the 2's complement of the number to be stored. Two words will be generated.
- e. Floating Point Data Format: An optional sign (+ or -), 0 to 7 decimal digits (0 through 9), mixed with an optional decimal point, and optionally followed by a decimal exponent consisting of the letter E, preceding a decimal number between +75 and -75. (Either the decimal point, the letter E, or the sign of the exponent must be present.) Two sequential memory cells are generated for each floating point data item using the following format:

s	•	• •	•	۰F	ŀ	•	•	·	•	•	•	•	•
0	•	۰F2	•	•	•	•	۰E	:•	•	•	•	•	

Examples:	0.1	=	+63146+31775
-	5.0325E2	=	+76720+00011
	-1.E-1	=	-14631+47775
	-503.25	=	-01060+00011

f. Floating Point Double Precision Data -Format: An optional sign (+ or -), 0 to 11 digits mixed with an optional decimal point, the letter D, followed by a decimal number between +75 and -75. Three sequential memory cells are generated for each double-precision floating point item using the following format:

s		•	•	·	۰Fŀ	•	•	•	•	•	•	•	•	•
0	•	•	•	•	۰F2	•	•	•	۰E	:•	•	•	•	•
0		•	•	•	۰F3۰	•	•	•	•	•	•	•	•	•

E = Characteristic (2's complement if negative).

F1, F2, F3 = Double-Precision Fraction (2's complement if negative).

- g. Alphanumeric Data Format: Two apostrophe characters (' ') followed by any number of characters (including blanks) until another pair of apostrophes is read. The characters within the apostrophe pairs are stored 2 per word (last character leftjustified, if necessary).
 - Example: "ALPHA TEST" is to be stored into memory starting at location 2000.

2000	1	100	000	111	001	100	$^{\rm AL}$
2001	1	101	000	011	001	000	PH
2002	1	100	000	110	100	000	A-
2003	1	101	010	011	000	101	TE
2004	1	101	001	111^{-1}	010	100	ST

- The above example is in ASR-33 code (FULL ASCII code). This code will be used internally by the assembler to represent alphanumeric data. The I/O handling subroutines will translate from external to internal code and vice versa when necessary depending upon the I/O device in use.
- h. Symbolic Address Data Format: Any symbolic address optionally followed by address arithmetic. The effective address will be stored in memory as a 15-bit address (similar to that generated by the EAC pseudo-op). The address may not be tagged as indexed or indirect.

FORM This pseudo-op is used to setup the format for the FDAT pseudo-op. There is no data generated and no memory locations are used. This pseudo-op allows the programmer to define the bit assignments of 16-bit words generated by the FDAT statement. Up to 8 fields are allowed but the total number of bits must not exceed 16. All FDAT statements that follow a FORM will be in the same format until another FORM is encountered.

Example:

FORM 6, 4, 3, 1, 2

This assigns the FDAT bits as follows:

Field 1 - 6 bits (bits 0-5) Field 2 - 4 bits (bits 6-9) Field 3 - 3 bits (bits 10-12) Field 4 - 1 bit (bit 13) Field 5 - 2 bits (bits 14-15)

FDAT This pseudo-op is used to generate data in a format which has been previously defined by a FORM statement. The variable field for this instruction will accept decimal, octal, and alphanumeric data, but will mask off the most significant bits not defined by the previous FORM statement. Multiple FDAT statements may be placed on a card separated by slashes (/). If the location field contains a symbol, it will be assigned the location of the first data item. Example (using the FORM defined above):

FDAT ''A'', 8, 7, 0, 1/'75, '13, 4, 1, 3

This will generate the following two consecutive octal words:

'003071 '173347

BSS Reserve a block of memory storage starting at the current location and extending for the number of words specified in the variable field. (If the variable field is symbolic, it must have been defined by a previous input line). The location field is optional but if a symbol is inserted in this field, it will refer to the first word in the block.

BES Same as BSS except that if the location field is occupied, it refers to the <u>last+l</u> word in the block.

This pseudo-op will generate the nec-CALL essary coding and actions to call in a subroutine from a library tape into memory. The CALL pseudo-op is then replaced by a subroutine transfer instruction (SPB) to the subroutine. The variable field contains the subroutine name. The location field, when occupied, refers to the resulting SPB instruction. Logic is contained within the loader to assure that only one copy of a subroutine is called into memory from the library tape regardless of the number of CALL's for that subroutine. The subroutines name must start with a letter and may contain from 1 to 6 characters. An equally good way to call external subroutines would be with a leading dollar sign on the subroutine's name.

Examples: SPB \$SQRT

or CALL SQRT

NAME When writing subroutines for inclusion into a library tape, the name by which the subroutine must be called as specified by the NAME pseudo-op. This must appear as the subroutine's first instruction line(s). The variable field consists of two symbolic names. The first is the name of the subroutine and is 1 to 6 characters long (FORTRAN compatible). The second name is the symbolic entry location for the subroutine and is 1 to 4 characters long, the first character being a letter. More than one NAME pseudo-op may be included in a subroutine if alternate names for the subroutine exist with either the same or different entry

points. Also, external variables are defined by the NAME pseudo-op.

- ZZZ The instruction bits (0 to 3) are set to 0000. The rest of the instruction is determined by the variable field and the presence of an indirect indicator (*) following the pseudo-op (ZZZ*).
- *** Same as ZZZ but indicates the instruction will be filled at run time.
- MOR This pseudo-op causes a pause in the assembly process useful when the source program is on more than one tape, and a pause is needed to change tapes.
- END This pseudo-op must appear as the last instruction in any program or subroutine being assembled and tells the assembly program that assembly is complete. If the variable field is not blank, it should specify the starting location of the program just assembled.
- LIST Set the mode of the symbolic output routine to list the output provided sense switch one is not ON. The assembler assumes the LIST mode until otherwise directed.
- NOLS Set the mode of the symbolic output routine to suppress the listing of output unless an error is detected. This pseudo operation remains in control until a LIST pseudo-operation is encountered.

SUMMARY OF PSEUDO-OPERATION INSTRUCTIONS

Table 3-4 summarizes the SEL 810B pseudo-operation instructions and gives specific examples for their use.

Symbolic Location	Pseudo- Operation	Variable Field Entry	Description
	ABS		Set Mode Absolute
	REL		Set Mode Relative
	MAP		Set Single Map Mode
	ORG	'1000	Set Origin of Program
ALPH	EQU	BETA+2	Set Symbol Equal to Symbol

Table 3-4. Summary of SEL 810B Pseudo-Operations

Symbolic	Pseudo-	Summary of SEL 810B Pseudo-O	
Location	Operation	Variable Field Entry	Description
IND	EQU	2	Set Symbol Equal to Number
OCT	DATA	'12734, -'21, +'6470	Octal Data
	DATA	9876, -3000, +24	Decimal Integer Data
MIX	DATA	23.456B10, -3B6, 12CO	Fixed Point Data
FLOT	DATA	22.3344EO,. 12345D2	Floating Point Data
ALPA	DATA	"HELP", "12-34, A.E2"	Alphanumeric Data
	DATA	X4, TEST+2, A-DLTA+1	Symbolic Address Data
	DATA	3,'77,1.23B4, -1233E-3,X4	Mixed Format Data
	FORM	6, 4, 3, 1, 2	Defines FDAT Format
	FDAT	"A", 8, 7, 0, 1/'75, '13, 4, 1, 3	Mixed Formatted Data
TABL	BSS	100	Block Storage Skip (Front Label)
	BES	5	Block Storage Skip (End Label)
	CALL	SIN	Library Tape Call
	NAME	SIN, S21	Library Subroutine Name
	ZZZ	ALPHA	Instr. Bits = 0000
	***	**	Word to be filled at run Time
	MOR		Pause When Assembling
	END	STRT	End of Program
	END		End of Subroutine
AD2	DAC	TIME, 1	Direct Address Constant, Indexed
AD1	DAC*	LEVL	Indirect Address Constant
	EAC	MEM2	Extended Address Constant
AD3	EAC	MEM2, 1	Extended Address Constant, Indexed
	LIST		List
	NOLS		No List

Table 3-4. Summary of SEL 810B Pseudo-Operations (Cont'd)

MACRO SYSTEM

The Macro System generates in-line coding

according to the respective prototype and parameter list assigned to a given Macro call name.

The general form of a Macro prototype is as follows:

Loc.	Oper.	Address, Index		
NAME	MACR EMAC	A SET OF DETAIL STATE- MENTS		
Columns 1-4		The call name of the Macro which can be any combina- tion of legal characters, blanks included.		
Column 5		Blank		
Columns 6-9		To denote the beginning of a prototype code MAC or MACR. To denote the end of a prototype code EMA or EMAC.		
Column 10		Blank		

Columns 11-72 Can be used as comments.

SPECIAL NOTE

Do not use an END or a \$ end of job statement in a Macro prototype.

MACRO PROTOTYPE

The prototype is a set of detail statements which can contain elements to be supplied either internally or from a list of parameters. Elements are of three basic types as follows:

- Internal to a given Macro prototype
- Parameter supplied by user
- Fixed element name

The internal assignment applies only to labels and must be of the form @X where the at sign (ASCII 300) must be the first character of the label. The X is a decimal value from one through 16 and can be assigned in any order (not necessarily monotonically) per Macro. Leading zeros are suppressed, @009 is the same as @9. Each call of the same Macro which contains internal labels will generate a unique respective set increased by the last assembler assigned label plus one. The assembler will not allow more than 999 internal labels to be generated. All assignments in excess of 999 will be flagged as an error. Example of internal label:

Loc.	Oper.	Address, Index
WAIT	MACR	NAME AND BEGIN PRO- TOTYPE
@1	NOP	INTERNAL LABEL FIXED OP
	NOP NOP BRU @1	FIXED OPERATION CODE FIXED OPERATION CODE FIXED OP CODE, INTER- NAL LABEL
	EMAC	END OF WAIT PROTO- TYPE

Every call to WAIT will generate @l into a unique label for each wait loop.

The user supplied parameter can apply to any field of a valid assembler statement. The form of a user parameter is #X where the number sign (ASCII 243) may appear anywhere in a label or value to be specialized and must be immediately followed by a decimal value from one through 16 representing the correct parameter number to be concatenated into the generated element. Leading zeros are suppressed on parameter number assignments. Parameters which are requested but omitted from the list are replaced by a single blank character. Parameter numbers in excess of 16 will not be processed and will be flagged as an error.

Example of user supplied parameter:

Loc.	Oper.		Address, Index				
FILL	MACR		NAME AND BEGIN PROTOTYPE				
	LAA	= #1	CHARACTER TO FILL				
	LBA	=#2	SIZE OF AREA IN DECIMAL				
	STA	#3+#2,1	AREA PLUS SIZE MINUS INDEX				
	IBS		INCREMENT INDEX				
	BRU EMAC	*-2	LOOP TO FILL AREA END OF MACRO PRO- TOTYPE				
		FORM OI	F THE CALL				
* STAT	EMENT IS						
*SAMPI		DATA,	SIZE, AREA				
	MFILL '240,80, TABL,						
*FILL V	VITH A SPA	CE					
*80 LOC	CATION TAI	BLE					
*LEFT	ADDRESS IS	5 CALLE	DTABL				

To call a Macro prototype code a M in column 5 followed by the name of the Macro in columns 6-9 with the parameter list in columns 11-72.

Parameter elements in the main program call list are separated by a single level of delimiter which can be a comma, left parenthesis, or right parenthesis. The parameter list may be terminated by one of the three delimiters. Extra sets of parenthesis can be added for clarity but each must be counted when assigning values to the elements of a detail entry. Elements can be assigned in any order or any set of digits provided a parameter exist for the desired elements.

Example:

I	Loc.	Oper.	Address, Index			
P	AR	MACR	USE ONLY EVEN NUM- BERED PARAMETERS			
		DATA EMAC	#2, #4, #6, #8, #10			
		• •				
		MPAR	(13) (6) (''A'') (3B5) ('377)			
* *	THE ABC FOLLOW		THE SAME AS THE			
			13, 16, "A", 3B5, '377, A PROTOTYPE OF THE			
F	PAC		USE SEQUENTIALLY NUM- BERED PARAMETERS #1, #2, #3, #4, #5			
		EMAC	" 1 , " 2 , " 3 , " 1 , " 3			
*	* NOTE THAT A, () MAY NOT BE USED AS * DATA IN A PARAMETER LIST, IF NEEDED,					

The fixed element name is any field in which the detail statement supplies the value, operation code, operand or any portion of a statement.

Example:

	Loc.	Oper.		Address, Index
	LDB1	MACR		LOAD DOUBLE PRE-
				CISION CONSTANT 1
		CLA		CLEAR THE MSB TO
				ZERO
		LBA	=1	LOAD A 1 IN LSB
		EMAC		END OF MACRO PRO-
				TOTYPE
*	USAGE OF	LDB1		
	1	MLDB1		NOTE THAT NO
×	PARAMETERS ARE NEEDED ALL DATA IS			
*	SUPPLIED	BY THE	PRO	TOTYPE

Comments may be entered in a prototype; however, only the asterisk and the next 24 positions will be retained when the prototype is specialized. If a detail line has comments as a continuation of a statement, they will not be processed at the time of specialization.

The MACRO storage area is normally 700_{10} words with a name table capacity of 30 names. The formula for computing the exact number of words needed to store a prototype is as follows:

Sum of words for each statement +1.

The words for each statement = 1 + (number of characters in location field + number of characters inop code field + number of characters in variable $field) <math>\div 2 + 0$ if the remainder is 1, 1 if the remainder is 0. Count internal and parameter supplied labels as 2, that is, #003 + #02 + 6 is counted as 7 characters. A general safe rule-ofthumb would be to multiply the number of detail lines by 5 to obtain the storage requirements for a Macro prototype.

SECTION IV INPUT / OUTPUT

GENERAL DESCRIPTION

The 810B Computer Input/Output (I/O) structure is designed particularly to meet the requirements of the on-line, real-time computer user. This computer application area imposes the most severe requirements on computer I/O capabilities due to the wide variety of peripheral devices required and the time-sharing mode of operation encountered. Many Systems Engineering Laboratories real-time systems not only have standard data processing peripherals such as card, paper tape, magnetic tape, disc and keyboard/printer devices, but also have a number of interface devices such as data acquisition systems, displays and control and communication units. Therefore, the I/O structure must enable connection of a large number of peripheral devices to the computer and must enable several devices to time-share communication with the computer.

The standard 810B Computer I/O structure consists of an Input/Output Processor (I/OP), which provides "party line" communication with peripheral devices or device controllers. Data is supplied over 64 direct information channels. Figure 4-1 shows how peripheral units are connected to the I/OP by means of the I/O Bus.

The standard I/OP alone is capable of meeting the I/O requirements of many systems. It is ideally

suited to real-time applications in that each I/O instruction causes the device addressed by the instruction to be connected to the computer, the data transfer to be made, and then the device to be disconnected. Therefore, the successive transfer of data words to/from two or more different peripheral devices requires no intervening housekeeping operations such as channel and device testing and connection.

The time-sharing capability of the I/OP is further enhanced by the fact that all Systems Engineering Laboratories peripheral devices contain their own data buffers. Hence, the I/OP which contains no buffer, is never busy buffering data to be transferred to/from a device. As a result a two word data transfer instruction can be executed in 3.0 microseconds + wait and the I/OP released immediately for data transfer to/from a different device.

In addition to the I/OP, Block Transfer Control (BTC) units can be added to the 810B to provide a fully-buffered data transfer capability between computer memory and peripheral units. BTC channels enable a block of words up to 32,767 in length to be transferred to or from a peripheral device. One memory cycle is stolen per word transferred.

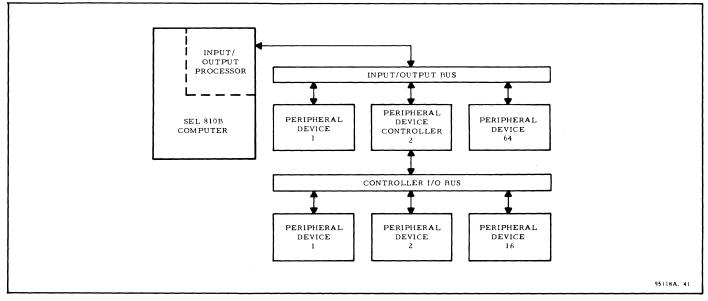


Figure 4-1. Connection of Peripheral Units to the Computer

The predominant reason for adding BTC units to the computer is to free the mainframe to perform internal processing functions while data is being transferred between memory and peripheral devices at high rates. For example, using two BTC units, a continuous stream of data words can be read into computer memory, blocked, and recorded on magnetic tape in gapped format, resulting in a loss of only slightly over two machine cycles (one for input, one for output) per word transferred. For a typical word rate of 20 KC, only an average of 1.50 microseconds of machine time is used per 50 microseconds of elapsed time to accomplish a single word input and output transfer function. The remainder of the time is available for performing such functions as scaling or limit checking of the data.

The optional Computer Graphics Processor (CGP) Model 84-235B is a high-speed data transferring control unit designed to satisfy the specialized needs of the SEL Computer Graphics Systems. The control unit is similar to the 810B optional Block Transfer Control (BTC) with the exception of its specialized operating characteristics and added control functions. A BTC, when outputting data, is unmindful of the nature of this data. However, the CGP examines each word as it comes from memory and either interprets the word as data and sends it to the Computer Graphics System or as an instruction and takes appropriate action.

The instructions allow the CGP to operate on its address counters, thereby freeing the 810B Computer from much of the control unit servicing, and allowing it more time to operate on the buffer areas of the system. This feature allows the use of subroutines to generate frequently used patterns.

The CGP is used in conjunction with the SEL 816A Computer Graphics System to provide the most efficient method of transferring data from the 810B Computer to the display unit. Using the CGP minimizes the amount of computer memory and transfer time required to support the display. The CGP also provides a high degree of flexibility in display format generation since the CGP contains the capability of executing the following instruction: (1) Branch Unconditionally, (2) Store Place and Branch, and (3) Stop. The first two instructions (which have the same execution capabilities as the corresponding computer instructions) enable the contents of noncontigous memory areas to be transferred automatically to the display. This capability enables display programs to be organized to provide maximum usage of closed subroutines that are stored in memory a single time and used as often as required in a given display format. The stop command enables the display unit to automatically control the refresh rate, and maintain a fixed rate regardless of the amount of data being displayed.

INPUT OUTPUT PROCESSOR

Figure 4-2 illustrates a more detailed block diagram of the SEL 810B I/O structure. It shows the connection of both the I/O structure to the computer and the peripheral units to the I/O Bus. Three of the five devices shown have additional connections to BTC units. However, all five devices shown and additional devices up to a total of 64 can be commanded by and can communicate with the computer under single-word program control. The additional, fully-buffered transfer capabilities of the units connected to the BTC units are described in the Block Transfer Control Unit section.

The I/OP provides a positive synchronization control for data flow between the computer and peripheral units. It can synchronize data transfer between a peripheral unit and either memory or the A-Accumulator. The data path for each word or character transferred is controlled by a program executing an input/output instruction.

The basic, automatic execution sequence for all I/O instructions consists of three steps:

1. Connect the device specified by the instruction to the I/O bus.

2. Execute the transfer directly between the device and the A-Accumulator or memory.

3. Disconnect the device from the I/O bus.

Three very significant features of this execution sequence are:

1. The device is always specified by the I/O instruction.

2. The device is always connected to and disconnected from the computer by the execution of the instruction.

3. Data transfers are always made directly between the specified device and the computer with no intermediate buffering.

The result of these three features is that the computer I/O structure is always available for use without testing. It is never "busy", except during the times that I/O instructions are being executed. No channel testing or selection is ever required. In addition, no unit selection instructions are required, since each I/O instruction causes the unit specified by the instruction to be selected for transfer.

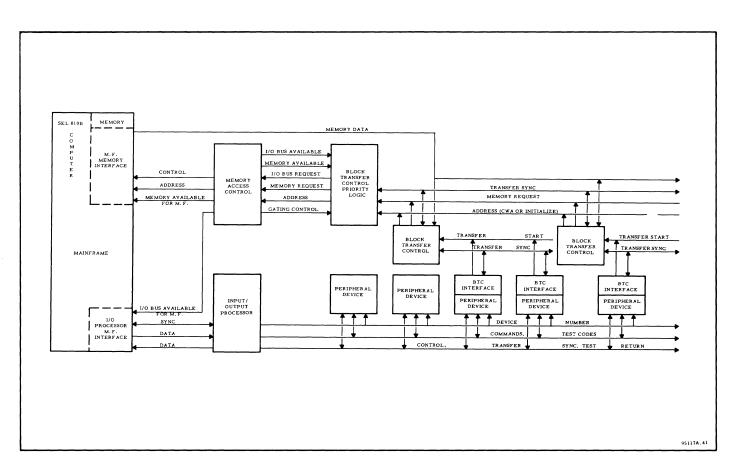


Figure 4-2. Input/Output Configuration and Computer Interface

The I/O Bus connects all peripheral devices to the I/OP in a "daisy-chained" manner, as shown in figure 4-3. The I/O Bus contains 16 data lines, six unit number lines, and numerous control lines.

The 16 data lines provide two-way communication paths. All data, CEU command words and TEU test words are transferred over these lines. Wordoriented units such as acquisition subsystems contain a full set of 16 cable drivers and terminators for the data lines. Character-oriented devices having character assembly buffers such as magnetic tape control devices also contain a full set of cable drivers and terminators. Character-oriented devices having character buffers such as paper tape punches and readers contain only eight to ten cable drivers and/or terminators. In this case, data commands and test codes are always received from the computer on the eight lines corresponding to computer bit positions 0-7. Some units also receive commands from bits 8 to 15. Single characters are always transferred to the computer on the data lines corresponding to bit positions 8-15. Characters having less than eight bits are rightjustified in the eight-bit field. The data lines connected to each peripheral unit are defined in Section VI.

The six unit number lines connected to each unit permit up to 64 individual units to be addressed by the computer.

The control lines consist of the signals named in table 4-1.

These lines are used to enable I/O instructions to be executed in the following basic sequence (TEU differs).

(1) The computer initiates execution by sending out the device (unit) number contained in the instruction. The computer also sends out the Instruction Sync and instruction command (Data, Command, Test, Input/Output) signals.

(2) The addressed device responds by sending the Unit Sync Return and Unit Test Return signals to the computer.

(3) After recognizing the Unit Sync Return signal, the computer tests the Unit Test Return signal for the unit status ("Ready" to execute command or "Not Ready").

Table 4-1. I/O Control Signals

Signal	Computer Commands
Instruction Sync	ALL I/O INSTRUCTIONS
Data Instruction	AIP, AOP, MIP, MOP
Command Instruc- tion	CEU
Test Instruction	TEU
Input/Output	AIP, AOP, MIP, MOP
Wait Flag	AIP, AOP, MIP, MOP, CEU
Unit Test Return	ALL I/O INSTRUCTIONS
Unit Sync Return	ALLI/O INSTRUCTIONS
Computer Data Here	AOP, MOP, CEU, TEU
Computer Data Accepted	AIP, MIP
Unit Data Accepted Computer Clock	AOP, MOP, CEU, TEU
Master Clear	

(4) If the device indicates "Ready", the data transfer is made. The Data Here and Data Accepted signals synchronize the transfer. For computer input transfer, the unit "Ready" signal also indicates "Data Here".

(5) After the transfer is completed, the computer tests the control lines from the device to insure that they have returned to the "Off" level. The next instruction is started in the following machine cycle.

The normal execution time for each immediate mode I/O instruction is four machine cycles. In addition, presence of the Wait Flag in an I/O instruction delays completion of instruction execution until the device indicates a "Ready". (The operation of the Wait Flag is described in Section II.)

The execution sequence is similar for all instructions except TEU. When a TEU instruction is executed, no "Ready" test is made before transfer of the test word. Transfer is made following recognition of the Unit Sync Return signal. The Test Return line is tested after the test word has been transferred to the device. The return signal is a particular unit status gated on the Test Return line by the value of the test word transferred to the device.

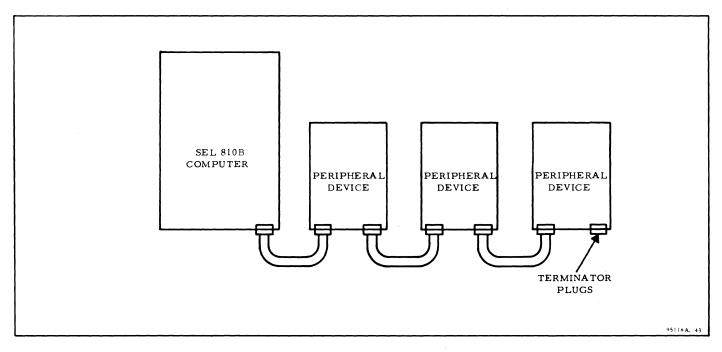


Figure 4-3. Peripheral Device Bus Connections

INPUT/OUTPUT BUS

The Basic 810B Computer is supplied with cable drivers and terminators which enable 16 units to be connected to the computer on one "daisy-chained" cable. Additional drivers, terminators and connectors are available if more than one cable chain is required.

BLOCK TRANSFER CONTROL UNIT

GENERAL CAPABILITIES

The SEL Block Transfer Control (BTC) unit is an optional computer input/output control unit which enables fully-buffered transfer of data between peripheral units and computer memory. The salient features of this unit are listed below:

Bits per Transfer	Full computer word
Maximum Words per Block	32,767
Maximum Transfer Rate	l,333,000 words per second
Memory Cycles Stolen per Transfer	One
Block Transfer Reinitialization	Automatic or program controlled
Maximum Number of BTC's per Computer	8
Maximum Number of CGP's per Computer	6
Maximum Number of Peripheral Devices per BTC	16

BTC OPERATION

The BTC contains two binary counters plus transfer initialization and synchronization logic. One of the counters stores the current word address (CWA) and the second stores the word count (WC). CWA defines the storage location for each word transferred to/from memory and WC defines the number of words to be transferred. The initial values for CWA and WC are obtained from two fixed locations in computer memory by the BTC each time a new block transfer is initiated (see table 5-2 for BTC memory location assignments). Each time a word is transferred between memory and the selected peripheral unit, CWA is incremented and WC is decremented. The block transfer is completed and an interrupt is generated when WC = 0. After a block transfer is completed, the BTC automatically initiates a new block transfer by obtaining a new initial set of CWA and WC values from the two dedicated memory locations. The block transfer sequence is ended by placing a terminate code in the WC word. The terminate code is a ONE in bit 0 (sign bit).

NOTE

The initial value of CWA is identified as the First Word Address (FWA). This allows the BTC starting address to be defined and distinguished from any other CWA.

The CWA value may be transferred from the BTC to the dedicated memory location by execution of a CEU command addressed to the device and containing a ONE in bit 13 of the command code. (See Appendix C.)

BTC Initialization and Data Flow

The BTC is initialized through the peripheral device to/from which the block transfer is to be made. Figure 4-2 shows the data and control path involved. The figure shows two peripheral devices connected to one BTC and a third peripheral device connected to a second BTC. These devices, as previously described, may communicate with the computer through execution of any of the I/O instructions. In addition, they may transfer data under BTC control, rather than under single-word program control.

Execution of the proper Command External Unit (CEU) instruction causes the device specified by the instruction to send an Initialize signal to the BTC to which it is cabled. In many peripheral devices, this instruction also causes the unit to initiate action to produce/accept data. When the BTC receives the Initialize signal from the device, it requests a memory cycle through the Memory Access Control (MAC). It also generates the address of the CWA memory location assigned to it. When the memory cycle is granted, the CWA value is transferred from the memory to the CWA counter in the BTC. A request for a second cycle is then made by the BTC and the address of the memory location containing WC is placed on the address lines by the BTC. When the second cycle occurs, WC is transferred from memory to the WC counter in the BTC. The terminate bit (bit 0) contained in the WC word is also tested and a latch is set if the terminate bit is a ONE, which signifies that no more block transfers are to be made after completion of the one being initialized. The maximum time for the entire initialization is four cycles for

the CEU execution, plus 2 cycles for the CWA and WC transfers, which occur immediately following CEU execution.

After BTC initialization, words are transferred between the selected peripheral device and memory over the I/O data lines under the joint control of the BTC, the BTC Priority Control and the MAC. A word transfer is initiated by the device which sends a Data Transfer Request line to the BTC. The Data Transfer Request signal causes the BTC to request a memory cycle through the MAC. When the MAC determines that the next cycle can be granted, a Memory Available signal is sent to the BTC. The BTC, in turn, sends a signal to the peripheral device which causes it to connect to the Unit I/O data lines, execute the data transfer, and then disconnect from the data lines. After completion of a word transfer, the CWA value is incremented and the WC value is decremented in the BTC counters. All words are transferred by repetition of this cycle, which is always initiated by the peripheral device.

When the value of WC is decremented to zero, the block transfer is terminated. If the terminate latch in the BTC had not been set by the terminate bit in the last WC word acquired from memory, a new block transfer is automatically initiated by the BTC. Re-initialization consists of acquiring new CWA and WC values from the memory locations assigned to the BTC. After re-initialization, an interrupt is generated which signifies that the transfer of the last block is completed and a new block transfer is initialized. The interrupt processing routine can then store in the dedicated locations the CWA and WC values for the next block transfer anytime prior to the completion of the current block transfer. This re-initialization technique reduces the problem of re-initializing block transfers under program control between the times of occurrence of two successive words in a continuous data stream.

If the terminate latch in the BTC had been set by the terminate bit in the last WC word acquired from memory, an interrupt is generated when the value of WC is decremented to zero and no new transfer is initialized by the BTC. In addition, the Data Transfer Request line from the peripheral signal is received. Hence, the BTC disconnects from the peripheral device.

BTC Priority and Timing

BTC's are granted memory cycle requests on a priority basis. The priority ordering function is performed by the BTC Priority Control. A unique priority is assigned to each BTC. The priority logic is structured similar to that of the interrupt priority logic, insuring that higher priority BTC's are always serviced before lower priority units. However, once a word transfer is initiated, it is not interrupted by a request from a higher priority BTC. In addition, BTC requests for memory cycles always take precedence over mainframe requests and can effectively "lock out" the mainframe if the peripheral transfer rate is high enough. BTC and P.I. Assignments are shown in table 5-3.

The maximum collective transfer rate for a BTC (or group of BTC's) is 1,333,000 words per second. Cycle stealing (or lockout) from the program is automatic and each BTC word transferred removes one cycle from the program. The BTC can gain access to the memory after a delay of one cycle except during the time of execution of the instructions listed in table 4-2 (the number of cycles refers to the number of consecutive cycles during which time the BTC cannot gain memory access). When these instructions are executed, the main program will hold out the BTC transfer for a maximum of the number of cycles indicated.

Table 4-2. Execution Times

Two Cycles	Four Cycle	s (or More)
IMS	CEU	МОР
	AOP	MIP
	AIP	TEU

SECTION V PRIORITY INTERRUPT SYSTEM

GENERAL DESCRIPTION

The SEL 810B Computer can have up to 98 individual levels of priority interrupts. Each level can be selectively enabled and disabled under program control except for two special interrupts (Power Fail Safe and Stall Alarm). Two standard levels plus the special power fail safe level are supplied with the basic computer. Additional, optional levels are available in groups of 12 levels each, except the first optional group contains ten levels.

Assignment of interrupts is highly flexible. Internal signals such as Overflow and Memory Parity can be connected to interrupt levels. BTC, End of Block signals and external signals from peripheral units and custom system components are connected to the levels which best fit the operation of each system.

Two special interrupts, Power Fail Safe/Auto Start and Stall Alarm, are provided. These levels are always enabled. Interrupt signals at these levels override all other computer functions, including Halt, I/O Wait and indirect address chaining. The Stall Alarm and Auto Start features are options described in Section VII.

A unique location in memory is assigned to each interrupt level. These locations are assigned in MAP 1 to keep the entire MAP 0 available for program usage. Location 1,002g is assigned to the highest priority programmable level, location 1,003g to the second highest level, etc. Tables 5-1 and 5-2 shows the assignment of interrupt locations and BTC locations.

When an interrupt signal is recognized by the mainframe, a wired-in instruction SPB*L (Store Place and Branch Indirect) is executed, where L is the address of the memory location assigned to the interrupt level. By storing the starting location of the interrupt processing routine in L, a linkage is provided to any point in memory. Since the address of the next instruction to have been executed in the interrupted program is stored in the interrupt routine entry point by the SPB instruction, a means for returning to the point of interrupt is provided. If the Program Protect and Instruction Trap option is included (and the computer is in the protected mode), the status of the Protect Latch is also stored in bit 0 of the interrupt routine entry point by the SPB instruction. When the program returns to the point of interrupt, the protect latch returns to the status present at the time of the interrupt.

The mainframe may be interrupted by a particular interrupt level provided that:

- (a) the level has been previously enabled
- (b) no higher level interrupt is active.

If a higher level interrupt is active when an interrupt signal occurs, the interrupt will be stored until the completion of execution of the higher level interrupt processing routine. The lower level routine will then be initiated. It will continue until completed or until interrupted by a higher level interrupt signal. In this case, the lower level routine will be completed after completion of the higher level routine. Program control will then be returned to the original point of interrupt. The priority logic enables any number of interrupt levels to be active at the same time. Routine execution is always performed in the order of priority of the active interrupts.

Table	5-1.	Priority	Interrupt	Assignments
-------	------	----------	-----------	-------------

(Octal) Memory Location	Interrupt Assignment
1000	Power Fail Safe/Restore
1001	Stall Alarm
1002	PI Group, Level 0,1 (Highest)
1003	0,2
1004	0,3
1005	0,4
1006	0,5
1007	0,6
1010	0,7
1011	0,8
1012	0,9
1013	0,10
1014	0,11 (Std)
1015	0,12 (Std)
1022	1,1

Table 5-1. Priority Interrupt Assignments (Cont'd)

(Octal) Memory

InterruptAssignment 4,3 4,4 4,5 4,6 4,7 4,8 4,9 4,10 4,11 4,12 5,1 5,2 5,3 5,4 5,5 5,6 5,7 5,8 5,9 5,10 5,11 5,12 6,1 6,2 6,3 6,4 6,5 6,6 6,7 6,8 6,9 6,10 6,11 6,12 7,1 7,2 7,3 7,4

Octal) Memory		errupt		Location
Location		ignment		1124
1023		1,2	· .	1125
1024		1,3		1126
1025		1,4		1127
1026		1,5		1130
1027		1,6		1131
1030		1,7		1132
1031		1,8		1133
1032		1,9		1134
1033		1,10		1135
1034		1,11		1142
1035		1,12		1143
1042		2,1		1144
1043		2,2		1145
1044		2,3		1146
1045		2,4		1147
1046		2,5		1150
1047		2,6		1151
1050		2,7		1152
1051		2,8		1153
1052		2,9		1154
1053		2,10		1155
1054		2,11		1162
1055		2,12		1163
1102 PI C	Froup, Level	3,1		1164
1103	1	3,2		1165
1104		3,3		1166
1105		3,4		1167
1106		3,5		1170
1107		3,6		1171
1110		3,7		1172
1111		3,8		1173
1112		3,9		1 174
1113		3,10		1175
1114		3,11		1202
1115		3,12		1203
1122	[4,1		1204
1123		4,2		1205

Table 5-1. Priority Interrupt Assignments (Cont'd)

(Octal) Memory Location	Interrupt Assignment
1206	7,5
1207	7,6
1210	7,7
1211	7,8
1212	7,9
1213	7,10
1214	7,11
1215	7,12

Table 5-2.	BTC	Memory	Location	Assignments
------------	-----	--------	----------	-------------

Memory Location	BTC or Interrupt Assignment		
1060	BTC 1	FWA	
1061	1	WD CNT	
1062	2	FWA	
1063	2	WD CNT	
1064	3	FWA	
1065	3	WD CNT	
1066	4	FWA	
1067	4	WD CNT	
1070	5	FWA	
1071	5	WD CNT	
1072	6	FWA	
1073	6	WD CNT	
1074	7	FWA	
1075	7	WD CNT	
1076	8	FWA	
1077	8	WD CNT	

DETAILED DESCRIPTION

The following paragraphs describe the priority interrupt system hardware and software in detail.

INTERRUPT CONNECTIONS

Two levels of interrupts are supplied with the basic computer. The priority levels of these interrupts are Group 0, Levels 11 and 12. The levels of the standard interrupts are placed at the bottom of Group 0 to make optional levels available of both higher and lower priorities. The two standard levels are connected through the I/O cable to all peripheral units, as required. Interrupt signals in each unit are connected to one or the other of the two levels under program control.

Connection of a unit interrupt to a standard priority level is performed by execution of the CEU instruction. The format of the second word of the CEU instruction for the peripheral units contain three bits used to connect and disconnect the standard unit interrupts. The interpretation of the three bits is given in table 5-3.

Table 5-3. Standard Interrupt CEU Bit Functions

Bit	Function
1 = ONE	Connect Levels Designated in Bits 2 and 3
2 = ONE	Connect/Disconnect Level 11, Group 0
3 = ONE	Connect/Disconnect Level 12, Group 0
1 = ZERO	Disconnect Levels Designated in Bits 2 and 3
2 = ZERO	Leave Level 11, Group 0 in Present State
3 = ZERO	Leave Level 12, Group 0 in Present State
Refer to Appen	ıdix C

INTERRUPT ENABLING / DISABLING

Interrupt levels can be selectively enabled and disabled, one group (up to 12 levels) at a time, by the two instructions:

- PIE (Priority Interrupt Enable)
- PID (Priority Interrupt Disable)

The second word of these two-word instructions has a three-bit group field and a 12-bit level field which designate the group and the one to twelve levels within the group to be affected by the instruction. The bit assignment is shown below:

0	- Bini Grou	ary p. Ng.	12	п	Unita 10	n	Level	7	6	5	4	ذ	2	1
0	1	3	4											15

The group field is binarily coded, group 00 being the highest priority group. The level field is unitarily coded, a ONE in bit 15 is signifying the highest priority level within a group (level 1). An instruction which will cause the five highest levels in group TWO to be enabled is written in assembly language as:

> PIE DATA '20037

Execution of this instruction leaves the seven lower levels within group TWO (if present) unaffected. They remain either enabled or disabled.

INTERRUPT LEVEL LOGIC

The logic for each interrupt level consists of three latches and associated gates. The latches are designated Request (R), Enable (E) and Active (A). The R latch is set by the external interrupt signal. It provides storage for the signal until the level becomes active and the interrupt routine is executed. The R latch is reset by execution of the two instructions which normally terminate each interrupt routine:

TOI	Turn Off Inter	rupt	
BRU*	Unconditional	Branch	Indirect

The R latch is also reset for all designated levels each time the PID instruction is executed.

The E latch is set by the proper group and level bits in the PIE instruction. It is reset by the same bit pattern in the PID instruction.

The logic expression for the signal (In) which interrupts the mainframe from interrupt level n is

 $In = Rn En \overline{(A_1 + A_2 + --- + An-1)}$

where A_1 in the "active" signal for the highest interrupt level. The mainframe tests for the presence of any I signal after fetching each instruction from memory, but prior to execution. If In is present, An is set and the instruction

 $SPB * 1001_8 + n (n < 46),$

is executed in two machine cycle times.

The A latch is reset by the signal generated by the execution of the TOI, BRU* instruction pair. Each time this instruction pair is executed, both the highest level A and R latches are reset, releasing the channel to accept a new interrupt signal.

A special interrupt card is available which provides an alternate means of effectively disabling interrupts. When this card is inserted at any interrupt level, the level can be made active under program control by execution of a PIE instruction specifying that level. Making the special level active inhibits any lower level from becoming active. Therefore, the lower levels are effectively disabled. Execution of the PID instruction addressing the special level causes the active (A) latch for that level to be reset. If the special interrupt card is inserted at the highest level (Group 0, Level 1), a means is provided for keeping higher priority levels from becoming active while lower priority levels are being processed. An interrupt processing routine at any level can be executed without interruption by having a PIE 1,0 immediately after the entry location and a PID 1,0 immediately before the exit (BRU*) location.

INTERRUPT ROUTINE PROGRAMMING

The Assembly language coding for the typical interrupt routine is shown in table 5-4.

Loc.	Operation	Address	Comments			
INT4	ZZZ	***	Storage Loca- tion for Re- turn Address			
	(Interrupt	(Interrupt Processing Routine)				
	TOI		Turn off Inter- rupt			
	BRU*	INT4	Return to Point of Inter- rupt			

Since an interrupt can occur at any time during the execution of the program, the interrupt routine has no method of determining which registers are being used by the interrupted program. Therefore, the interrupt routine must save and restore any registers which it uses, in order to allow the interrupted program to continue upon return: Example:

Loc.	Operation	Address	Comments
INT4	ZZZ	**	
	STA	ASAV	SAVE A-Acc
i	STB	BSAV	SAVE B-Acc
	(Interrupt	Processing	g Routine)
	LAA	ASAV	Restore A-Acc
	LBA	BSAV	Restore B -Acc
	TOI		х.
	BRU*	INT4	

There are five instructions that do not allow an interrupt to be serviced until the next instruction is executed:

TOI	(Turn off Interrupt)
PID	(Priority Interrupt Disable)
PIE	(Priority Interrupt Enable)
SPB	(Store Place and Branch)
CSB	(Copy Sign of B-Acc.)

The TOI instruction inhibits servicing the interrupt for one instruction to allow the BRU* to be executed at the exit of the interrupt routines. This is to insure that the proper active latch (A) is reset.

The PID is included to allow the execution of a PIE immediately following the PID in order to reset any possible requests on the selected levels before enabling them. The inclusion of the PID also permits the disabling of higher level interrupts as the first instruction of an interrupt routine. However, this should be used with caution, because the PID instruction resets the request latches (R) on its second cycle. The duration of an interrupt signal must be greater than the longest non-interruptable instruction chain to guarantee that the interrupt is not erased when a PID instruction is executed. If this condition cannot be met, the special interrupt card (described later) is recommended.

When it is desired to use the same subroutine at more than one interrupt level, it must be allowed to complete the execution for the lower levels or be made re-entrant. The following example shows a method to allow the lower levels to complete.

SUBR	ZZZ PID DATA	** XYZ	Disable all higher levels
(Sul	proutine)		
	PIE DATA BRU*	XYZ SUBR	Enable same levels

The fact that the SPB instruction, used to enter the subroutine, and the PIE, used immediately preceding the BRU* are non-interruptable, allows the routine to complete before a possible re-entry. The SPB instruction being non-interruptable also allows the coding of re-entrant subroutines, in that the exit location can be saved before an interrupt (which can also use the routine) can occur.

The CSB instruction is non-interruptable to allow the execution of an AMA, SMA, or NEG immediately following the CSB. This is necessary because the CSB instruction must always be followed by an AMA, SMA, or NEG instruction because the carry latch is reset at the completion of execution of all instructions except CSB.

When multiple interrupt signals are connected to the same interrupt level, a certain amount of testing is necessary to determine the source of the interrupt. The following example in figure 5-1 shows the use of two typewriters on the same standard output interrupt level.

The first portion of the coding is executed at a lower priority level.

				73 80 IDENTIFICATION
LOC.	OPER.	ADDRESS, INDEX		
1	6	11 25 T	50 I	72
+ + + + + + + + + + + + + + + + + + +	PI FI	<u>↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ </u>	· · · · · · · · · · · · · · · · · · · 	
┝᠇᠇᠇┤	D AIT A	',4,0,0,0, , , , , , , , , , , , , , , ,	┡╺╋╺╋┙┥┙┙┙┙	
+ + + + + + + + + + + + + + + + + + +	LAA	INTRI IIIISETITUPIIINTERRUPTIIOCATI	<u> </u>	
┝᠇᠇᠇ᠯ	SITA	┨ ╅ _╋ ╹ _╋ ╹ _╋ ╹ _╋ ⁵ ╋╋╋╋╋╋╋╋╋╋╋╋╋╋╋╋╋╋╋╋╋╋╋╋╋╋	╷ ╸╸╸╸╷┥╸╸╞╺╎╺╺╺╺╺╺╸	
€~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~		┹ ╊╉┍╗╔╗╗	╷ ╻╺┫╺╋╶┨╺╋╺╋╺╋╺╋╸╋╸╋╸╋╸╋╸╋╸╋╸╋╸╋╸╋╸╋╸╋╸╋╸╋╸╋╸╋╸	┺ ┞┽┽┾┾┾┾┾┾┾┾┾┾┝┝
	LAA	C _I N _I T _I I + + + T _I E _I S _I T + F _I O _I R + M _I E _I S _I S _A G _I E + I _I N + P _I R _I O	,G _I R _I E _I SS +FI-FI-FI-FI-FI-FI-FI-FI-FI-FI-FI-FI-FI-F	<mark>┟┽╁╌╅╼╡┽┲╶╋╌╋╌╋╶╋╺╋╸</mark>
++++	SAZ		╷ ┟┟┎┎┊┧╎┥┝╎╎┥┥┥┥╸	╎ ╴┼╶╁╶╁╶╁╶╁╶╁╶╁╶┨╶ ┥
	BIRIU	*-12+1-1+1-1LIAISIT-ICHARACTERINOT-ISTAR	╷ [┯] ᠇ ╒ ╷ <mark>┙╷╴╷╴╷╶╷╶╷╶╷╴╷╴╷</mark>	+ + + + + + + + + + + + + + + + + + +
	LIAIA	= ⁻ ¹ ¹ ⁰ + + + + 1 ³ ^E I ^T + ¹ C ^H A ^R A ^R C ^T E ^R + 1 ^C O ^U H ^N I ^T + ¹ + + ^T O	<u> ⁻ ¹ ⁰ </u>	
┝╍╺┙	SITA	$C_1 N_1 T_1 1_1 + + + + + + + + + + + + + + + + +$		
┝᠇᠇᠇┤	LAA	$\mathbf{M}_{\mathbf{I}}\mathbf{E}_{\mathbf{I}}\mathbf{S}_{\mathbf{I}}\mathbf{I}_{\mathbf{I}} + \mathbf{I}_{\mathbf{I}} + \mathbf{I}_{\mathbf{I}}\mathbf{S}_{\mathbf{I}}\mathbf{E}_{\mathbf{I}}\mathbf{T}_{\mathbf{I}}^{-} + \mathbf{U}_{\mathbf{I}}\mathbf{P}_{\mathbf{I}} + \mathbf{L}_{\mathbf{I}}\mathbf{A}_{\mathbf{I}}\mathbf{S}_{\mathbf{I}}\mathbf{T}_{\mathbf{I}} + \mathbf{A}_{\mathbf{I}}\mathbf{D}_{\mathbf{I}}\mathbf{D}_{\mathbf{I}}\mathbf{R}_{\mathbf{I}}\mathbf{E}_{\mathbf{I}}\mathbf{S}_{\mathbf{I}}\mathbf{S}_{\mathbf{I}}^{+} + \mathbf{I}_{\mathbf{I}} + \mathbf{I}_{\mathbf{I}}$	+ ^N + ^D + ^E + ^X + ^E + ^D + + + + + + + + + + + + + + + + + + +	┝╶╪╶╪╶┽╶┽╶┽╶┽╶┽╶┽╶┽
	S _I T _I A	I 4 A W I 4 + + + I 1 N T 0 + T Y P E W R I 1 T E R I 1 + 0 U T P U T	· · · · · · · · · · · · · · · · · · · 	· • • • • • • • • • • • • • • • • • • •
┝᠇᠇᠇┤	CIEU	TIP11, IW, I + ICONNECT + ITYPEWRIITER I I III	_I E _I R _I R _I U _I P _I T _I I I I I I I I I I I I I I I I I I I	┝╶┽╶┼╶┽╶┽╶┼╶┽╶┽╶┽
┟┽┽┽╽		<u> </u> , + ⁵ + ⁰ + ⁰ + ⁰ + ¹ + + + + + + + + + + + + + + + + + + +	╸╸╸╸╸╸╸╸╸╸	· • • • • • • • • • • • • • • • • • • •

⁹⁵¹¹⁸A.51

Figure 5-1. Sample Program for Two Typewriters on the Same Standard Output Interrupt Level

			73 80
LOC.	OPER.	ADDRESS, INDEX	IDENTIFICATION
	6	11 25 50	72
			* * * * * * * * * * * * * * *
	LAA	C ₁ N ₁ T ₁ Z ₁ T ₁ E ₁ S ₁ T ₁ F ₁ O ₁ R ₁ M ₁ E ₁ S ₁ S ₁ A ₁ G ₁ E ₁ T ₁ N ₁ P ₁ R ₁ O ₂ G ₁ R ₁ E ₁ S ₁ S ₁ + + + + + + + + + + + + + + + + + + +	• • • • • • • • • • • • • • • • • • •
	SAZ	0, N, T,Y,PIE,W,R,I,T,E,R,2	***
	B _I R _I U	*,-,2, , , , , , , , , , , , , , , , , ,	
	LAA	=1-12101 -1-12001 -1-12001	
	STA		* * * * * * * * * * * * * * * *
	LAA	MES12, I I SET - UPI LAST, ADDRES15+1, I DEXED I I I I I DEXED I I I I I I I DEXED I I I I I I I I I I I I I I I I I I I	****
<u> </u>	STA	$I_{1}A_{1}W_{1}2_{1} + I_{1}N_{1}T_{1}O_{1} + T_{1}Y_{1}P_{1}E_{1}W_{1}R_{1}T_{1}E_{1}R_{1} + 2_{1} + OU_{1}T_{1}P_{1}U_{1}T_{1} + I_{1} + I$	++-++++++++++++++++++++++++++++++++++++
	CEU	T ₁ P ₁ 2, W + CONNECT TYPEWRITER 2 + INTERRUPT	+++++++++++++++++++++++++++++++++++++++
	DATA		+++++++++++++++++++++++++++++++++++++++
T	$\frac{1}{1}$	1	+++++++++++++++++++++++++++++++++++++++
			* * * * * * * * * * * * * * *
	LAA		****
++	SAZ		****
	BRU		****
	LAA	$=_{1}{1}3_{1}0_{1}+_{1}+_{1}S_{1}E_{1}T_{1}+_{1}C_{1}H_{1}A_{1}C_{1}T_{1}E_{1}R_{1}+_{1}C_{1}O_{1}U_{1}N_{1}T_{1}+_{1}1+_{1}T_{1}O_{1}+_{1}{1}3_{1}0_{1}+_{1}+_{1}+_{1}+_{1}+_{1}+_{1}+_{1}+$	+++++++++++++++++++++++++++++++++++++++
	S,T,A		+++++++++++++++++++++++++++++++++++++++
-++-	LAA	MEISI31 + + + SET + -1UP + LAST + APPERESS++1+1+1VPEXED + + + + + + + + + +	*****
+++	S T A		+++++++++++++++++++++++++++++++++++++++
-++-	C ^{EI} UI	<u>T</u> PH1++W++++CPHV+NECT++TYPEWR+ITER++1++I++FR+R+UPT+++++++++++++++++++++++++++++++++	*********
-++‡		<u> </u> + ⁵ + ⁰ + ⁰ + ⁰ + ¹ +++++++++++++++++++++++	+++++++++++++++++++++++++++++++++++++++
++ [7+++7	*************************************	* * * * * * * * * * * * * * *
-++-	++++	<u></u> <mark>╶╶╴╸╸╴╴╸╴╴╴╴╴╴</mark>	+++++++++++++++++++++++++++++++++++++++
NIT	z _I z _I z	*** + + + + + + + + + + + + + + + + + +	+++++++++++++++++++++++++++++++++++++++
-+-+-	SITIA	$\overset{A_{1}s_{l^{A}}}{\overset{A_{1}s_{l^{A}}}{\overset{A_{1}}}{\overset{A_{1}}{\overset{A_{1}}{\overset{A_{1}}}{\overset{A_{1}}{\overset{A}}}}}}}}}}}}}}}}}}}}}}}}}}}}}}}}$	+ + + + + + + + + + + + + + + + + + + +
+++	S,T,B	^B I ^S ^A I ^V +++++++++++++++++++++++++++++++++++	********
+++	LIAIA	$c_{i} \mathbf{N}_{i} \mathbf{T}_{1}^{1} \mathbf{I}_{1} \mathbf{I}_{1} \mathbf{I}_{1} \mathbf{I}_{1} \mathbf{I}_{1} \mathbf{E}_{1} \mathbf{S}_{i} \mathbf{F}_{1} \mathbf{I}_{1}^{1} \mathbf{Y}_{1} \mathbf{E}_{i} \mathbf{W}_{i} \mathbf{R}_{i} \mathbf{I}_{i}^{1} \mathbf{T}_{i} \mathbf{E}_{i} \mathbf{R}_{i} \mathbf{I}_{i}^{1} \mathbf{I}_{i} \mathbf{F}_{i} \mathbf{Q}_{i} \mathbf{R}_{i} \mathbf{I}_{i} \mathbf{Q}_{i} \mathbf{R}_{i} \mathbf{I}_{i} \mathbf{X}_{i} \mathbf{E}_{i} \mathbf{I}_{i} \mathbf{I}_{i} \mathbf{V}_{i} \mathbf{E}_{i} \mathbf{I}_{i} \mathbf{I}_{$	+++++++++++++++++++++++++++++++++++++++
-++-	S _I A _I Z _I	<u>┥┶┲┼┲┲┲┲┲┲┲┲┲┲┲┲┲┲┲┲┲</u>	+++++++++++++++++++++++++++++++++++++++
-+-+-	BIRIUI	$\frac{T_{1}Y_{1}P_{1}I_{1}+I_{1}+I_{1}Y_{1}P_{1}E_{1}W_{1}P_{1}I_{1}T_{1}E_{1}P_{1}+I_{1}+I_{1}P_{1}C_{1}T_{1}I_{1}V_{1}E_{1}+I_$	+++++++++++++++++++++++++++++++++++++++
¥+2+ +		$\frac{ c_iN_iT_i ^2_{i-1-i-1-i}}{ c_iN_iT_i ^2_{i-1-i-1-i}} + \frac{ T_i ^2_{i-1}}{ T_i ^2_{i-1}} + \frac{ T_i ^2_{i-1}}{ T_i ^2_{i-1-i-1}} + \frac{ T_i ^2_{i-1-i-1}}{ T_i ^2_{i-1-i-1-i-1}} + T_i ^2_{i-1-i-1-i-1-i-1-i-1-i-1-i-1-i-1-i-1-i-1$	+++++++++++++++++++++++++++++++++++++++
<u></u>	SIAIZ	<u>+++++++++++++++++++++++++++++++++++++</u>	
+++	BRU		*****
X ₁ I ₁ T	4 <u>4</u> 4	$ \underbrace{A_{1}S_{A}V}_{I} \underbrace{A_{1}I}_{I} \underbrace{A_{1}E_{1}S_{1}T_{1}O_{I}}_{I} \underbrace{A_{1}C_{1}C_{1}U_{1}}_{I} \underbrace{A_{1}T_{1}O_{I}}_{I} \underbrace{A_{1}I_{1}}_{I} \underbrace{O_{1}A_{1}}_{I} \underbrace{A_{1}I_{1}}_{I} \underbrace{O_{1}A_{1}}_{I} \underbrace{A_{1}I_{1}}_{I} \underbrace{O_{1}A_{1}}_{I} \underbrace{A_{1}I_{1}}_{I} \underbrace{O_{1}A_{1}}_{I} \underbrace{A_{1}I_{1}}_{I} \underbrace{O_{1}A_{1}}_{I} \underbrace{A_{1}I_{1}}_{I} \underbrace{O_{1}A_{1}}_{I} \underbrace{A_{1}I_{1}}_{I} \underbrace{A_{1}I} \underbrace{A_{1}I_{1}}_{I} \underbrace{A_{1}I} \underbrace{A_{1}I} \underbrace{A_{1}I} \underbrace{A_{1}}_{I} \underbrace{A_{1}I} \underbrace{A_{1}}_{I} \underbrace{A_{1}}_{I} \underbrace{A_{1}}_{I} \underbrace{A_{1}}_{I} \underbrace{A_{1}}_{I} \underbrace{A_{1}} A_$	+++++++++++++++++++++++++++++++++++++++
┝╃╋╋	L _I B _I A _I	<u> </u> ₿ ₁ \$	+++++++++++++++++++++++++++++++++++++++
┝╋╋╋╋	TIOII	<u>+++++++++++++++++++++++++++++++++++++</u>	+ + + + + + + + + + + + + + + + + + + +
┝╺╸╸┥	B _I R _I U _I *	<u>11NIT + + + + + + + + + + + + + + + + + + +</u>	+++++++++++++++++++++++++++++++++++++++
Y _P 1	TIAIBI		+++++++++++++++++++++++++++++++++++++++
	MOP!*	TIPI1 + + + + TIRY + PUTIPIUT + TO + TYPE, WR ITER + 1+ + + + + + + + + + + + + + + + + +	+++++++++++++++++++++++++++++++++++++++
AW1	z _i z _i zi	***	* * * * * * * * * * * * * *
┝╺╄╺╕╸╋	B _I R _I U		****
+++	I _M S	$ \underbrace{ [C_{N_{1}}T_{1}]_{1}}_{1} + \underbrace{ A_{1}C_{1}C_{1}E_{1}P_{1}T_{1}E_{1}P_{1}}_{1} + \underbrace{ A_{1}C_{1}C_{1}P_{1}E_{1}P_{1}}_{1} + \underbrace{ A_{1}C_{1}C_{1}E_{1}P_{1}}_{1} + \underbrace{ A_{1}C_{1}C_{1}E_{1}P_{1}}_{1} + \underbrace{ A_{1}C_{1}C_{1}P_{1}P_{1}}_{1} + \underbrace{ A_{1}C_{1}C_{1}P_{1}}_{1} + \underbrace{ A_{1}C_{1}C_{1}}_{1} + \underbrace$	+++++++++++++++++++++++++++++++++++++++
┝╃┾╂	BRU		+++++++++++++++++++++++++++++++++++++++
┝╃┲╋	CIEIN	$\frac{1}{2} \left[\frac{1}{2} \left$	* * * * * * * * * * * * * *
┝┹╋╋	DIALTIA	<u>+</u> +++++++++++++++++++++++++++++++++++	+ + + + + + + + + + + + + + + + + + + +
++++	BRU		

Figure 5-1. Sample Program for Two Typewriters on the Same Standard Output Interrupt Level (Cont'd)

				73 80 IDENTIFICATION
LOC.	OPER.	ADDRESS, INDEX	50	72
	MO PI	, Τ ₁ Ρ ₁ 2 ₁ , <u>1</u> , <u>1</u> Τ ₁ R ₁ Y, <u>1</u> Ο ₁ U]T, <u>1</u> Τ ₁ Ο ₁ , <u>1</u> Τ ₁ Y ₁ P ₁ E ₁ W ₁ R ₁ I ₁ T ₁ E ₁ R		
I A W 2	zzz	* ₁ * ₁ +	+1 1 1 N D E X E D 1 1 1 1 1	+++++++++++++++++++++++++++++++++++++++
++++	BRU	EXIT T F BUSY, DIDID NOT CAUSE INTE		+++++++++++++++++++++++++++++++++++++++
	IMIS	CINTI2 I I ACCEPTEDI I IINCREMENT ICHAR	ACTER COUNT 2	+++++++++++++++++++++++++++++++++++++++
	BRU		╷ ╷╷╻╻╷	$\frac{1}{1}$
 ++++	CIEIUI	$\mathbf{T}_{\mathbf{I}} \mathbf{P}_{\mathbf{I}^2}$, $\mathbf{W}_{\mathbf{I}}$, $\mathbf{I}_{\mathbf{I}}$, $\mathbf{C}_{\mathbf{I}} \mathbf{O}_{\mathbf{I}}$, $\mathbf{V}_{\mathbf{I}}$, $\mathbf{T}_{\mathbf{I}}$, $\mathbf{E}_{\mathbf{I}} \mathbf{E}_{\mathbf{I}} \mathbf{P}_{\mathbf{I}}$, $\mathbf{O}_{\mathbf{I}}$, $\mathbf{I}_{\mathbf{I}}$, $\mathbf{D}_{\mathbf{I}}$, $\mathbf{E}_{\mathbf{I}}$, $\mathbf{C}_{\mathbf{I}} \mathbf{O}_{\mathbf{I}}$, $\mathbf{U}_{\mathbf{I}}$, $\mathbf{E}_{\mathbf{I}}$, $\mathbf{C}_{\mathbf{I}}$, $\mathbf{U}_{\mathbf{I}}$, $\mathbf{U}_{\mathbf{I}$, $\mathbf{U}_{\mathbf{I}}$, $\mathbf{U}_{\mathbf{I}$, $\mathbf{U}_{\mathbf{I}}$, \mathbf{U}	<u>, 1^NITIERRRUPET </u>	╶╶╶╶╶╶╶╶╶╶╶╶╶╶╶╶╶╶╶╶╶
 ++++	DIAITIA	 '+ ¹ + ⁰ + ⁰ + ⁰ + ⁰ + ¹	*******	+++++++++++++++++++++++++++++++++++++++
-	B _I R ₁ U ₁	E _I X ₁ I ₁ T ₁ + E _I X ₁ I ₁ T ₁ + I + I + I + I + I + I + I + I + I +		
I NITIR	D AICI	IINDI + + + + IINITERRRUPT + ROUTIINE ENTRY	+ ^A +D+D+R+E+S+S++++++++++++++++++++++++++++	+++++++++++++++++++++++++++++++++++++++
C _{INT1} 1	z _i z _i z _i	*I*I I I I I I I I I I I I I I I I I I	PEWR + + + + + + + + + + + + + + + + + + +	*********
Силитт	Z1 Z1Z1	<mark></mark> * <mark>ŧ*<mark>ŧ-ŧ-ŧ-ŧ-ŧ-ŧ</mark>СŀĦ<mark>ŀAŀŖŀAŀC</mark>ĬŢŀĘŀŖŧ-ŧĊŀOŀŨ_INIŢŀĘŀŖŧ-ŀFŀOŀŖŧ-ŧŢŀŶ</mark>	PEWRITER + 121 - 1 - 1 - 1 - 1	******
A _I S _I A _I V	Z ₁ Z ₁ Z ₁	*+*+++++++++**************************	* * * * * * * * <u>* * * * * * *</u> * *	*****
B _I S _I A _I V	z _i z _i z _i	* ₁ * ₁ +	· · · · · · · · · · · · · · · · · · · 	**************
M _{IE} SI	D _I A _I C _I	M ₁ S ₁ ¹ + ¹ + ¹ + ⁰ + ¹	* * * * * * * + + + + + + + + + + + + +	╆╺╋╺╋╺╋╺╋╺╋╺╋╺╋╺╋╺╋╺╋
M _I EISI2	D _I A _I C _I	M ¹ S ¹ ² ⁺ ⁺ ² ⁰ ¹ ⁺ ¹ + ⁺ +	 	+ • • • • • • • • • • • • • • • • • • •
M _I EISI3	DIAC	M ¹ ^S ¹ ³ ⁺ ¹ ³ ¹ ⁰ ¹	• • • • • • • • • • • • • • • • • • • 	****
M ₁ S ₁ 1	DIAITIA	<mark>╜_╏┻╷┺┟С╷<mark>┛╒┺╒</mark>╒╔┟[┲]╷┰_┛╷╜╷╸┥╴┝╴╸╸╸╸╸╸╸╸╸╸╸╸╸╸╸╸</mark>	*****	*****
M ₁ S ₁ 2	DIAITIA	[,] [,] ^A ^{,B} ^{,C} ^{,D} ^{,E} ^{,F} ^{,G} ^{,H} ^{,I} ^{,J} ^{,I} ^{,K} ^{,L} ^{,M} ^{,N} ^{,O} ^{,P} ^{,Q} ^{,R} ^{,S} ^{,T} ^{,"} + + + + + + + + + + + + + + + + + + +	* * * * * * * * * * * * * * * * * * *	* * * * * * * * * * * * * * *
M4S131	DIAITIA	<mark>᠃᠂ᠰ᠋ᡖᡰᡄᡃ᠋ᡋ᠇᠋ᢄᡃᡄ᠖ᡃᡆ᠋᠋᠇᠋᠋᠋ᠴ᠂ᢘᡰ᠋᠋᠇ᢂᢆᡅᡃᠣ᠋ᡉᡃᡦᡃᡦᡃᡌᡃᡈ᠈ᡓ᠋᠋᠋᠇᠒ᡰ᠕ᢥᢘᡰᠷᠯ᠋ᡘᡰ᠋᠋᠋ᢌᡃ᠋᠇</mark>	· · · · · · · · · · · · · · · · · · · 	+++++++++++++++++++++++++++++++++++++++
++++	++++	}····	, , , , , , , , , , , , , , , , , , , 	+++++++++++++++++++++++++++++++++++++++
++++	++++	 • • • • • • • • • • • • • • • • • • •	+ + + + + + + + + + + + + + + + + + + 	+++++++++++++++++++++++++++++++++++++++
++++	++++	$\begin{array}{c} \begin{array}{c} \begin{array}{c} \begin{array}{c} \end{array} \end{array}$	* * * * * * * * * * * * * * * * * * * *	****
++++	++++	 • • • • • • • • • • • • • • • • • • •	+ + + + + + + + + + + + + + + + + + + 	+++++++++++++++++++++++++++++++++++++++
L		L	• • • • • • • • • • • • • • • • • • •	

95118A, 51

Figure 5-1. Sample Program For Two Typewriters on the Same Standard Output Interrupt Level

(Cont'd)

SECTION VI PERIPHERAL DEVICES

INTRODUCTION

This section contains brief descriptions of the standard peripheral devices available with the SEL 800 series of computers. Although part of each description consists of physical characteristics, the primary emphasis is on the programming aspects of the equipment.

CONSOLE TYPEWRITER (MODEL NO. 81-711-02A Device No. 1)

The standard console typewriter (Model 81-711-02A) provided with the 810B Computer is a modified Teletype Model ASR-33 send/receive typewriter. The modifications introduced by Systems Engineering Laboratories include the interface and control electronics that make the basic device more flexible when used with the computer.

The unit responds to the following commands:

- a. Select Reader Mode
- b. Select Keyboard Mode
- c. Reader and Keyboard Off

The Model 81-711-02A consists of a typewriter keyboard and printer plus an eight-level paper tape punch and reader. It is supplied as standard equipment with 810B Computers and is mounted on a stand adjacent to the computer. A manual switch is provided to enable either on-line or off-line operation. Specifications for the Model 81-711-02A are defined in table 6-1.

Table 6-1.	Model 81-711-02A Console	Typewriter
Specifications		

Characteristics	Specification
Paper Tape Input Speed	20 characters per second
Output Speed	10 characters per second for print and punch
Code	ASCII

Table 6-1. Model 81-711-02A Console Typewriter Specifications (Cont'd)

Characteristics	Specification
Number of Printable Characters	63
Characters per Line	72
Paper Tape	Standard 1-inch roll
Dimensions	22 inches wide x 35 inches high x 18 inches deep including stand
Power	115 VAC, 60 cps, single phase. dc voltages required by the interface and con- trol electronics are supplied by the com- puter power supplies.

ASR-33 PROGRAMMING

When the ASR-33 is operated on-line, the input devices (keyboard and reader) operate separately from the output devices (printer and punch). As a result, the paper tape reader can operate independently at the rate of 20 characters/second. If printing and punching of input characters are desired, each character read into the computer is transferred back to the output devices under program control. Both printing and punching occur unless the punch is turned off manually. Printing and punching can be performed at rates up to 10 characters/second.

The ASR-33 coupler contains two character buffers, one for input and one for output. The presence of two buffers plus the separation of the input and output devices enables both input and output data transfers to occur, both at the maximum rate. For example, a paper tape can be read at 20 characters/ second while an independent set of characters is printed at 10 characters/second. The presence of the buffers in the ASR-33 also results in the computer and I/O bus being "busy" for only four machine cycles eachtime a character is transferred between the computer and the ASR-33.

The two standard peripheral unit interrupt levels can be connected in the associated buffer by program control. Each time the input buffer, which stores characters from the keyboard and reader, receives a new character it generates an interrupt at Group 0, Level 11. Each time the output buffer, which stores characters for printing and punching, is emptied it generates an interrupt at Group 0, Level 12.

The CEU commands sent to the ASR-33 select the desired input mode and connect/disconnect the two interrupts. The single output mode is selected by the on-line switch and requires no CEU command execution. As a result, any time an AOP or MOP instruction having a unit 1 address is executed, a character will be printed. The character will also be punched if the punch is turned on.

Bit coding for the CEU second word function codes are shown in table 6-2. Bit numbering corresponds to the positions (0-15) in a computer word.

Function	Bit (=1)
Priority Interrupt Connect	1
Computer Input Interrupt	2
Computer Output Interrupt	3
Reader Mode	4
Keyboard Mode	5
Mode Clear	6

Bits 1-3 permit the two ASR-33 interrupts to be selectively connected to the two standard peripheral unit interrupt levels. The input interrupt, used with keyboard and reader entry, is connected to PI Level 11, Group 0 by execution of the instruction:

CEU	1
DATA	'60000

It is disconnected by execution of the instruction:

CEU	1
DATA	'20000

The corresponding instructions which connect and disconnect the output mode interrupt to PI Level 12, Group 0 are:

Connect	CEU DATA	1 '50000
Disconnect	CEU DATA	1 '10000

Execution of this instruction causes an interrupt to be generated immediately so that the first as well as all succeeding characters can be transferred by the same interrupt processing routine. The connect commands must be preceded by a PIE instruction to enable the computer to be interrupted.

Execution of the input mode select commands, which contain bits 4, 5, or 6 cause the input buffer to be cleared as well as the specified input unit to be selected. In transferring between reader and keyboard modes, it is not necessary to mode clear. This mode clear signal allows turning both modes off.

Reader or keyboard operation is controlled by "select reader mode" or "select keyboard mode" bits in the CEU instruction. The reader control operates on an automatic character call-up philosophy. As a result of each AIP or MIP instruction, a data character is transferred to the computer and action is initiated to start calling up the next character on the tape. This feature simplifies the programming by turning the reader on and off (with CEU instructions) only once for each group of characters desired.

NOTE

Each time a character is read, the tape advances, and the character cannot be read again. The programmer must insure that a GEU to clear the reader mode ("select keyboard mode", or "select mode clear") and turn the reader off is generated within 10 ms after each desired group of characters is accepted by the computer. Otherwise, the next character on the tape will be read, stored in the input data buffer, and subsequently lost when the buffer is cleared or loaded with new data.

No test unit (TEU instruction) hardware is required on the ASR-33 coupler. A test for "busy" is built in each instruction. It can become desirable to perform this test without attempting to command the ASR-33 coupler. To accomplish this, an I/O instruction to unit one, with no bits set in the second word, will skip when not busy without changing the unit.

For the most part, the input section and the output section operate completely independently under one unit number. For reference, the busy times are approximately as follows:

Reader	50 milliseconds*
Keyboard	100 milliseconds
Output Devices	100 milliseconds

*First 10-12 milliseconds of each cycle left open to allow turn-off.

The reader control switch is not used for on-line operation and can be left in either the Start or the Neutral position while running.

Useful Keyboard Features are as follows:

- a. Here is Key Punches all zeros. Can be used to generate leader or spaces for off-line operation. Recommended way of pulling tape into punch. Will not input into computer.
- b. Rub Out Key Punches all ones.
- c. Repeat Key Repeats characters as long as a character key and the repeat key are depressed.
- d. Line Feed, Carriage Return Keys Self explanatory.
- e. Control Key Used to generate special functions (such as WRU, BELL and TAB) which appear on the top portion of the character keys. The control (CTRL) key must be held down to transmit these functions when depressing the character keys.

NOTE

The bell is a convenient, audible alarm that can be generated by the program.

The correspondence between computer bit positions and paper tape levels is shown in the data flow diagram (figure 6-1).

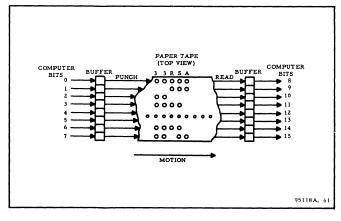


Figure 6-1. Paper Tape Data Flow Diagram

A programming routine for keyboard input and printer output is shown in table 6-3. This routine inputs, packs and stores a pair of characters. It also prints the characters as they are typed in.

Operation	Address	Comments
CEU	1	Select Keyboard
DATA	'2000	
BRU	*-2	Not Ready
AIP	1	Input Character
BRU	*-1	Not Ready
LSL	8	Shift to Output
AOP	1	Print Keyed Char- acter
BRU	*-1	Not Ready
AIP	1, R	Input Character, Merge
BRU	*-1	Not Ready
STA	WORD	
LSL	8	
АОР	1	Print Keyed Char- acter
BRU	*-1	Not Ready

Table 6-3.	Programm	ing Routi	ne for	Console	Key-
Boar	d Input and	Console	Printer	• Output	

NOTE

This routine is interruptable because no wait flags are used.

PAPER TAPE READER (MODEL NO. 81-510A, 300 CPS- DEVICE NO. 2)

The SEL Model 81-510A series paper tape reader reads eight-level one-inch paper or mylar tapes at synchronous speeds of up to three hundred characters per second in the forward direction. The paper tape reader responds to the following commands:

- a. Reader Enable
- b. Reader Disable

Specifications for the Model 81-510A Paper Tape Reader are defined in table 6-4.

Characteristic	Specifications
Speed	300 characters per second
Levels	8
Operation	Asynchronous start/stop
Size	19-inches wide x 7-inches high x 24-inches deep
Sensing	Photocell
Drive	Pinch roller
Power	115 volts, 60 cps ± 10%, 1.4 amp. nominal
Temperature	10° C to 35° C
Controls	Power on/off Run/load

Table 6-4. Model 81-510A Paper Tape Reader Specifications

PAPER TAPE PUNCH (MODEL NO. 81-520A, 110 CPS- DEVICE NO. 2)

The SEL Model 81-520A Paper Tape Punch punches eight-level paper or mylar tapes at speeds up to 110 tape characters per second. A sprocket hole is punched with each character. Punch power may be turned on or off under computer program control.

The unit responds to the following commands:

- a. Turn punch, tape feed power on
- b. Turn punch, tape feed power off

Specifications for the Model 81-520A Paper Tape Punch are defined in table 6-5.

Table 6-5. Model 81-520A Paper Tape Punch Specifications

Characteristic	Specifications
Speed	110 characters per second
Levels	8
Operation	Asynchronous start/stop

Table 6-5.	Model	81-520A	Paper	Tape	Punch
Spe	ecificat	ions (Cor	ntinued)	

Characteristic	Specifications
Size	19-inches wide x 14-inches high x 24-inches deep
Punches	High-carbon steel pins
Drive	Sprocket
Power	115 volts, 60 cps ± 10%, 2 amps. nominal
Temperature	10°C to 35°C
Controls	Power on/off and tape feed

Other features include built-in tape storage reel, tape cutter and transparent chad box.

PERFORATED PAPER TAPE SPOOLER (MODEL NO. 80-530A)

The SEL Model 80-530A Paper Tape Spooler supplies and spools 5, 6, 7 or 8 level paper tapes at rates up to 400 characters per second. The tape is rewound by a manually controlled switch at a 1000 characters per second rate.

No command or test functions are required from the computer. Specifications for the Model 80-530A paper tape spooler are defined in table 6-6.

Table 6-6.	Model	80 - 530A	Paper	Tape	Spooler
	S_{I}	pecificati	ons		

Characteristic	Specifications		
Feed Speed	40 inches per second		
Rewind S peed	100 inches per second		
Tape Channels	5, 6, 7 or 8 levels, inter- changeable		
Reel Hubs	Standard NAB reel dimen- sions		
Reel Diameter	8-inch O. D.		
Reel Capacity	400 feet of 4-mil tape		
Interlock	Tape break or no tape		
Mounting	Any upright position		
Size	19-inches wide x 10-1/2 inches high x 8-5/8-inchess deep		

Characteristic	Specifications		
Power	115VAC ± 10%, 60 cps, single phase		
Temperature	5° C to 45° C		
Controls	Power on/off Rewind		

Table 6-6. Model 80-530A Paper Tape Spooler Specifications (Cont'd)

HIGH SPEED PAPER TAPE PUNCH/READER SYSTEM (MODEL NO. 81-525A-DEVICE NO. 2)

The paper tape system consists of a high-speed photoelectric reader and a high-speed punch. The photoelectric reader is capable of reading 6, 7, or 8-level paper tape at rates of 300 characters per second. The punch punches 8-level tape at a maximum rate of 110 characters per second.

Since both the punch and the reader are given the same unit number, they can be considered, from the software standpoint, as one unit with both input and output capabilities. Since they each have a separate buffer, they can be operated at maximum speed simultaneously.

There are two CEU commands that can be given the paper tape reader. These commands are "Reader Enable" and "Reader Disable". When the reader enable command is issued, the buffer will be filled with a character and the tape will advance one character position. This will occur only if the buffer is initially clear. While the reader is enabled, the tape will advance one position and the buffer will be filled each time an AIP or MIP is serviced by the reader. When the reader disable command has been issued, the reader will not advance when an AIP or MIP is serviced.

There are two CEU commands that can be given to the punch. They are "Punch Feed Power ON" and "Punch Feed Power OFF". The punch feed power on command must be given before transferring data to the punch. After punch tape feed power has been turned on, the punch will punch the tape and advance one character position each time an AOP or MOP is serviced.

There are two standard interrupts furnished with the Systems Engineering Laboratories high-speed paper tape system. One is the "Buffer Ready" interrupt from the punch and the other is the "Buffer Ready" interrupt from the reader. A routine for copying paper tape is given in table 6-7.

Table 6-7. Programming Routine for Copying Paper Tape

Operation	Address	Comments
CEU	2, W	
DATA	'5000	Punched Feed Power On/ Reader Enable
AIP	2, W	Input Character from Reader
LSL	8	
AOP	2, W	Output Character to Punch
BRU	*-3	

MAGNETIC TAPE (MODEL NO. 80-615 SERIES- DEVICE NO. 6 AND 7)

The SEL Model 80-615A magnetic tape transports use 1/2-inch mylar tape and have either 7 or 9 tracks. Specifications for the transports are given in table 6-8. The 7-track units are IBM 729 compatible and the 9-track units are IBM 2400 compatible. The tape control unit (unit 6 or 7) acts as a coupler for up to eight magnetic transports. Any standard SEL magnetic tape unit can be coupled to the computer by the tape control unit (TCU).

The control panel for the TCU contains the following indicators:

- a. CRC Error (option on 9-track system only)
- b. Ready
- c. Parity Error (Lat/Long)
- d. Read/Write Status
- e. Busy
- f. End-of-Tape
- g. Load Point
- h. Density (200/556/800)
- i. Binary Mode

Start or Minimum Gap Spanning Time *+ Model Tape Speed Stop Time* 9 Track 7 Track 615-07 45 9 msec. 25.4msec. 21.7 msec. ips 16.4 14.2 msec. 615-09 75 6 msec. ips msec. 615-11 120 3.8 msec. 10.2 msec. 8.9 msec. ips 615-12 9.2 8.1 msec. 150 ips 3.5 msec. msec.

Table 6-8. Model 80-615 Magnetic Tape Transport Specifications

*Nominally +20%

+Time between the writing of the last character in one record to the writing of the first character in the next record.

- j. BCD Mode
- k. EOF
- 1. Characters/Word
- m. Transport Number Selected

The control panel for the TCU also contains a threeposition switch that allows the characters in the Write, Read or LRCC registers to be displayed.

MAGNETIC TAPE PROGRAMMING

CEU Instruction

In order to allow optimum programming of the magnetic tape system, a quasi-unitary bit assignment is used for the CEU data words. In order to represent all of the functions, two data word formats, called FORMAT 0 and FORMAT 1, are used. The desired format is selected by placing a 0 or a 1 in bit position 4 of the data word.

The quasi-unitary bit assignment allows more than one set-up or command function to be executed with one CEU instruction provided the selected bits are logical and do not contradict each other.

Format 0, CEU

The first tape operation command must be a CEU instruction using the format 0 data word which includes the set-up bits to select BCD or Binary tape density (200, 556, 800), transport number (0-7) and characters per word. These set-up bits must be entered in every CEU FORMAT 0 data word. Control bits are also provided to enable or disable one or both selectable interrupts, to rewind the tape, to erase four inches of tape, and to input the current word address of the BTC.

Table 6-9 lists the bit functions for the second word of the format 0, CEU command.

Table 6-9. CEU, Format 0, Second Word, Bit Functions

Bits	Function
0	Must be ZERO.
1 = ONE	Connects the interrupt selected with bits 2 and 3. If bits 2 and 3 are ZERO,bit 1 is ignored.
l = ZERO	Disconnects the interrupt selected with bits 2 and 3. If bits 2 and 3 are ZERO, bit 1 is ignored.
2 = ONE	Selects word transfer ready inter- rupt for enable or disable.
3 = ONE	Selects end of record interrupt for enable or disable.
4 = ZERO	Selects the FORMAT 0 CEU data word.
*5 = ONE	Rewinds selected tape transport. Bits 10-12 must contain desired transport number.
*6 = ONE	Erases four inches of tape.
7 = ONE	Sets-up the TCU to transfer BCD data with even parity.
7 = ZERO	Sets-up the TCU to transfer Binary data with odd parity.

Bits	Function
8 & 9	Selects tape density 00 2 - 200 bpi 01 ₂ - 556 bpi 10 ₂ - 800 bpi
10-12	Selects tape transport (0-7)
13 = ONE	Inputs current word address of BTC into the corresponding de- dicated locations.
14 & 15	Selects characters per word $01_2 - 1$ char/word $10_2 - 2$ char/word $11_2 - 3$ char/word $00_2 - 4$ char/word
*Bits 5	5 and 6 cannot both be ONE in the

Table 6-9. CEU, Format 0, Second Word, Bit Functions (Cont'd)

*Bits 5 and 6 cannot both be ONE in the same CEU Format 0 instruction (only logical bit combinations should be selected).

Tape Format

Figure 6-2 shows the magnetic tape format for a format 0 data word.

Format 1, CEU

The Format 1 CEU data word is completely unitary. Care must be taken to assure that only logical bit combinations are selected. Table 6-10 lists the bit functions for the second word of the format 1, CEU command.

Table 6-10.	CEU,	Format 1,	${\tt Second}$	Word,
	Bit	Functions		

Bits	Function
0 = ONE	Initializes Block Transfer.(Mustnot be used if Bit 13 is ONE).
1 = ONE	(Same as Format 0)
1 = ZERO	(Same as Format 0)
2 = ONE	(Same as Format 0)
3 = ONE	(Same as Format 0)
4 = ONE	Selects the Format 1 CEU data word.
**5 = ONE	Write Record - the tape will move forward as the information being sent from the computer re- places the previous information on the tape. When this information ceases to come from the computer, a longitudinal check character is written and a record gap is gene- rated.
	In the case of a 9 track system, the CRC character is written 4 character times previous to the LRCC character and follows the last data character by 4 character times.
**6 = ONE	Approximately 3-1/2 inches of tape are erased and an EOF mark is written.

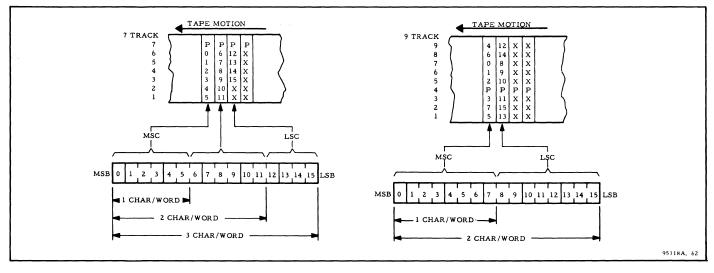


Figure 6-2. Magnetic Tape Format 0 Data Word

Table 6-10.	CEU,	Format 1,	${\tt Second}$	Word,
	Bit Fui	nctions (Co	nt'd)	

Bits	Function	Bits	Function	
**7 = ONE	Read Record - the tape will move forward to the next re- cord gap leaving the data on the	14 & 15	be used if bit 0 is at ONE). Unused	
	tape undisturbed. When the tape is in motion, the data is transferred to the computer as it is encountered on the tape. This transfer will continue to take place until the next record gap is reached or until the com- puter stops requesting the data. If the computer stops request-	four inches operation. if one of th no more da	ese commands should not follow an erase oftape,write record or write end of file This is not a hardware restriction, but ese commands is executed and there is ta on the tape, the transport will run all off of the reel unless it is manually	
	ing the data before the next re- cord gap is encountered, a data overflow condition exists. This condition can be tested. A read operation should not	by bits 5 th single CEU	nly one motion command, represented rough 11 in Format 1 can be used in a data word. Two or more motion com- ne same instruction are contradictory.	
**8 = ONE	follow a write, erase four inches of tape or write EOF operation. * Advances tape forward one re-	"write rec the TCU is	nination of an "erase four inches of tape", ord", or "write end-of-file" operation left in write status. All other opera- eave the TCU in read status.	
MAG - UNE	cord, leaving the read/write heads in the middle of the next record gap. The data on the tape is undisturbed.		ction ction is used to query the status of a With the magnetic tape system all	
**9 = ONE	Advances tape one file, leaving the read/write heads in the middle of the record gap follow- ing the next end of file mark. The data on the tape is undis- turbed. *	status conditions are reset by every motion com- mand issued to the TCU. The transport tested is always the one that is set up at the time the TEU instruction is executed. The TEU codes are set up to skip on the "GO" condition.		
**10 = ONE	Backspaces one record, leaving the read/write heads in the middle of the previous record gap. The data on the tape is undisturbed.	each statu: If more tha a single T being a ''N skipping.	The TEU data word is completely unitary, in that each status condition has a single bit assignment. If more than one status condition is queried with a single TEU instruction, any one status condition being a "NO GO" will inhibit the instruction from skipping. The tests that may be initiated by the TEU instruction are listed in table 6-11.	
**11 = ONE	Backspaces one file, leaving the read/write heads in the gap preceding EOF mark. The data	Table 6	11. TEU, Second Word, Bit Functions	
	on the tape is undisturbed.	Bits	Function	
12	Not Used	0 = ONI	Skip On Not Busy - When the TCU is capable of executing a motion com- mand, the next instruction is skipped If the TCU is not capable of execut- ing a motion command, the next in- struction is executed.	
13 = ONE	Inputs current word address of BTC into the corresponding dedicated location (Must not	l = ONI	Skip On No End-Of-File - When no EOF status is present, the next in- struction is skipped. When an EOF	

Table 6-10. CEU, Format 1, Second Word, Bit Functions (Cont'd)

Table 6-11. TEU, Second Word, Bit Functions (Cont'd)

Bits	Function	Bits	Function
	status is present the next in- struction is executed.	8 = ONE	Skip On Rewinding - If the transport that is selected is mechanically re-
2 = ONE	Skip On No Overflow - When no overflow status is present, the next instruction is skipped. When an overflow status is pre- sent, the next instruction is ex- ecuted. An overflow occurs when the data request from the computer is dropped before an end-of-record gap is reached on the tape. (Occurs only when reading magnetic tape.)		winding, the next instruction is skipped. If it is not rewinding, the next instruction is executed. (When a rewind command is issued to the TCU, the rewind status does not exit until the mechanical motion has started. Therefore, before discon- necting a tape transport which has been given a rewind command, it is necessary to test for the rewind status. Once the mechanical motion has started, the tape transport can
3 = ONE	Skip On Load Point - If the read/ write heads are positioned at load		be deselected and the rewinding will continue while another tape transport is being commanded.)
	point the next instruction is skipp- ed. If the read/write heads are not positioned at load point, the next instruction is executed.	9 = ONE (9-Track only)	Skip On No CRC Error - If no cycle Redundancy Check error exists the next instruction is skipped. If a Cycle Redundancy Check error ex-
4 = ONE	Skip On End-Of-Record Interrupt- If the magnetic tape end-of-record causes an interrupt, the next instruction is skipped. If the magnetic tape end-of-record did not cause an interrupt the next instruction is executed.		ists the next instruction is executed. The 9-track tape system, when writ- ing tape, generates a cycle redundancy character (CRC) and writes it on the tape after each record. When the tape is being read back into the com- puter, the cycle redundancy character
5 = ONE	Skip On No Parity Error - When no parity error status is present the next instruction is skipped. When a parity error status is present, the next instruction is executed.	10-15	is generated again and is compared with the one written on the tape. If the two characters do not compare, a CRC error exists and can be tested for with a TEU instruction. Unused
6 = ONE	Skip On Write Ring In - When the		
	write ring (file protect ring) is in the reel, the next instruction is skipped; when the write ring is not in the reel, the next in- struction is executed. The ab- sence of the write ring prevents the destruction of data on tapes, such as library tapes. The write ring must be in the reel in order to write on the tape.	an optional c enables a ful peripheral us netic tape co with a BTC u out BTC. He with all tape usage or low	ic block transfer control (BTC) unit is omputer input/output control unit which ly-buffered transfer of data between nits and computer memories. All mag- ntrol units are designed to operate unit and over the standard I/O bus with- owever, a BTC unit is normally used control units, except in some low transfer rate applications. The I/O
7 = ONE	Skip On No End-Of-Tape - If the end of tape marker has not been sensed, the next instruction is executed. The tape transports will not stop when the end-of- tape is sensed.	usage of fow transfer rate applications. The 170 handlers in the standard software assume the presence of a BTC unit connected to the TCU. Word Transfer (No BTC) With Magnetic Tape This method of transfer can be used with any magnetic tape unit available with the SEL 810B. When	

Table 6-11. TEU, Second Word, Bit Functions (Cont'd)

using this method of transfer, the data is output by means of an AOP or MOP instruction or input by means of an AIP or MIP. In this mode, endof-record is generated when the data flow to the tape unit ceases. When reading, EOR is sensed in the normal manner.

Care must be taken when outputting to the TCU without the BTC, to insure that the data is available to the TCU in the time required by the speed of the tape transport and the recording density. The same considerations must be made when reading tape or data will be lost.

Interrupts With Magnetic Tape

There are two standard interrupts available with SEL tape transports. One interrupt is an "endof-record" interrupt that occurs when an EOR is written or sensed. The second standard interrupt available is the "Word" interrupt which is used if no BTC is available.

In the condition of output, or writing on the tape, the TCU will interrupt (starting with the second word) anytime its word buffer is ready to receive data. In the condition of input, or reading from the tape, the TCU will interrupt (starting with the first word) anytime its word buffer is ready to send data.

Other interrupts such as EOF interrupt, parity error interrupt, end-of-tape interrupt, and information overflow interrupts are optionally available.

Tape Transport Selector Switches

The tape transport selector switches should not be changed on any of the tape transports unless the transport is in the local mode.

Sample Magnetic Tape Program for BTC

Table 6-12 gives a routine that will read one record from tape and will terminate the BTC with the completion of that transfer since the terminate bit was set in the block length word.

Table 6-12.	Programming Routine for Magnetic
	Tape With BTC

Loc.	Oper.	Address	Comments		
READ	ZZZ	**			
	STA	SAVA	Save A-Accumu- lator		

Table 6-12.	Programming Routine for Magnetic
	Tape With BTC (Cont'd)

Loc.	Oper.	Address	Comments
-	LAA	LOC	
	STA	FWA	Set up First Word Address
	LAA	SIZE	
	STA	ΒL	Set up Block Length
	CEU	6	Set up Tape Deck l
	DATA	'113	Read/556/Binary/ 3 Char. Per Word
	BRU	*-2	
	CEU	6	Initialize BTC/ Enable EOR Intr/ Start Tape Mo- tion
	DATA	'156000	
	BRU	*-2	
	LAA	SAVA	Restore A-Accu- mulator
	BRU*	READ	Exit
SAVA	ZZZ	**	
LOC	DAC	BLOCK	Address of Buffer
SIZE	DATA	'101750	Buffer Size, 1000 Words

HIGH SPEED PRINTER (MODEL NO. 80-700 SERIES- DEVICE NO. 5)

Three models of line printers are available which differ in printing speeds and type printer used.

Model 81-731A - 600 LPM

Model 81-732A - 1000 LPM

Model 81-733A - 300 LPM (Shuttle)

The printers use plain or pre-printed standard perforated multi-part, fan-folded paper stock. Horizontal formatting is computer controlled while vertical format can be determined by either computer commands or a vertical control loop.

The printer responds to the CEU and TEU command and test functions listed in table 6-13.

Table 6-13. CEU and TEU Second Word Bit Format for High Speed Printer

CEU Commands

Bits	Function
9	Fill Buffer or Format Tape Chan- nel # if Bit 4 is a one
8	Clear Buffer
7	Print
5	Paper Advance One Line
4	Paper Advance to Loop Channel N
6	Paper Advance to Top of Form

TEU Commands

Bits	Function
9	Test Printer Inoperable
8	Test Bottom of Form
6	Test Parity Error
4	Test Busy

Specifications for the high-speed line printers are defined in table 6-14.

Table 6-14.	Model 80-700 Series,	High Speed
	Printer Specifications	

Characteristic	Specification
Speed	300, 600 or 1000 lines per minute
Characters per Line	Up to 120
Paper Width	Up to 20 inches
Print Area	Horizontal - 10 Characters/ inch;Vertical - 6 lines/ inch

Table 6-14.	Model 80-70) Series,	High Speed
Print	er Specificat	ions (Con	t'd)

Characteristic	Specification		
Type Face	Open Gothic		
Number of Characters	64		
Code Wheel Sensing	Photocell		
Vertical Form Control	8 Channel		
Power Requirements	115V, 60 cps, single phase		
Size	56-inches wide x 34-inches high x 30-inches deep		
Controls	 a. Power on/off b. Master clear c. Single space d. Top of form e. Index tractors f. Format control g. Parity error 		

HIGH SPEED LINE PRINTER PROGRAMMING

Systems Engineering Laboratories line printers have a 120-character buffer. This buffer must be filled in the order that the characters are to appear in the printed line. If the entire buffer is not to be filled, it must be cleared prior to filling. When a function is issued to clear the buffer, spaces are loaded into the buffer by the printer logic. Since the buffer holds only 120 characters, the printer logic will not reply to an attempt to send the 121st character, and the program will "hang up" if a wait flag is used with the AOP or MOP.

The Systems Engineering Laboratories printer logic is designed to accept simultaneous commands as long as they are not in mechanical or logical conflict. For example, no two mechanical commands such as "Print" and "Advance Paper One Line" can be given at the same time. Also the buffer cannot be commanded to clear and fill at the same time. The third type of conflict that must be avoided is trying to combine the command "Advance to Format N" with anything else, since another command code would make the function ambiguous.

There are three modes for advancing paper on the line printer. One mode is to advance the paper only one line. Another is to advance the paper to the top of form, which is the logical top of page (where the printing is to begin). The third mode is to advance to format N. This can be only one line or as much as a full page, depending on where the next punch is in the channel specified by N. N = 0 corresponds to top of form.

An example of programming the line printer is shown in table 6-15. This routine prints one line. It assumes that the data has been stored one character per word.

Table 6-15.	Programming	Routine	for	High
	Speed Print	ter		

Loc.	Oper.	Address	Comments
	SPB	LINE	
	DAC	FWA	First Word Ad- dress
	DATA	WC	Character Count
LINE	ZZZ	**	Enter
	STA	SAVA	
	STB	SAVB	Save Registers
	LAA*	LINE	
	STA	ADRS	Setup First Word Address
	IMS	LINE	
	LAA*	LINE	
	NEG		
	ТАВ		Set up Negative Character Count
	IMS	LINE	Setup Exit Ad- dress
	CEU	5,W	
	DATA	'2200	Clear Buff/Ad- vance one Line
	CEU	5,W	
	DATA	'100	Fill Buff
	LAA*	ADR S	
	AOP	5,W	Output Charac- ters

Table	6-15.	Programm	ing	Routine	\mathbf{for}	High
	Spe	ed Printer	(Co	nt'd)		

Loc.	Oper.	Address	Comments
	IMS	ADRS	
	IBS		
	BRU	*-4	Character Count not Zero
	CEU	5,W	
	DATA	'400	Print Line
	LAA	SAVA	
	LBA	SAVB	Restore Regis- ters
	BRU*	LINE	Exit
SAVA	ZZZ	**	
SAVB	ZZZ	**	
ADRS	ZZZ	**	

PUNCHED CARD READER (MODEL NO. 81-450A 400 CPM-Device No. 4)

The SEL Model 81-450A Card Reader reads standard 80-column cards at a maximum rate of 400 cards per minute. Reading is column by column. The binary values of all 12 rows are transferred to the computer for each column read. Specifications for the punched card reader are defined in table 6-16.

Table 6-16. Model No. 81-450A Punched Card Reader Specifications

Characteristic	Specification
Speed	400 cards/minute
Read Mechanism	Photoelectric
Hopper Capacity	1000 cards
Dimension	22-inches wide x 52-inches high x 30-inches deep
Power	115 VAC ±10%, 60 cps, ±3 cps
Power	

PUNCHED CARD READER PROGRAMMING

The unit responds to a Feed a Card CEU command. This command also functions as a test. If a unit is ready and no Wait flag is used, the program counter will skip to the next sequential address. If it is not ready, the program executes the next sequential address. If a Wait flag is used, the computer will wait until the unit is ready before it executes the CEU.

Table 6-17 gives a "CARD INPUT" subroutine for reading a card.

Table 6-17.	Programming	Routine	for	Punched
	Card Read	er		

Loc.	Oper.	Address	Comments
CARD	ZZZ	**	Entry
	LAA*	CARD	
	STA	IADD	Store IBUF Add- ress
	LBA	=-80	
	CEU	4, W	Feed A Card
	DATA	'4000	
WDIN	AIP	4, W	Input Character
	STA*	IADD	Store in Input Buffer
	IBS		Increment B (In- dex) and Skip
	BRU	WDIN	Not Last Word
	IMS	CARD	Setup Exit
	BRU*	CARD	Exit to $L + 2$
IADD	ZZZ	**	

This routine is called by the sequence:

L	SPB	CARD
	DAC	IBUF

It stores 80 characters, right-justified into 80 consecutive memory locations starting at location IBUF. Control is returned to L + 2.

X-Y INCREMENTAL PLOTTER (MODEL NO. 81-810A AND 81-812A— DEVICE NO. 11)

The SEL Models 81-810A and 81-812A incremental plotters are high-speed digital, two-axis plotters. The actual plot is produced by the movement of a pen over the surface of the chart paper.

The units responds to the following command and test instructions:

- a. -Y (Drive carriage right)
- b. +Y (Drive carriage left)
- c. -X (Rotate drum up)
- d. +X (Rotate drum down)
- e. -X-Y (Rotate drum up and drive carriage right)
- f. +X-Y (Rotate drum down and carriage
 right)
- g. -X+Y (Rotate drum up and drive carriage left)
- h. +X-Y (Rotate drum down and drive carriage left)
- i. Pen up
- j. Pen down

Specifications for the X-Y Incremental Plotters are defined in table 6-18.

Characteristic	Model 81-810A Specifications	Model 81-812A Specifications
Chart Width	12 inches	31 inches
Speed (X, Y direction)	18,000 steps/ minute 0.005 inches.	12,000 steps/ minute
	12,000 steps/ minute 0.01 inches.	
Pen Up/Down	600 move- ments per minute	600 movements per minute

Table 6-18. Model No. 81-810A and 81-812A X-Y Plotter Specifications

Table 6-18. Model No. 81-810A and 81-812A X-Y Plotter Specifications (Cont'd)

Characteristic	Model 81-810A Specifications	Model 81-812A Specifications
Resolution	0.005 inches (81-810-01A) or 0.01inches (81-810-02A)	0.01 inches
Plot Width	11 inches	29.5 or 11 inches
Chart Drive	Sprocket	Sprocket
Power	115 volts,±10%, 60 cps, 1.6 amps nominal	Same as 81-810A
Temperature	10°C to 35°C	10° to 35°C
Manual Con- trols	Drum forward/ reverse Carriage right/ left Pen up/down Single step and continuous modes Power on/off	

The Y-axis plot is produced by lateral movements of the pen carriage and the X-axis plot by rotary motion of the chart drum. Provisions for Z-axis modulation are also incorporated. The plotter responds to the CEU command functions listed in table 6-19.

The basic movements on each axis (X or Y) are at 0 degrees and 90 degrees, and the X, Y combinations yield 45-degree angles. The remaining two movements are pen up and pen down.

Table 6-19. CEU Second Word Bit Format for X-Y Plotter

Bit	Function	
4	Pen Down	
5	Pen Up	
6	Drum Down	
7	Drum Up	
8	Carriage Left	
9	Carriage Right	

6-14

The Plotter can be made to move manually (offline) by means of knobs on a continuous or on a single step basis.

Under computer control (on-line) only step movements are provided.

X-Y PLOTTER PROGRAMMING

Plotter addressing is handled by means of the CEU instruction. Six bits of the second word of the CEU command are used as control bits. The list of all the possible commands and their bit configurations is shown in table 6-20.

Table 6-20.	X - Y	Plotter	Comman	ds and	Bit
Configu	ratio	n in CE	U Second	Word	

Command	Bit Configuration	Command	Bit Configuration
Pen Up	('2000)	X+ Y+	('1200)
Pen Down	('4000)	X+ Y-	('1100)
X+	('1000)	X-Y-	('500)
Y +	('200)	X-1-	(500)
х-	('400)	37 37 1	(1(00)
Y -	('100)	X - Y +	('600)

Program examples:

CEU	'11,W	Unit 'll Wait
DATA	'2000	Pen Up
CEU	'11	Unit '11
DATA	'2000	Pen Up
BRU	BUSY	Calcomp Busy

Table 6-21 shows a Calcomp visual diagnostic routine. It plots a polygon using all the possible movements including pen up and pen down at a predetermined scale factor. The calling sequence is: SPB CDIA. The A-Accumulator is loaded with the desired scale factor. This program lowers the pen, draws an octagon using the eight movements and raises the pen before halting.

Table 6-21. Diagnostic Routine for X-Y Plotter

Loc.	Oper.	Address	Comments
CDIA	ZZZ		Calcomp Visual Diagnostic

Loc.	Oper.	Address	Comments
	NEG		Set Scale Factor Counter
	STA	HOWN	Save It
	STA	TIMS	
	LBA	MEIG	(-8) No. of Dif- ferent Movements
	CEU	'11,W	Pen Down, Wait
	DATA	'4000	
GO	CEU*	'11,W	
	DAC	MEIG +9,1	
	IMS	TIMS	Completed Sca le Factor
	BRU	GO	No, Output Same Command
	IBS		Yes, Test for all Movements Done
	BRU	*+4	No, Do Next One
	CEU	'11,W	Yes, Bring Pen Up
	DATA	'2000	
	BRU*	CDIA	Exit
	LAA	HOWM	
	STA	TIMS	
	BRU	GO	
HOWM	ZZZ	**	
TIMS	ZZZ	**	
MEIG	DATA	- 8	
	DATA	'200	
	DATA	'1200	
	DATA	'1000	
	DATA	'1100	

Table 6-21. Diagnostic Routine for X-Y Plotter (Cont'd)

Loc.	Oper.	Address	Comments
	DATA	'100	
	DATA	'500	
	DATA	'400	
	DATA	'600	

MOVABLE HEAD DISC STORAGE (MODEL NO. 81-653A-Device No. 13)

The SEL Model 81-653A Disc Storage System (DSS) consists of a Disc Control Unit (DCU) and a Disc Storage Drive.

The DSS is a random access, bulk storage device with a Storage capability of 1,536,000, 16-bit words. The disc is subdivided into tracks, surfaces and sectors. Each recording surface of the disc is accessed by a moveable head. The head can be moved to any of 100 tracks. Each track contains 16 sectors. Figure 6-3 shows the track and sector layout of a recording surface.

The ten recording surfaces of the disc pack are addressed by the moveable head assembly. Each surface is read/written by an individual head. Figures 6-4 and 6-5 show the head arrangement in relation to the recording surfaces.

Each sector will store 96 words, thus each track of a recording surface will store 1,536 words and an entire recording surface will store 153,600 words.

Considering a disc pack as 100 cylinders, 15,360 words can be written/read in each cylinder without moving the head assembly.

The disc rotates at 2400 RPM. This gives a maximum latency time of 25 milliseconds. Figure 6-6 shows the time required to move the head "n" positions. The data transfer rate of the disc system is 78.125 KHz, or one word every 12.8 microseconds.

MOVEABLE HEAD DISC STORAGE PROGRAMMING

The CEU instruction is used to command the DCU. There are two types of CEU second word formats, DISC SEEK and DISC DATA. The disc seek command is used to position the head assembly to the required track. Refer to Appendix C for the CEU second word formats for the disc.

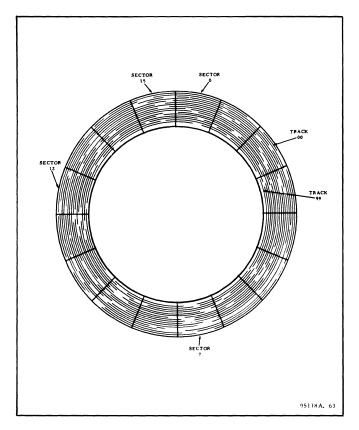


Figure 6-3. Track and Sector Layout

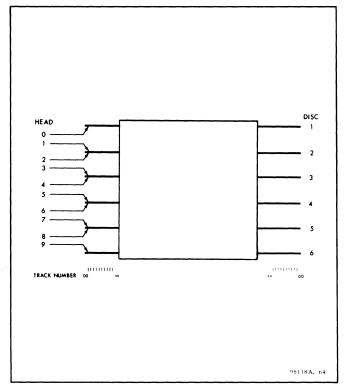


Figure 6-4. Movable Head Arrangement - Recording Surface

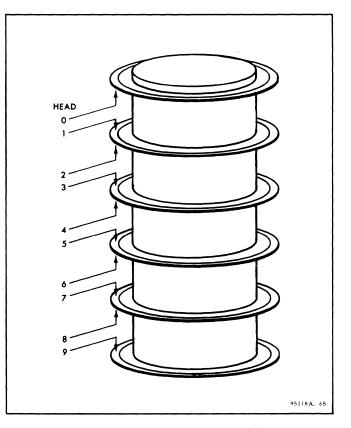


Figure 6-5. Head Position

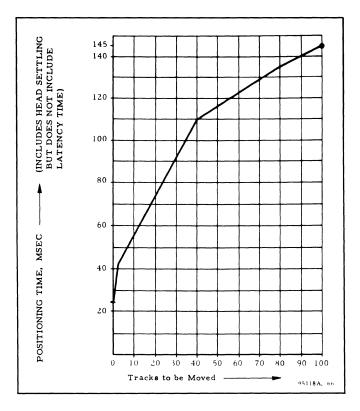


Figure 6-6. Typical Head Positioning Time Chart

The Disc Control Unit accepts a total of five commands from the computer which define all permissable disc operations. These commands are:

a.	Seek Track Zero	
b.	Seek N Tracks Forward	Disc Seek Mode
c.	Seek N Tracks Reverse	
d.	Write Sector I, Head J	Disc Data Mode
e.	Read Sector I, Head J \int	Disc Data Mode

The three seek commands enable the ten physically connected heads to be positioned to any desired track number. The program can keep track of the current head assembly position and command the head assembly to be moved a specified number of tracks in either direction to position the heads to a new track number. The positioning mechanism is quite reliable, but an absolute verification of the new head position can be obtained by recording track and sector identification in one or all sectors per track and reading a sector containing I. D., each time the heads are repositioned.

An alternate method of track accessing consists of sending the heads to track zero after each disc transfer is completed. Use of this technique enables absolute rather than relative track addressing but it does increase the minimum time between successive disc operations.

The read and write commands enable any sector on the ten tracks currently under the disc heads to be written or read.

To seek track 00 (when the current track is unknown) the following CEU is executed:

CEU '13	CEU '13, W
DATA '10	DATA '10
BRU*-2	

Once the head is positioned at any track, motion commands specify the number of tracks to be moved, and the direction of movement (forward or reverse).

For example, assume if the head assembly is at track 50 and the new positions are to be, successively, track 55, track 71, track 38 and track 43. The instructions listed in table 6-22 must be executed to move the head assembly to the required tracks.

Table	6-22.	Movea	ble	Head	Disc	Storage,
	Move	ement (Com	mand	s	

Command	Movement		
CEU '13, W DATA '132	Forward, 5 tracks		
CEU '13, W DATA '412	Forward, 16 tracks		
CEU '13, W DATA '1031	Reverse, 33 tracks		
CEU '13, W DATA '132	Forward, 5 tracks		

Note that in a disc seek command bit 12 of the CEU second word is always a ONE.

Head and sector selection are performed by executing the CEU with the disc data second word format.

For example, to read sector 12, head 5 the CEU instruction would be;

CEU '13, W DATA '6121

and to write sector 7, head 7 the CEU instruction would be;

CEU '13, W DATA '3562.

The two standard interrupts can be connected/disconnected by the execution of the CEU instruction with the appropriate combination of bits 1, 2 and 3 of the CEU second word. The seek error interrupt occurs when a motion command occurs that cannot be executed; for example, the heads are at track 70 and a "forward 70 tracks" command is given. The seek complete interrupt occurs when the heads are at the selected track.

The TEU instruction can be used to test for seek complete, seek error, disc pack on line, read overflow, write overflow, checksum error, DCU ready and unit busy.

Data is transferred between the disc and the computer one word at a time, in not more than 96 words (1 sector) blocks. When using the Block Transfer Control Unit with the disc, the BTC is initialized with bit 0 of the Disc Data CEU. If more than one sector of data is to be read or written, the terminate bit should be set in the word count location so that the interrupt processing routine can handle the bookkeeping functions (sector and head modification, first word address of buffers, etc.).

For non-BTC operation, a data word must be presented to the disc each 12.8 microseconds. Otherwise data will be dropped during the transfer. This transfer rate restricts other operations that can be performed concurrently in the computer. Therefore, all transfers are normally made between disc and computer via BTC units. The I/O disc handler routines assume BTC operation.

FIXED HEAD DISC STORAGE (MODEL NO. 81-654A-Device No. 13)

The Fixed Head Disc Storage Unit (DSU) provides random-access bulk storage of output data from any SEL Series 800 computer. Storage capacity, when used with a SEL 810B Computer, is up to 909, 312 16-bit words. Units with from one to eight recording surfaces are available. There are 64 fixed recording heads per surface (refer to figure 6-7). Each surface contains 64 recording tracks with each track divided into 16 sectors (refer to figure 6-8). Each sector provides storage for 111, sixteen bit data words. Average access time for data recording or retrieval is 8.3 milliseconds. Maximum access time is 17 milliseconds. Word transfer rate to or from the storage unit is 112.5 KHz for 16-bit word storage. The data format is completely under program control and can take any required form. The Disc Control Unit (DCU) regulates data transfer between the DSU and the computer. The DCU also performs all track and sector selections required to store and retrieve data from specified disc locations. A checksum is generated and written at the end of each recorded sector. A checksum is also generated from the data read from a sector and compared to the checksum written at the end of that recorded sector. Additional specifications are listed in tables 6-23 and 6-24.

Table	6-23.	Model	81-654A	Fixed	Head	Disc
	St	orage S	Specificat	ions		

Characteristic	Specification
Speed	3600 RPM, 16.7 milli- seconds/revolution ±S%*
Capacity	Bits - 3,637,248 16-Bit Words - 227,328

Table 6-23.	Model 81-654A Fixed Head Disc
Storage	Specifications (Cont'd)

Characteristic	Specification
Word Rate	112.5 KHz
Bit Rate	1.8 MHz
Number of Tracks	64 per surface 128 per disc
Access Time	Average - 8.3 milli- seconds Maximum 16.7+S* milliseconds
Time Between Sectors	Approximately 37.2 microseconds +S*
Error Detection	Checksum
Interrupts (2)	 Program Error or Checksum Error Read and Write Overflow
Recording Density	1000 BPI
Disc Coating	Nickel-Colbalt Magnetic Plating
Prerequisitor	Connection to Computer BTC unit.
Options	Expansion of storage capacity to 14,548,992 bits by increasing num- ber of discs to four and number of heads to 512.
*S=Induction Motor S 3 to 4%	lippage, approximately

Table 6-24 lists the fixed head disc capacity.

	Bits	Words	Sector	Track	Sur- faces
Per Word	16	-	-	-	-
Per Sector	1776	111	-	-	-
Per Track	28,416	1776	16	-	-

Table 6-24. Model 81-654-128A Disc Storage Capacity Specifications (One Disc)

	Bits	Words	Sector	Track	Sur- faces
Per Sur- face	1,818,624	113,664	1024	64	-
	3,637,248			128	2

Table 6-24. Model 81-654-128A Disc Storage Capacity Specifications (One Disc) (Cont'd)

FIXED HEAD DISC STORAGE PROGRAMMING

TEU and CEU word formats for the 810B are included in tables 6-25 and 6-26. During on-line operation, commands for testing and transferring data to the DCU from the computer are generated by instructions within the computer program. Test function are performed by TEU (Test External Unit) instruction. Data transfer is accomplished by AOP (Accumulator Out to Peripheral), MOP (Memory out to Peripheral), AIP

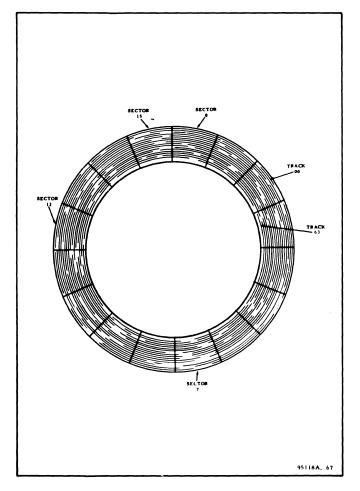


Figure 6-7. Fixed Head Track and Sector Layout

(Accumulator In from Peripheral) or MIP (Memory in from Peripheral).

The DCU accepts a total of five command instructions (CEU).

The DCU accepts a total of seven test instructions (TEU) used to verify the status of the disc storage system. Each of these instructions test circuits in the DCU which have two conditions, set or reset. If the reset state is detected when the particular test is performed, a program skip is initiated. Conversely, if the set state is detected, no skip is initiated.

PRIORITY INTERRUPTS FOR FIXED HEAD DISC

The DCU is equipped with two priority interrupts designated #1 and #2. The interrupt circuits are enabled or disabled by CEU command instructions in the computer program. A single CEU command instruction can enable or disable one or both interrupts.

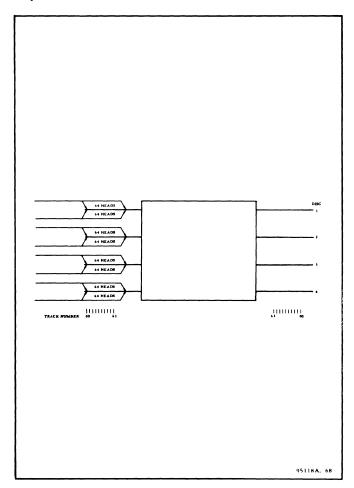


Figure 6-8. Fixed Head Arrangement - Recording Surface

Bits	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Skip On No Program Error	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Skip On Disc On Line	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
S kip On No Disc Read Overflow	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
S kip On No Disc Write Overflow	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
Skip On No Parity Error	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
S kip On No Disc File Area Protected	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
S kip On Disc Controller Not Busy	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0

Table 6-25. Fixed Head Disc TEU Second Word Format

Table 6-26. Fixed Head Disc CEU Second Word Format

Bits	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Select Track	втс	P/I Enable Disable	P/I No. l	P/I No.2	256	128	64	Trac 32	k Nu 16	mbe: 8	r 4	2	1	втс	1	1
Write Sector N	BTC	P/I Enable Disable	P/I No.l	P/I No.2		Not	Used			8	Sec 4	tor 2	1	BTC	Write 1	0
Read Sector N	BTC	P/I Enable Disable	P/I No. l	P/I No.2		Not	Used			8	Sec 4	tor 2	1	BTC	0	Read 1
Write Starting At Sector N Sequential	BTC	P/I Enable Disable	P/I No. l	P/I No.2	Seq. 1		ot Us	ed		Star 8	tin 4	g Se 2	ctor l	BTC	Write 1	0
Read Starting At Sector N Sequential	втс	P/I Enable Disable	P/I No. 1	12/1 No. 2	Seq. 1		ot Us	ed		Star 8	tin 4	g Se 2	ctor 1	BTC	0	Read 1

SECTION VII Options

PROGRAM PROTECT AND INSTRUCTION TRAP (MODEL 81-080B)

When the program protect option is included in an 810B Computer, the memory is divided into 16 areas of 1024 (-1B option) or 2048 (-2B option) words each. A 16-bit protect register is included in the computer which stores the protect status of each memory area. The protect logic causes an interrupt to be generated if an instruction attempts to write into a protected memory area when the computer is operating in the "Unpriviledged" state. Instructions are provided for loading and storing the contents of the protect register.

The status of the program protect mode ("Priviledged" or "Unpriviledged") is maintained by the "protect latch" (PL).

The protect latch operates as follows:

- a. The protect latch is set ON ("Priviledged" state) when the mode key switch is turned from disable to enable. Thereafter, it is turned ON each time a priority interrupt occurs.
- b. When the protect latch is ON, any instruction in a protected memory area can be executed.
- c. Any instruction in an unprotected memory area can also be executed provided that no attempt is made to write into a protected memory area. However, execution of any legal instruction in an unprotected area causes the protect latch to be turned OFF.
- d. When the protect latch is OFF, any attempt to execute an instruction which attempts to write into a protected area will generate a protect violation interrupt, regardless of whether the instruction is stored in a protected or unprotected area. This feature prevents unprotected programs from randomly entering protect programs.
- e. Any priority interrupt will turn ON the protect latch and any instruction within the interrupt subroutine which is not in

a protected area will turn OFF the protect latch. To insure that the protect status that was present at the time of the interrupt is returned after the interrupt subroutine is completed, the protect latch status is stored as follows: after the interrupt has occurred, when the wired SPB instruction is executed. the status of the protect latch is stored in bit 0 of the effective address defined to store the program counter contents. When the TOI and BRU indirect (or LOB) instructions are executed following the interrupt subroutine, the protect latch is returned to the status present at the time the interrupt occurred.

There are two control panel indicators associated with the program protect feature. One indicator displays the status of the protect latch, and the other indicator (located adjacent to the A-Accumulator display indicators) is lit when the protect register is selected for display on the row of indicators that normally displays the A-Accumulator contents.

VARIABLE BASE REGISTER, (MODEL 81-042B)

The variable base register is a 6-bit register which allows any MAP to be used as a reference or base MAP. Whenever the MAP and index bits of an instruction are set to logical zero, the contents of the VBR are treated as the most significant bits of, and appended to, the nine-bit operand address. If the MAP bit is set to a logical ONE, the most significant six bits of the operand address are defined by the most significant six bits of the program counter. If the VBR is set to zero, the 810B oprates just as if no VBR were present.

The contents of the VBR are not appended to memory reference instructions having a ZERO MAP bit when indexing is specified in the instruction (Index Bit = 1). This feature permits relative addressing by indexing to be performed independently of the VBR contents. For example, execution of the instructions,

LBA = '1000

LAA 0,1

causes the contents of '1000 to be loaded into A-Accumulator regardless of the VBR contents.

STALL ALARM (MODEL 81-043B)

The stall alarm is designed to correct, or inform the computer operator of, the stalling or "hanging up" of computer operations. If after 32 machine cycles (0:79 microseconds per cycle), the computer program counter has not advanced, an "Override" interrupt is generated. This override interrupt is capable of interrupting an indirect chain, an I/O instruction, or even a Halt condition. The interrupt routine assigned to this level can take the most suitable corrective action.

AUTO START (MODEL 81-041B

The auto start feature provides the capability of the computer to return to a "run" condition automatically in the event that, after being "lost", power is restored. An "override" interrupt is provided which allows return to the regular program or to a recovery routine.

TABLE TOP (MODEL 81-057B)

This option provides a table-height writing surface mounted on the front of the computer cabinet for the convenience of the operator.

INPUT/OUTPUT PARITY (MODEL 81-210B)

This option is made available for use with special computer interfaces such as serial communication links in which a parity bit is transmitted with each word. This option consists of a means to transfer the parity bit stored in memory with each computer word output transfer, and to check the parity bit accompanying each computer input word transfer.

The operation of the I/O Parity unit is controlled by the external interface unit which must send a request for parity checking/transfer with each input/output word request. If a parity error is detected for an input transfer, the error signal is transferred to the external unit.

INDEX REGISTER (MODEL 81-006B)

The optional index register can be used to perform the same indexing functions as those performed using the B-Accumulator. This additional, programmable register is made available to make the SEL 810B Computer even more powerful in applications involving extensive operand and address manipulation. Two single precision, or one double precision operand can be manipulated in the Aand B-Accumulators and the optional index register can be used for address manipulation without disturbing the contents of the B-Accumulator.

A program controlled flip-flop is supplied with the optional index register which acts as an index register pointer. When this flip-flop is set to one state, the execution of an instruction containing the Index Flag causes the contents of the B-Accumulator to be added to the operand address. When the flip-flop is set to the other state, the presence of an Index Flag in an instruction causes the contents of the optional index register to be added to the operand address.

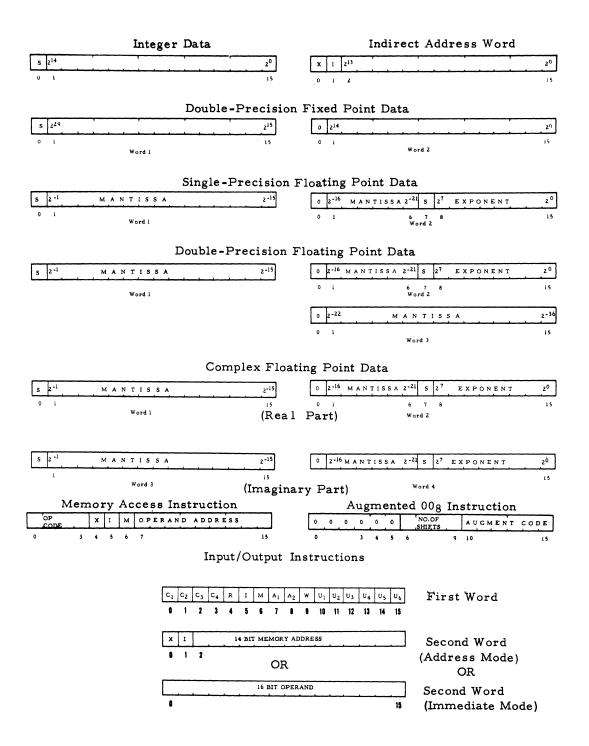
Instructions are provided to change and to test the state of the index pointer flip-flop. Therefore, either the B-Accumulator or the optional index register can be used for indexing in a program, or both can be used in the same program for double indexing operations.

The eight instructions provided with the optional index registers are: LIX, STX, SXB, IXS, TAX, TXA, XPX, and XPB. Refer to the SEL 810B Instruction List Summary and to Section II of this manual for description of these instructions.

60Hz REAL-TIME CLOCK (MODEL 81-031B)

This option provides interrupt signals at the frequency of the ac input power supplied to the computer.

APPENDIX A SEL 810B COMPUTER WORD FORMATS



APPENDIX B SEL PERIPHERAL DEVICE OCTAL CHARACTER CODES ALPHABETIC CHARACTERS

	Teletype	Line Printer		Hollerith	n C a rd Code
Character	ASR-33 & ASR-35	(Truncated ASCII)	IBM/BCD	Octal Code	Card Rows
A	301	01	61	4400	12-1
В	302	02	62	4200	12-2
С	303	03	63	4100	12-3
D	304	04	64	4040	12-4
E	305	05	65	4020	12-5
F	306	06	66	4010	12-6
G	307	07	67	4004	12-7
н	310	10	70	4002	12-8
I	311	11	71	4001	12-9
J	312	12	41	2400	11-1
К	313	13	42	2200	11-2
L	314	14	43	2100	11-3
М	315	15	44	2040	11-4
Ν	316	16	45	2020	11-5
0	317	17	46	2010	11-6
Р	320	20	47	2004	11-7
Q	321	21	50	2002	11-8
R	322	22	51	2001	11-9
S	323	23	22	1200	0 - 2
Т	324	24	23	1100	0 - 3
U	325	25	24	1040	0 - 4
v	326	26	2.5	1020	0 - 5
w	327	27	26	1010	0-6
х	330	30	27	1004	0 - 7
Y	331	31	30	1002	0-8
Z	332	32	31	1001	0-9

95118A, B1

APPENDIX B (CONT'D) SEL PERIPHERAL DEVICE OCTAL CHARACTER CODES

	Teletype	Line Printer			Card Code
Character	ASR-33 & ASR-35	(Truncated ASCII)	IBM/BCD	Octal Code	Card Rows
0	260	60	12	1000	0
1	261	61	01	0400	1
2	262	62	02	0200	2
3	263	63	03	0100	3
4	264	64	04	0040	4
5	265	65	05	0020	5
6	266	66	06	0010	6
7	267	67	07	0004	7
8	270	70	10	0002	8
9	271	71	11	0001	9

NUMERIC CHARACTERS

SPECIAL SYMBOLS OR FUNCTIONS

Symbol Or	Teletype	Line Printer			Card Code
Function	ASR-33 & ASR-35	(Truncated ASCII)	IBM/BCD	Octal Code	Card Rows
@	300	00	57	2006	11-8-7
]	333	33	75	4022	12-8-5
١	334	34	36	1012	0-8-6
]	335	35	55	2024	11-8-5
t	336	36	32	1202	0-8-2
•	337	37	77	4006	12-8-7
Space	240	40	20	-	-
1	241	41	52	3000	11-0
11	242	42	37	1006	0-8-7
#	243	43	35	1022	0-8-5
\$	244	44	53	2102	11-8-3
%	245	45			
&	246	46			

95118**A, B**2

SEL PERIPHERAL DEVICE OCTAL CHARACTER CODES

Symbol Or	Teletype	Line Printer	[Hollerith	Card Code
Function	ASR-33 & ASR-35	(Truncated ASCII)	IBM/BCD	Octal Code	Card Rows
t	247	47	14	0042	8-4
(250	50	34	1042	0-8-4
)	251	51	74	4042	12-8-4
*	252	52	54	2042	11-8-4
+	253	53	60	4000	12
,	254	54	33	1102	0-8-3
_	255	55	40	2000	11
	256	56	73	4102	12-8-3
/	257	57	21	1400	0 - 1
:	272	72	15	0022	8-5
;	273	73	56	2012	11-8-6
<	274	74	76	4012	12-8-6
=	275	75	13	0102	8-3
>	276	76	16	0012	8-6
?	277	77	72	5000	12-0
Carriage Return	215				
Line Feed	212				
Bell	207				
Delete	377				

SPECIAL SYMBOLS OR FUNCTIONS

95118A. B3

		0	1	2*	з*	4	5	6	7	8	9	10	11	12	13	14	15
Magnetic Tape Format 0		0	ect = 1 onn = 0	Word Transfer Ready Interrupt	End of Record Interrupt	0	Rewind	Erase 4 Inches of Tape	BCD = 1 Binary = 0	Den	sity ≭ ≭		Tape Transpor	t	Current Word Address In	Cha Per	racters Word
Magnetic Tape Format 1	BTC Initi			Word Transfer Ready Interrupt	End of Record Interrupt	1	Write Record	Write End of File	Read Record	Advance Record	Advance/ End of File	Backspace Record	Backspace End of File		Current Word Address In		
ASR-33/35				In	Out	Reader Mode	Key Mode	Clear									
Paper Tape Reader and Punch				In	Out	F	eed	Reader Enable	Reader Disable								
Card Reader/ Punch				In	Out	Feed Card Reader		Read Stacker Offset	Feed Card Punch	Eject Card (Punch)	Punch Stacker Offset						
X-Y Plotter				Process Complete		Pen Down	Pen Up	Drum Down	Drum Up	Carriage Left	Carriage Right						
Line Printer				End of Print	Buffer Not Busy	Advance Paper To Format Tape Chan- nel ** *	Advance l Line	Top of Form	Print Or F	Clear Buffer Format "n" if	Fill Buffer Bit 4						
Moveable									Number of	Tracks to be	Moved						
Head Disc Seek				Seek Error	Seek Complete		64	32	16	8	4	2	1	1		FWD ★★★★	REV ****
Moveable Head				Seek	Seek	Se	ector Number				Head	Number				Write	Read
Disc Data				Error	Complete	8	4	2	1	8	4	2	1	0			
Fixed Head Disc				Checksum Error or	Read Overflow or				1	 Frack Numbe	r J					1	1
Select Track				Program Erro r	Program Error	256	128	64	32	16	8	4	2	I			
Fixed Head Disc Read				Checksum Error or Program Error	Read Overflow or Write Overflow	Read Sequential					4		Starting Secto	r 		0	Read 1
Fixed Head Disc Write				Checksum Error or Program Error	Read Overflow or Write Overflow	Write Sequential					•		Starting Secto	>		Write]	0
CRT				Overflow	Stop	Display On	Display Off										

APPENDIX C Sel 810 Peripheral Device Command and test code formats

#Interrupt Levels: Bit 2 = Group 1, Level 1 Bit 3 = Group 1, Level 2

 #4 Magnetic Tane Density:

 Bits
 Density
 Bits
 Daracters

 8
 9
 14
 15
 Per Word
 0 0 200 BPI 0 1 0 1 556 BPI 1 0 1 0 800 BPI 1 1

1 2 #**. When a one is present in bit position 4, advance to the format tape channel number (expressed in octal) represented by the bits present in positions 7, 8 and 9.

**** To seek track 00 both bits must be zero.

C-1

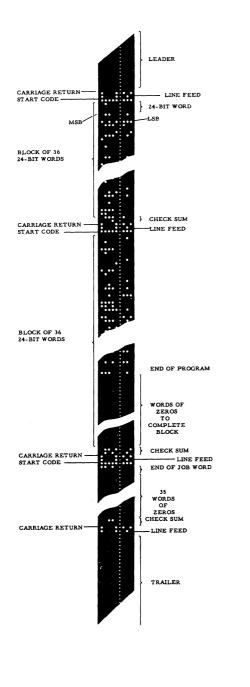
95118A. Cl

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Card Reader and Punch										Skip No Punch Error						
Movable Head Disc					Skip Seek Complete	Skip No Seek Error	Skip on Beginning of Disc	Skip on Beginning of Sector	Skip Pack on Line	Skip No Read Overflow	Skip No Write Overflow	Skip No Checksum Error	Skip No File Unsafe	Skip DCU Ready	Skip Not Busy	
Fixed Head Disc	Skip on No Program Error	Skip on Disc on Line	Skip on No Disc Read Overflow	Skip on No Disc Write Overflow	Skip on No Checksum Error	Skip on No Disc File Area Protected	Skip on Disc Control Not Busy	,								
Magnetic Tape	Skip on Not Busy	Skip on No End of File	Skip on No Overflow	Skip on Load Point	Skip on End of Record Interrupt	Skip on No Parity Error	Skip on Write Ring In	Skip on No End of Tape	Skip on Rewinding	Skip on No CRC Error (9 Track Only)						
Line Printe r					Skip Not Busy		Skip No Parity Error		Skip No Bottom of Form	Skip if Printer Operable						
Interval Timer									Disable Zero Count Interrupt							

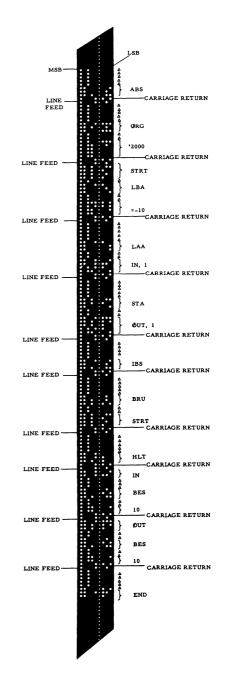
SEL 810 PERIPHERAL DEVICE COMMAND AND TEST CODE FORMATS

TEU SECOND WORD FORMAT

APPENDIX D SEL 810 PAPER TAPE FORMATS



SEL 810B ASSEMBLER AND COMPILIER OBJECT PROGRAM OUTPUT TAPE, MUST BE LOADED WITH THE SEL MNEMBLER LOADER PROGRAM.



ASC II CODE ASSEMBLER SOURCE INPUT IN CARD FORMAT IMAGE. MUST BE LOADED BY ASSEMBLER.

95118A. DI

APPENDIX E SEL 810 ASSEMBLER OUTPUT FORMATS

0	0	0	0	0	0	0	0		DATA	
DI	RE	СТ	LO	٩D	:]	Dat	a o	r N	Ion-memory-referencing instructions.	
0	1	R	OP	С	OD	E	x	I	ADDRESS	
0 	1 	L	L			_	1	L		
0 MI	1 EM	L	L			_	1	L	ADDRESS INSTRUCTIONS: R = Relocation flag (DAC) OP = '13, 14-bit address co	onstant
0 MI	1 EM	L	L			_	1	L	INSTRUCTIONS: R = Relocation flag	

1	1	R	OP CODE	1	LITERAL
		L	[]		╘╴╴╢╶╴╢╴╴╢╌╴╢╌╴╢╴╴╢╌╴╢╌╴╢╌╴╢╌╴╢╌╴╢╌╴╢

LITERAL REFERENCING INSTRUCTIONS:

C_D 0_0_0_0_0_0 N_0_0_0_0_0_0 SIZE S1 S2 S3	1 0	R	OF	, c	OD	Ε	2	x	Ι		I	1		ADI	DRI	ESS	LEN	1G.	ГН			1	
S1 S2 S3	C D	0	0	0	0	0	0)	N	0	0	0	0	0	0			- 1	ر s	ΙŻ	Ε,		
			_S1									S	52						,	S3	5		
S4 S5 S6 S6			_S4			1						S	55							S6	, ,		

SUBROUTINE OR COMMON:CD = 00: Subroutine definition (NAME)CD = 10: Common defn.Address = relative entry point.Address = lengthAddress = relative entry point.

= 11: Common request Address = rel. to block. N = negation flag = 01: Subroutine call (CALL) Address = 0

1 1 R CODE O N	ADDRESS
SPECIAL ACTION:	
Code = 00, Establish Load Point	= 06, Turn on CHAIN flag
= 01, END Jump	= 07, Turn on Load flag
= 02, STRING	= 10, END-OF-JOB
= 03, 9-Bit ADD-TO	
= 04, 14-Bit ADD-TO (DAC)	

= 05, 15-Bit ADD-TO (EAC)

95118A.E1



APPENDIX F

NUMERICAL INFORMATION

NUMERICAL OCTAL TO DECIMAL CONVERSION

OCTAL MULTIPLICATION

M•N or **N•M**

=	1	2	3	4	5	6	7	10
1	1	2	3	4	5	6	7	10
2	2	4	6	10	12	14	16	20
3	3	6	11	14	17	22	25	30
4	4	10	14	20	24	30	34	40
5	5	12	17	24	31	36	43	50
6	6	14	22	30	36	44	52	60
7	7	16	25	34	43	52	61	70
10	10	20	30	40	50	60	70	100

OCTAL ADDITION

M+N or N+M

m	1	2	3	4	5	6	7	10
1	2	3	4	5	6	7	10	11
2	3	4	5	6	7	10	.11	12
3	4	5	6	7	10	11	12	13
4	5	6	7	10	11	12	13	14
5	6	7	10	11	12	13	14	15
6	7	10	11	12	13	14	15	16
7	10	11	12	13	14	15	16	17
10	11	12	13	14	15	16	17	20

 $^{8^0 = 1}$ $8^1 = 8$ $8^2 = 64$ $8^3 = 512$ $8^4 = 4096$ $8^5 = 32768$

NUMERICAL INFORMATION

TABLE OF POWERS OF TWO

2 ⁿ	n	2-*						
1	0	1.0						
2	1	1.0 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	-					
1 28	7	0.007 812	2					
256	8	0.003 906	25					
51 2	9	0.001 953	1 25					
1 024	10	0.000 976						
2 048	11	0.000 488	281 25					
4 096	12	0.000 244	140 625					
8 1 9 2	13	0.000 122	070 312	5				
16 384	14	0.000 061	035 156	25				
32 768	15	0.000 030	517 578	1 25				
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						
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	1 25			
4 1 9 4 3 0 4	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 9	5		
67 108 864	26	0.000 000						
134 217 728	27	0.000 000	007 450	580 596	923 828 1	1 25		
268 435 456	28	0.000 000	003 725	290 298	461 914 (0625		
536 870 912	29	0.000 000	001 862	645 149	230 957 (031 25		
1 073 741 824	30	0.000 000						
2 147 483 648	31	0.000 000	000 465	661 287	307 739	257 812 5		
4 294 967 296	32	0.000 000						
8 589 934 592	33	0.000 000						
17 179 869 184	34	0.000 000						
34 359 738 368	35	0.000 000	000 029	103 830	450 (33	103 613 2	31 25	
68 719 476 736	36	0.000 000	000 014	551 915	228 366	851 806 6	40 625	
137 438 953 472	37	0.000 000						
274 877 906 944	38	0.000 000						
549 755 813 888	39	0.000 000	000 001	818 989	403 545	856 475 8	30 078 125	5
1 099 511 627 776	40	0.000 000	000 000	909 494	701 772	928 237 9	915 039 06	25
2 199 023 255 552	41	0.000 000	000 000	454 747	350 886	464 118 9	57 519 53	1 25
4 398 046 511 104							178 759 76	
8 796 093 022 208	43	0.000 000	000 000	113 686	837 721	616 029 7	739 379 88	2 81 2 5
17 592 186 044 416	44	0.000 000	000 000	056 843	418 860	808 014 8	369 689 94	1 406 25
35 184 372 088 832	45						434 844 97	
70 368 744 177 664	46							5 3 51 562 5
140 737 488 355 328	47	0.000 000	000 000	007 105	427 357	601 001 8	858 711 24	2 675 781 25
281 474 976 710 656	48	0.000 000	000 000	003 552	713 678	800 500 9	929 355 62	1 337 890 625

NUMERICAL INFORMATION

OCTAL-DECIMAL INTEGER CONVERSION TABLE

	i	0	1	2	3	4	5	6	7
0000 1 0000							00.05	0006	0007
to to	0000 0010	0000						0006 0014	
0777 0511	0020				0019			0022	
(Octal) (Decimal)	0030	0024	0025	0026	0027	0028	0029	0030	
	0040				0035				
Octal Decimal	0050	0040		0042	0043			0046 0054	0047
10000 - 4096	0060	0048		0050		0052			0063
20000 - 8192									
30000 - 12288	0100	0064				0068		0070	
40000 - 16384		0072			0075			0078 0086	
50000 - 20480	0120	0080				0092		0094	
60000 - 24576 70000 - 28672		0096						0102	
10000-100/1		0104	0105	0106	0107	0108	0109	0110	
	0160							0118	
	0170	0120	0121	0122	0123	0124	0125	0126	0127
	0200	0128	0129	0130	0131	0132	0133	0134	0135
	0210	0136	0137	0138	0139	0140	0141	0142	0143
	0220							0150	
	0230	0152						0158	
	0240	0160	0161	0162	0103	0104	0103	0166 0174	
	0250	0176	0177	0178	0179	0180	0181	0182	0183
	0270	0184	0185	0186	0187	0188	0189	0190	0191
							0107	0100	
	0300	0192	0193	0194	0195	0196	0197	0198 0206	0199
	0320	0200	0201	0210	0211	0212	0213	0214	0215
	0330	0216	0217	0218	0219	0220	0221	0222	0223
	0340	0224	0225	0226	0227	0228	0229	0230	0231
		0232						0238	
			0241	0242	0243	0244	0245		
	0360		0249		0251	0252		0254	0255
	0370	0248	0249	0250 2	0251	4	0253	6	7
1800 0512	0370	0248	0249	0250 2	0251	4	0253	6	7
to to	0370 1000 1010	0248 0 0512 0520	0249 1 0513 0521	0250 2 0514 0522	0251 3 0515 0523	0252 4 0516 0524	0253 5 0517 0525	6 0518 0526	7 0519 0527
to to 1777 1023	0370 1000 1010 1020	0248 0 0512 0520 0528	0249 1 0513 0521 0529	2 0514 0522 0530	0251 3 0515 0523 0531	4 0516 0524 0532	0253 5 0517 0525 0533	6 0518 0526 0534	7 0519 0527 0535
to to	0370 1000 1010 1020 1030	0248 0 0512 0520 0528 0536	0249 1 0513 0521 0529 0537	0250 2 0514 0522 0530 0538	3 0515 0523 0531 0539	4 0516 0524 0532 0540	0253 5 0517 0525 0533 0541	6 0518 0526 0534 0542	7 0519 0527 0535 0543
to to 1777 1023	1000 1010 1020 1030 1040	0248 0512 0520 0528 0536 0544	0249 1 0513 0521 0529 0537 0545	2 0514 0522 0530 0538 0546	3 0515 0523 0531 0539 0547	4 0516 0524 0532 0540 0548	0253 5 0517 0525 0533 0541 0549	6 0518 0526 0534 0542 0550	7 0519 0527 0535 0543 0551
to to 1777 1023	0370 1000 1010 1020 1030	0 0512 0520 0528 0536 0544 0552	0249 1 0513 0521 0529 0537 0545	2 0514 0522 0530 0538 0546 0554	3 0515 0523 0531 0539	4 0516 0524 0532 0540 0548 0556	0253 5 0517 0525 0533 0541 0549 0557	6 0518 0526 0534 0542 0550 0558	7 0519 0527 0535 0543 0551 0559
to to 1777 1023	0370 1000 1010 1020 1030 1040 1050	0 0512 0520 0528 0536 0544 0552	1 0513 0521 0529 0537 0545 0553 0561	2 0514 0522 0530 0538 0546 0554 0562	3 0515 0523 0531 0539 0547 0555 0563	4 0516 0524 0532 0540 0548 0556 0564	0253 5 0517 0525 0533 0541 0549 0557 0565	6 0518 0526 0534 0542 0550 0558	7 0519 0527 0535 0543 0551 0559 0567
to to 1777 1023	1000 1010 1020 1030 1040 1050 1060	0248 0512 0520 0528 0536 0544 0552 0560 0568	0249 1 0513 0521 0529 0537 0545 0553 0561 0569	2 0514 0522 0530 0538 0546 0554 0554 0562 0570	3 3 0515 0523 0531 0539 0547 0555 0563 0571	4 0516 0524 0532 0540 0548 0556 0564 0572	0253 5 0517 0525 0533 0541 0549 0557 0565 0573	6 0518 0526 0534 0542 0550 0558 0566 0574	7 0519 0527 0535 0543 0551 0559 0567 0575
to to 1777 1023	1000 1010 1020 1030 1040 1050 1060	0 0512 0520 0528 0536 0544 0552 0560 0568 0576	0249 1 0513 0521 0529 0537 0545 0553 0561 0569	2 0514 0522 0530 0538 0546 0554 0562 0570 0578	3 0515 0523 0531 0539 0547 0555 0563 0571 0579	4 0516 0524 0540 0548 0556 0564 0572 0580	0253 5 0517 0525 0533 0541 0549 0557 0565 0573	6 0518 0526 0534 0542 0550 0558 0566	7 0519 0527 0535 0543 0551 0559 0567 0575
to to 1777 1023	0370 1000 1010 1020 1040 1050 1060 1070 1100 1110	0 0 0512 0528 0536 0544 0552 0560 0568 0568 0576 0584 0592	1 0513 0529 0537 0545 0553 0561 0569 0577 0585 0593	2 0514 0520 0538 0546 0554 0562 0570 0578 0586 0594	3 0515 0523 0531 0539 0547 0555 0563 0571 0579 0587 0595	4 0516 0522 0540 0548 0556 0564 0572 0580 0588 0596	0253 5 0517 0523 0541 0549 0557 0565 0573 0581 0589 0597	6 0518 0526 0534 0550 0558 0566 0574 0582 0590 0598	7 0519 0527 0535 0543 0551 0559 0567 0575 0583 0591 0595
to to 1777 1023	0370 1000 1010 1020 1040 1040 1060 1060 1070 1100 1110 1120 1130	0 0512 0520 0536 0544 0552 0560 0568 0576 0584 0592 0600	1 0513 0521 0527 0545 0553 0569 0569 0577 0585 0593 0601	2 0514 0522 0538 0546 0554 0552 0570 0578 0584 0584 0594 0602	3 0515 0523 0531 0539 0547 0555 0563 0571 0579 0587 0585 0603	4 0516 0524 0540 0548 0564 0564 0564 0564 0588 0596 0604	0253 5 0517 0523 0541 0549 0557 0565 0573 0581 0589 0597 0605	6 0518 0526 0534 0550 0558 0566 0574 0582 0590 0598 0606	7 0519 0527 0535 0543 0551 0559 0567 0575 0583 0591 0595 0607
to to 1777 1023	0370 1000 1010 1020 1030 1040 1040 1060 1070 1100 1110 1110 1110	0 0512 0520 0528 0536 0544 0552 0560 0568 0568 0576 0584 0590 0608	1 0513 0521 0529 0537 0545 0553 0569 0577 0585 0591 0501 0609	2 0514 0522 0530 0538 0546 0554 0562 0570 0578 0586 0594 0602 0610	3 0515 0523 0531 0539 0547 0555 0563 0571 0579 0587 0593 0603 0611	4 0516 0524 0540 0548 0564 0564 0564 0564 0588 0588 0588 05684 0564	0253 5 0517 0525 0533 0541 0549 0557 0565 0573 0581 0589 0597 0505 0581	6 0518 0526 0534 0550 0558 0566 0574 0582 0590 0598 0606 0614	7 0519 0527 0535 0543 0559 0567 0575 0583 0591 0599 0567 0575
to to 1777 1023	1000 1010 1020 1040 1050 1060 1100 1110 1120 1140 1140 1150	0248 0512 0520 0528 0544 0552 0560 0568 0576 0584 0592 0600 0608 0616	1 0513 0521 0529 0535 0553 0561 0569 0577 0585 0593 0601 0609 0617	2 0250 0514 0522 0530 0538 0546 0552 0570 0570 0578 0586 0594 0610 0610	3 0515 0523 0531 0537 0555 0563 0571 0579 0587 0595 0603 0611 0619	4 0516 0524 0540 0548 0556 0564 0572 0580 0588 0596 0588 0596 0604 0612 0620	0253 5 0517 0525 0533 0541 0549 0557 0565 0573 0565 0573 0581 0589 0597 0605 0613 0613 0621	6 0518 0526 0534 0542 0550 0558 0566 0574 0582 0590 0598 0606 0614 0622	7 0519 0527 0535 0543 0559 0567 0575 0583 0591 0599 0607 0615 0615
to to 1777 1023	0370 1000 1010 1020 1030 1040 1040 1060 1070 1100 1110 1110 1110	0248 0512 0520 0528 0544 0544 0544 0544 0544 0544 0544 054	1 0513 0521 0537 0545 0553 0553 0569 0577 0585 0593 0601 0609	2 0250 0514 0522 0530 0538 0546 0554 0554 0554 0554 0554 0554 0559 0610 0618 0618	3 0515 0523 0531 0539 0547 0555 0563 0571 0579 0587 0593 0603 0611	4 0552 0540 0540 0548 0556 0564 0556 0568 0556 0588 0596 0604 0612 0620 0620	0253 5 0517 0525 0533 0541 0549 0557 0565 0573 0581 0589 0597 0605 0613 0621	6 0518 0526 0534 0542 0550 0558 0566 0574 0582 0590 0598 0606 0614 0622 0630	7 0519 0527 0535 0543 0551 0559 0567 0575 0583 0599 0607 0615 0622 0631
to to 1777 1023	0370 1000 1010 1020 1030 1050 1050 1050 1050 1050 1050 105	0248 0512 0520 0528 0536 0544 0552 0560 0568 0576 0584 0592 0600 0608 0608 0608 0608	1 0513 0521 0529 0537 0545 0553 0553 0553 0553 0561 0569 0609 0609 0609 0609	0250 2 0514 0522 0530 0546 0554 0556 0594 0502 0570 0570 0594 0602 0610 0602 0610 0618 0626	0251 3 0515 0523 0531 0555 0563 0571 0555 0603 0611 0619 0627 0635	4 0516 0524 0532 0540 0556 0556 0556 0556 0556 0556 0556	0253 5 0517 0525 0541 0549 0557 0565 0573 0605 0605 0605 0603 0621 0629 0637	6 0518 0526 0534 0550 0558 0556 0574 0582 0590 0606 0614 0622 0630 0638	7 0519 0527 0535 0543 0551 0567 0583 0595 0607 0615 0622 0631 0635
to to 1777 1023	0370 1000 1010 1020 1030 1040 1050 1040 1050 1040 1050 1040 1100 1110 11200 1150 1150 1150 1150 11	0248 0512 0520 0528 0536 0544 0552 0560 0568 0568 0568 0576 0568 0568 0568 0568 0568 0568 0568 056	1 0513 0521 0529 0537 0545 0553 0569 0569 0569 0569 0569 0569 0609 0609	0250 2 0514 0522 0530 0538 0546 0542 0570 0578 0586 0594 0610 0618 0626 0634 0642	0251 3 0515 0523 0531 0539 0547 0555 0563 0571 0595 0603 0611 0619 0627 0635 0643	4 0516 0524 0524 0524 0532 0540 0544 0554 0564 0568 0568 0568 0568 0568 0568 0568 0568	0253 5 0517 0525 0533 0541 0549 0557 0565 0573 0665 0573 0665 0613 0621 0629 0637 0645	6 0518 0526 0534 0542 0550 0556 0574 0582 0590 0598 0590 0598 0614 0622 0630 0638 0646	7 0519 0527 0535 0559 0559 0559 0559 0559 0599 0607 0615 0623 0639 0639 0639
to to 1777 1023	0370 1000 1010 1020 1030 1040 1050 1050 1050 1050 1050 1050 105	0248 0512 0520 0528 0536 0544 0552 0566 0556 0568 0556 0568 0556 0568 0592 0600 0608 0616 0624 0622 0640	1 0513 0521 0529 0537 0545 0553 0553 0569 0577 0585 0593 0601 0609 0617 0625 0633 0641	2 0250 2 0514 0522 0530 0538 0546 0554 0554 0554 0554 0554 0554 0602 0610 0618 0602 0610 0626 0634	0251 3 0515 0523 0531 0537 0555 0563 0571 0595 0603 0601 0611 0619 0627 0635	4 0516 0524 0540 0556 0556 0556 0556 0558 0596 0604 0612 0628 0604 0628 0636	0253 5 0517 0525 0533 0541 0557 0557 0557 0557 0557 0557 0605 0613 0629 0603 0629 0637	6 0518 0526 0532 0550 0558 0566 0574 0582 0590 0598 0606 0514 0622 0638 0638 0646 0654	7 0519 0527 0533 0559 0559 0559 0597 0599 0607 0619 0622 0631 0635 0647 0655
to to 1777 1023	0370 1000 1010 1020 1030 1040 1070 1100 1110 1120 1140 1140 1140 1140 114	0248 0512 0520 0528 0536 0544 0552 0560 0568 0568 0568 0576 0568 0568 0568 0568 0568 0568 0568 056	0249 1 0513 0521 0529 0537 0545 0553 0569 0577 0685 0609 0617 0625 0633 0641 0649	2 0514 0522 0530 0546 0554 0554 0554 0554 0622 0610 0618 0626 0634 0642 0658	0251 3 0515 0523 0531 0539 0543 0555 0663 0611 0619 0627 0635 0643 0659	0252 4 0516 0524 0548 0566 0564 0572 0588 0596 0604 0612 0620 0628 0636 0620 0628 0636	0253 5 0517 0525 0533 0541 0549 0557 0555 0557 0555 0557 0605 0603 0613 0621 0627 0633 0645 0633 0645	6 0518 0526 0532 0550 0558 0566 0574 0582 0590 0598 0606 0514 0622 0638 0638 0646 0654	7 0519 0527 0543 0551 0559 0567 0595 0607 0615 0607 0615 0622 0633 0647 0655
to to 1777 1023	0370 1000 1010 1020 1030 1040 1050 1070 1120 1130 1140 1156 1170 1220 1210 1220 1240	0248 0512 0520 0528 0536 0544 0556 0568 0568 0568 0568 0600 0608 0610 0624 0624 0622 0640 0648 0656 0672	0249 1 0513 0521 0537 0545 0553 0569 0577 0585 0601 0609 0617 0603 0601 0641 0649 0657 0663	0250 2 0514 0522 0530 0546 0554 0562 0570 0578 0586 0594 0602 0610 0618 0628 0658 0558 0558 0558 0558 0558 0558 0558 0558 0558 0568 0654 06588 06588 06588 06588 06588 06588 06588 06588 06588	0251 3 0515 0523 0539 0547 0555 0563 0571 0595 0603 0611 0619 0627 0635 0643 0651 0659 0665	0252 4 0516 0524 0530 0548 0556 0564 0552 0580 0588 0596 0664 0612 0620 0620 0620 0620 0620 0644 0652 0660 0664	0253 5 0517 0525 0533 0541 0549 0557 0565 0573 0581 0589 0597 0605 0613 0621 0623 0613 0621 0637 0645 0653 0661	6 0518 0526 0534 0558 0558 0558 0558 0558 0606 0614 0622 0630 0638 0646 0654 0654	7 0519 0527 0543 0551 0555 0555 0555 0555 0555 0555 055
to to 1777 1023	0370 1000 1010 1022 1030 1040 1055 1060 1010 1120 1130 1140 1120 1150 1150 1150 1120 1210 1220 122	0248 0512 0520 0522 0533 0544 0552 0568 0568 0568 0568 0568 0668 0668 0668	0249 1 0513 0521 0529 0537 0545 0553 0569 0659 0609 0617 0625 0633 0601 0609 0649 0649 0649 0657 0663 0681	0250 2 0514 0522 0530 0546 0554 0554 0562 0570 0578 0586 0594 0682 0610 0618 065	0251 3 0515 0523 0539 0547 0555 0563 0571 0579 0687 0611 0619 0627 0635 0611 0659 0663 0659 0663	0252 4 0516 0524 0532 0540 0540 0556 0564 0556 0596 0696 0696 0696 0696 0696 0696	0253 5 0517 0525 0533 0541 0559 0549 0557 0665 0613 0621 0629 0633 0663 0663 0663 0663 0665	6 0518 0526 0534 0550 0558 0550 0590 0590 0590 0590 0590	7 0519 0527 0535 0543 0559 0559 0657 0615 0622 0631 0633 0647 0655 0662 0647 0655 0662
to to 1777 1023	0370 1000 1010 1020 1030 1050 1050 1050 1050 1050 1100 1110 1120 1130 1140 1150 1150 1150 1160 1210 1210 1210 1220 1230 1240 1250	0248 0512 0512 0528 0536 0544 0552 0560 0568 0564 0592 0600 0608 0616 0592 0600 0608 0616 0624 0624 0622 0640 0624	0249 1 0513 0529 0537 0545 0553 0569 0577 0865 0609 0617 0625 0633 0641 0649 0657 0665 0673 0689	0250 2 0514 0520 0530 0538 0546 0554 0552 0570 0578 0586 0526 0626 0634 0622 0650 0626 0653 0666 0654 0666 0654 0666 0656 0666 0656 065	0251 3 0515 0523 0531 0553 0563 0563 0563 0603 0603 0627 0635 0643 0651 0627 0655 0663 0667 0667	0252 4 0516 0524 0532 0548 0556 0588 0564 0652 0628 0628 0628 0628 0628 0628 0628 062	0253 5 0517 0525 0533 0541 0565 0573 0665 0665 0665 0665 0665 0665 0665 066	6 0518 8 0526 0534 0550 0558 0566 0574 0582 0590 0614 0622 0630 0638 0646 0654 0670 0678 0670	7 0511 0527 0533 0542 0555 0567 0557 0557 0557 0557 0557 0559 0615 0622 0631 0652 0662 0663 0666 0667 0667 0667
to to 1777 1023	0370 10000 1010 1020 1030 1040 1055 1060 1107 1110 1120 1110 1110 1110 1110 111	0248 0512 0520 0528 0544 0552 0560 0568 0566 0668 0624 0602 0668 0664 0662 0664 0668 0664 0668 06664 0668 06664 0668 06666	0249 1 0513 0521 0529 0537 0545 0553 0569 0609 0617 0625 0633 0601 0641 0649 0657 0655 0653 0657 0655 0653 0659 0657 0657 0657 0657 0657 0657 0557 0559 0557 0559 0557 0559 0557 0559 0557 0559 0557 0559 0557 0559 0557 0559 0557 0559 0557 0559 055	0250 2 0514 0522 0530 0538 0546 0554 0552 0570 0578 0582 0594 0602 0594 0618 0626 0634 0618 0658 0666 0674 0682 0690 0698	0251 3 0515 0523 0531 0537 0555 0663 0571 0687 0603 0661 0627 0635 0663 06643 06659 0667 0659 06691	0252 4 0516 0524 0532 0540 0556 0568 0572 0580 0588 0596 0604 0628 0604 0628 0668 0668 0668 0668 0668 0668 0668	0253 5 0517 0525 0533 0541 0541 0549 0557 0565 0573 0565 0573 0581 0589 0597 0603 0621 0629 0637 0645 0669 0669 0670 0685 0669 0701	6 0518 0526 0534 0550 0556 0574 0582 0550 0576 0576 0614 0622 0630 0638 0646 0652 0662 0670 0662 0662 0662 0662 0670 0678	7 0519 0522 0533 0555 0555 0555 0593 0699 0607 0633 0663 0663 0666 0667 0668 0668 0669 0669 0669
to to 1777 1023	0370 1000 1012 1030 1050 1050 1050 1100 1110 1120 1130 1140 1170 1210 1220 1220 1220 1220 1220 122	0248 0512 0520 0528 0536 0546 0552 0560 0568 0576 0584 0592 0600 0608 0616 0622 0640 0648 0652 0664 0648 0656 0664 0652 0666 0668 0666 0668 0666 0668 0669 0668 0669 0669	0249 1 0513 0529 0537 0545 0553 0553 0569 0573 0669 0617 0625 0633 0641 0649 0657 0649 0657 0649 0657 0649 0657 0649 0657 0649 0649 0657 0649 0659 0657 0659 0559 055	2 0514 0514 0520 0530 0546 0546 0546 0594 0620 0618 0620 0638 0620 0638 0620 0638 0620 0638 0642 0650 0654 0650 0658 0666 0659 0658 06674 0652 0650 0658 0674 0652 0659 0654 0654 0654 0654 0654 0654 0654 0654	0251 3 0515 0523 0531 0555 0563 0555 0563 0571 0555 0663 0611 0619 0627 0643 0651 0655 0663 0651 0659 0667 0669 0699	4 0516 0524 0532 0540 0556 0556 0588 0596 0612 0620 0620 0628 0620 0628 0620 0628 0620 0626 0626	0253 5 0517 0525 0533 0541 0557 0555 0573 0585 0597 0689 0597 0689 0597 0689 0597 0689 0693 0653 0653 0653 0653 0653 0653 0653 065	6 0518 0526 0534 0558 0558 0558 0558 0558 0558 0598 0614 0622 0630 0638 0646 0654 0654 0656 0678 0678 0678	7 0519 0527 0535 0543 0555 0555 0555 0555 0659 0607 0635 0662 06635 06647 06635 06647 06635 0667 0667 0667 0667 0669 0703 0699 0703
to to 1777 1023	0370 10000 1010 1020 1030 1040 1050 105	0248 0512 0520 0528 0544 0552 0560 0568 0576 0584 0552 0600 0608 0616 0584 0616 0584 0656 0624 0624 0624 0628 0626 0624 0628 0626 0628 0656 0624 0628 0656 0656 0656 0657 0672 0680 0657 0672 0680 0657 0672 0680 0657 0672 0670 0672 0670 0672 0670 0672 0670 0672 0670 0672 0670 0672 0670 0572 0570 0578 0576 0578 0578 0578 0578 0578 0578 0578 0578	0249 1 0513 0521 0529 0537 0545 0553 0569 0609 0609 0609 0657 0625 0633 0641 0649 0657 0665 0673 0659 0697 0689 0697 0689 0697 0689 0697 0689 0697 0689 0697 0698 0697 0677 0697 0697 0697 0697 0697 0697 0677 0657 0677 0677 0657 0677 0677 0677 0675 0677 07777 07777 07777 07777 0777777 07777 07777 07777 07777	0250 2 0514 0522 0530 0546 0554 0562 0570 0570 0570 0602 0602 0610 0626 0634 0642 0658 0666 0658 0666 0658 0666 0658 0666 0658 0652 0650 0658 0602 0610 0620 0610 0620 0658 0602 0602 0602 0600 0658 0602 0706 070	0251 3 0515 0523 0531 0547 0555 0603 0611 0619 0627 0625 0663 0659 0663 0659 0663 0659 0663 0659 0663	0252 4 0516 0524 0524 0548 0556 0568 0568 0604 0612 0620 0628 0628 0626 0628 0626 0628 0626 0628 0650 0628 0650 0654 0652 0708 0708	0253 5 0517 0525 0533 0549 0557 0605 0613 0589 0605 0613 0629 0637 0665 0663 0661 0629 0667 0665 0663 0661 0669 0677 0665	6 0518 0526 0532 0550 0558 0556 0574 0558 0558 0558 0606 0518 0622 0630 0638 06456 0662 0670 0678 0662 0670 0678 0662 0670 0678 0702 0710 0718	7 0519 0527 0535 0555 0555 0555 0555 0555 0555 0655 06
to to 1777 1023	0370 1000 1010 1020 1030 1050 1050 1050 1070 1100 1120 1130 1200 1210 1220 1230 1240 1230 1240 1230 1260 1270 1300 1320 1220 1200	0248 0520 0520 0520 0520 0544 0552 0560 0544 0552 0560 0568 0566 0664 0662 0662 0664 0664 0664 0664 06	0249 1 0513 0521 0529 0537 0545 0553 0553 0553 0553 0601 0669 0633 0641 0649 0657 0655 0673 0655 0673 0659 0637 0705 0689 0697 0705 0721 0721 0721 0721 0721 0721 0721 0721 0721 0721 0721 0721 0721 0721 0721 0721 0757 0755 0757 0771 077	2 0250 0514 0522 0530 0538 0554 0554 0554 0554 0554 0554 0554 055	0251 3 0515 0523 0531 0537 0555 0663 0571 0697 0695 0667 0663 0667 0663 0667 0665 0667 0667 0675 0663 0671 0699 07075 0723	4 0516 0524 0524 0522 0540 0556 0564 0576 0588 0596 0688 0596 0628 0636 0628 0636 0668 0668 0668 0668 0668 0668 066	0253 5 0517 0525 0533 0541 0557 0665 0573 0581 0589 0597 0603 0603 0663 0663 0663 0663 0663 066	6 0518 0526 0534 0558 0558 0558 0558 0558 0598 0614 0622 0630 0638 0646 0654 0654 0670 0678 0694 0702	7 0519 0527 0543 0551 0553 0567 0575 0567 0595 0607 0635 0665 0665 0665 0665 0665 0665 0665
to to 1777 1023	0370 10000 1010 1020 1030 1040 1050 1040 1070 1100 1110 1120 1130 1140 1150 1140 1150 1120 1220 1230 1240 1250 1240 1250 1240 1250 1240 1250 1240 1250 1240 1250 1240 1250 1250 1250 1250 1250 1250 1250 125	0248 0512 0520 0528 0536 0544 0552 0560 0568 0656 0664 0652 0660 0664 0652 0660 0664 0652 0660 0664 0652 0660 0664 0672 0680 0656 0656 0656 0656 0656 0656 0657 0720 0720 0720 0720 0736	0249 1 0513 0521 0529 0537 0545 0553 0569 0657 0609 0609 0617 0625 0633 0601 0669 0657 0653 0665 0653 0667 0659 0657 0713 0712 0737 077	0250 2 0514 0522 0530 0546 0554 0546 0554 0622 0610 0626 0634 0622 0602 0602 0602 0658 0666 0674 0652 0658 0626 0658 0626 0634 0622 0600 0618 0622 0610 0619 0622 0610 0626 0634 0622 0610 0626 0626 0634 0622 0610 0626 0658 0626 0658 0626 0658 0626 0658 0626 0658 0626 0658 0626 0658 0626 0627 0737 0738 077	0251 3 0515 0523 0531 0539 0555 0563 0571 0595 0603 0611 0627 0635 0663 0659 0667 0659 0663 06691 0659 06690 0715 0723 0739	0252 4 0516 0524 0548 0556 0568 0568 0604 0612 0628 0604 0628 0660 0628 0660 0628 0660 0628 0660 0628 0660 0628 0660 0668 0670 0628 0660 0670 0670 0670 0670 0700 0700 0710 0724 0730 0740 0710 0740 0740 0740	0253 5 0517 0525 0533 0549 0557 0565 0573 0565 0603 0605 0603 0669 0669 0669 0669 0669 0669 0701 0665 0669 0701 0702 0701 0702 0717 0725 0733	6 0518 0526 0534 0550 0558 0566 0574 0582 0598 0606 0614 0622 0630 0638 0662 0659 0622 0670 0622 0670 0622 0670 0710 0712 0712 0712	7 0519 0527 0544 0555 0544 0555 0545 0557 0595 0662 06631 06631 06631 06631 06631 06631 0663 0667 06687 06687 0701 0711 0722 0733
to to 1777 1023	0370 1000 1010 1020 1030 1050 1050 1050 1050 1050 1050 105	0248 0512 0520 0528 0532 0564 0552 0560 0568 0552 0600 0608 0616 0624 0632 0632 0640 0648 0656 0664 0652 0660 0668 06664 0652 0650 0668 06664 0704	0249 1 0513 0521 0553 0553 0553 0553 0553 0553 0559 0557 0585 0593 0609 0617 0609 0633 0641 0649 0657 0633 0641 0649 0657 0705 0659 0633 0641 0659 0633 0641 0659 0633 0641 0659 0633 0651 0755 0659 0633 0659 0770 0729 0729 0739 0729 0745 0749 0770 0770 0770 0770 0770 0770 0770 0770 0770 0770 0770 0770 0770 077	2 0514 0514 0520 0530 0538 0554 0554 0554 0554 0554 0554 0554 055	0251 3 0515 0523 0531 0539 0555 0653 0571 0595 0603 0611 0619 0627 0635 0643 0651 0659 0667 0635 0667 0635 0667 0635 0667 0639 0677 0635 0643 0631 0639 0677 0635 0643 0631 0639 0677 0635 0643 0631 0639 0677 0635 0643 0631 0639 0677 0635 0643 0631 0639 0677 0635 0643 0631 0639 0677 0635 0643 0631 0639 0677 0635 0643 0631 0639 0677 0635 0643 0631 0639 0677 0635 0643 0631 0639 0677 0635 0637 0777 0777 0777 0773 0773 0773 0773 0777 0773 0773 0777 0773 0777 0773 0777 0773 0777 0773 07777 0777 07777 0777 07777 07777 07777 07777 07777 07	4 0516 0524 0532 0540 0556 0564 0556 0588 0596 0688 0596 0688 0596 0688 0596 0688 0612 0620 0620 0668 0652 0700 0668 0652 0700	0253 5 0517 0525 0533 0541 0557 0557 0557 0557 0557 0559 0577 0605 0633 0631 0637 0645 0653 0669 0677 0665 0669 0677 0665 0669 0701 0709 0707 0725 0733 0741	6 0518 0526 0534 0558 0558 0558 0558 0558 0590 0598 0614 0622 0630 0646 0622 0630 0648 0622 0630 0648 0659 0670 0670 0702 0702 0710 0718	7 0519 0527 0535 0543 0559 0595 0605 0615 0623 0647 0647 0647 0695 0667 0695 0667 0695 0667 0695 0667 0695 0670 0705 0695 0670 0647 0695 0647 0695 0647 0695 0647 0695 0647 0695 0647 0695 0647 0695 0647 0695 0615 0647 0655 0615 0623 0615 0647 0655 0647 0655 0755 0655 0755 0655 07555 0755 0755 0755 0755 0755 0755 0755 0755 0755 0755 0
to to 1777 1023	0370 10000 10101 10200 10300 1040 10500 1040 10500 1040 10500 10400 10500 10400 10500 10400 10500 10400 10500 10400 105000 10500 1000000	0248 0512 0520 0528 0536 0544 0552 0560 0568 0568 0568 0568 0656 0624 0624 0624 0622 0660 0668 0656 0656 0656 0656 0652 0672 0672 0672 0736 07736	0249 1 0513 0521 0529 0537 0545 0553 0569 0673 0669 0673 0669 0677 0655 0673 0673 0673 0671 0697 0705 0705 0737 0745 0737 0745 0753 0753 0753 0753 0753 0753 0753 0753 0753 0753 0753 0753 0753 0753 0753 0753 0753 0753 0755 0755 0755 0755 0755 0755 0757 0755 07577 0757 0757 0757 0757 0757 0757 0757 0757 0757 07	2 0514 0522 0530 0546 0554 0556 0596 0602 0610 0618 0626 0634 0652 0658 0658 0626 0658 0658 0658 0658 0658 0658 0658 065	0251 3 0515 0523 0531 0547 0555 0663 0671 0693 0667 0659 0663 0659 0663 0659 0663 0659 0663 0659 0663 0659 0663 0707 0715 0715	4 0516 0524 0524 0524 0548 0566 0564 0604 0612 0620 0604 0628 0604 0628 0600 0628 0600 0628 0600 0628 0600 0628 0600 0628 0600 0628 0600 0600	0253 5 0517 0525 0533 0541 0549 0557 0605 0603 0605 0603 0629 0637 0665 0663 0662 0665 0663 0661 0669 0667 0665 0663 0709 0707 0717 0725 0733 0741 0757	6 0518 0526 0534 0550 0558 0566 0574 0582 0598 0606 0614 0622 0630 0638 0662 0659 0622 0670 0622 0670 0622 0670 0710 0712 0712 0712	7 0519 0527 0543 0551 0567 0575 0582 0595 0607 0615 06622 0633 06635 06635 06635 06635 06635 06635 06635 06635 0701 0711 0712 0712 0713 0714 0733 0744 0755

	0	1	2	3	4	5	6	7
0400	0256	0257	0258	0259	0260	0261	0262	0263
0410	0264	0265	0266	0267	0268	0269	0270	0271
0420	0272	0273	0274	0275	0276	0277	0278 0286	0279
0430 0440	0280	0281 0289	0282 0290	0283 0291	0284 0292	0285 0293	0286	0287
0440	0296	0209	0290	0299	0300	0293	0302	0303
0460	0304	0305	0306	0307	0308	0309	0310	0311
0470	0312	0313	0314	0315	0316	0317	0318	0319
0500	0320	0321	0322	0323	0324	0325	0326	0327
0510	0328	0329	0330	0331	0332	0333	0334	033
0520	0336	0337	0338	0339	0340	0341	0342	0343
0530 0540	0344 0352	0345 0353	0346 0354	0347 0355	0348 0356	0349 0357	0350 0358	0351
0550	0360	0355	0362	0363	0350	0365	0356	0367
0560	0368	0369	0370	0371	0372	0373	0374	0375
0570	0376	0377	0378	0379	0380	0381	0362	0383
0600	0384	0385	0386.		0388	0389	0390	0391
0610	0392	0393	0394	0395	0396	0397	0398	0399
0620	0400	0401	0402	0403	0404	0405	0406	040
0630	0408	0409	0410	0411	0412	0413	0414	0415
0640 0650	0416 0424	0417 0425	0418 0426	0419 0427	0420 0428	0421 0429	0422 0430	0423
0660	0432	0433	0434	0435	0436	0437	0438	0439
0670	0440	0441	0442	0443	0444	0445	0446	044
0700	0448	0449	0450	0451	0452	0453	0454	0455
0710	0456	0457	0458	0459	0460	0461	0462	0463
			0466	0467	0468	0469	0470	0471
0720	0464	0465						
0730	0472	0473	0474	0475	0476	0477	0478	
0730 0740	0472 0480	0473 0481	0474 0482	0475 0483	0476 0484	0477 0485	0478 0486	0481
0730 0740 0750	0472 0480 0488	0473 0481 0489	0474 0482 0490	0475 0483 0491	0476 0484 0492	0477 0485 0493	0478 0486 0494	0483 0493
0730 0740 0750	0472 0480	0473 0481	0474 0482	0475 0483	0476 0484	0477 0485	0478 0486	0479 0487 0495 0503 0511
0730 0740 0750 0760	0472 0480 0488 0496	0473 0481 0489 0497	0474 0482 0490 0498	0475 0483 0491 0499	0476 0484 0492 0500	0477 0485 0493 0501	0478 0486 0494 0502	0487 0495 0503
0730 0740 0750 0760 0770	0472 0480 0488 0496 0504	0473 0481 0489 0497 0505	0474 0482 0490 0498 0506	0475 0483 0491 0499 0507	0476 0484 0492 0500 0508	0477 0485 0493 0501 0509	0478 0486 0494 0502 0510	0487 0495 0503 0511
0730 0740 0750 0760	0472 0480 0488 0496 0504	0473 0481 0489 0497 0505	0474 0482 0490 0498 0506	0475 0483 0491 0499 0507	0476 0484 0492 0500 0508	0477 0485 0493 0501 0509 5	0478 0486 0494 0502 0510 6 0774 0782	0487 0495 0503 0511 7 7 0775 0785
0730 0740 0750 0760 0770	0472 0480 0488 0496 0504 0 0 0	0473 0481 0489 0497 0505	0474 0482 0490 0498 0506 2 0770	0475 0483 0491 0499 0507 3 0771 0779 0787	0476 0484 0492 0500 0508 4 0772 0780 0788	0477 0485 0493 0501 0509 5 0773 0781 0789	0478 0486 0494 0502 0510 6 0774 0782 0790	0487 0495 0503 0511 7 7 0775 0783 0795
0730 0740 0750 0760 0770 1400 1410 1420 1430	0472 0480 0488 0496 0504 0504 0768 0776 0784 0792	0473 0481 0489 0497 0505 1 0769 0777 0785 0793	0474 0482 0490 0498 0506 2 0770 0778 0786 0794	0475 0483 0491 0499 0507 3 0771 0779 0787 0795	0476 0484 0492 0500 0508 4 0772 0780 0788 0796	0477 0485 0493 0501 0509 5 0773 0781 0789 0797	0478 0486 0494 0502 0510 6 0774 0782 0790 0798	0483 0493 0503 0511 0511 7 7 0773 078 079 079
0730 0740 0750 0760 0770 1400 1410 1420 1430 1440	0472 0480 0488 0496 0504 0 0768 0776 0784 0792 0800	0473 0481 0489 0497 0505 1 0769 0777 0785 0793 0801	0474 0482 0490 0498 0506 2 0770 0778 0786 0794 0802	0475 0483 0491 0499 0507 3 0771 0779 0787 0795 0803	0476 0484 0492 0500 0508 4 0772 0780 0788 0796 0804	0477 0485 0493 0501 0509 5 0773 0781 0789 0797 0805	0478 0486 0494 0502 0510 6 0774 0782 0790 0798 0806	048 0495 0503 0511 7 7 0775 078 079 079 079
0730 0740 0750 0760 0770 1400 1410 1420 1430 1440 1450	0472 0480 0488 0496 0504 0 0768 0776 0776 0784 0792 0800 0808	0473 0481 0489 0497 0505 1 0765 0777 0785 0793 0801 0809	0474 0482 0490 0498 0506 2 0770 0778 0776 0778 0786 0794 0802 0810	0475 0483 0491 0499 0507 3 0771 0779 0787 0795 0803 0811	0476 0484 0492 0500 0508 4 0772 0780 0788 0796 0804 0812	0477 0485 0493 0501 0509 5 0773 0781 0789 0797 0805 0813	0478 0486 0494 0502 0510 6 0774 0782 0798 0806 0814	0487 0495 0503 0511 7 0775 0785 0795 0795 0807 0815
0730 0740 0750 0760 0770 1400 1410 1420 1430 1440	0472 0480 0488 0496 0504 0 0768 0776 0784 0792 0800	0473 0481 0489 0497 0505 1 0769 0777 0785 0793 0801	0474 0482 0490 0498 0506 2 0770 0778 0786 0794 0802	0475 0483 0491 0499 0507 3 0771 0779 0787 0795 0803	0476 0484 0492 0500 0508 4 0772 0780 0788 0796 0804	0477 0485 0493 0501 0509 5 0773 0781 0789 0797 0805	0478 0486 0494 0502 0510 6 0774 0782 0790 0798 0806	0487 0495 0503 0511 7 0775 078 079 079 0807 081 082
0730 0740 0750 0760 0770 1400 1410 1420 1430 1440 1450 1460	0472 0480 0488 0496 0504 0 0768 0776 0784 0792 0800 0808 0816	0473 0481 0489 0497 0505 1 0769 0777 0785 0793 0801 0809 0817	0474 0482 0490 0498 0506 2 0770 0778 0786 0786 0794 0802 0810 0818	0475 0483 0491 0499 0507 3 0771 0779 0787 0795 0803 0811 0819	0476 0484 0492 0500 0508 4 0772 0780 0788 0796 0804 0812 0820	0477 0485 0493 0501 0509 5 0773 0781 0789 0797 0805 0813 0821	0478 0486 0494 0502 0510 6 0774 0792 0790 0798 0806 0814 0822	0487 0495 0503 0511 7 7 0775 0783 079 080 079 080 081 082 083
0730 0740 0750 0760 0770 1400 1410 1420 1430 1450 1450 1450 1470	0472 0480 0488 0496 0504 0 0 0 0 7 6 0 7 8 0 7 7 6 0 7 8 0 7 7 6 0 7 8 0 7 7 6 0 7 8 0 7 7 6 0 0 7 8 0 0 7 8 0 0 7 8 0 0 0 8 0 0 0 4 8 0 0 0 4 8 0 0 0 4 8 0 0 0 0	0473 0481 0489 0497 0505 1 0705 0777 0785 0793 0801 0809 0817 0825	0474 0482 0490 0498 0506 2 0770 0778 0776 0778 0776 0778 0778 0794 0802 0810 0818 0826	0475 0483 0491 0499 0507 3 0771 0779 0787 0795 0803 0811 0819 0827	0476 0484 0492 0500 0508 4 0772 0780 0788 0796 0804 0812 0820 0828 0836 0844	0477 0485 0493 0501 0509 5 0773 0781 0787 0797 0797 0805 0813 0821 0829 0837 0845	0478 0486 0494 0502 0510 6 0774 0782 0790 0798 0806 0814 0822 0830 0838 0846	0487 0495 0503 0511 7 7 0775 0795 0795 0795 0805 0815 082 083 0845
0730 0740 0750 0760 0770 1400 1410 1420 1430 1440 1440 1440 1440 1440 1510 1510	0472 0480 0498 0496 0504 0504 0768 0776 0784 0776 0800 0808 0808 0816 0824 0832 0840 0848	0473 0481 0489 0497 0505 0755 0755 0769 0777 0785 0793 0801 0805 0801 0805 0817 0825	0474 0482 0490 0506 2 0770 0778 0786 0794 0810 0810 0818 0826 0818 0826	0475 0483 0491 0507 3 0771 0779 0787 0795 0803 0811 0819 0827 0835 0843 0851	0476 0484 0492 0500 0508 4 0772 0780 0788 0796 0804 0812 0820 0828 0828 0844 0852	0477 0485 0493 0501 0509 773 0773 0781 0789 0797 0805 0813 0821 0829 0821 0829 0837 0845	0478 0486 0494 0502 0510 0714 0782 0790 0798 0806 0808 0808 0814 0822 0830 0838	0487 0495 0503 0511 7 7 0775 0795 0795 0795 0805 0815 082 083 0845 0835
0730 0740 0750 0760 0770 1400 1420 1430 1440 1440 1440 1440 1450 1450 1510 1520 1520	0472 0480 0498 0504 0504 0768 0776 0792 0830 0898 0816 0824 0848 0848 0848	0473 0481 0489 0505 1 0769 0777 0785 0801 0809 0817 0825 0831 0849 0849	0474 0482 0490 0506 2 0770 0778 0786 0786 0786 0786 0882 0882 0882 0810 0818 0826 0834 0850 0858	0475 0483 0491 0507 3 0771 0779 0785 0803 0811 0819 0827 0843 0851	0476 0484 0492 0500 0508 4 0752 0780 0786 0804 0820 0828 0836 0844 0852 0854	0477 0485 0493 0501 0509 0509 0509 0773 0781 0783 0781 0787 0805 0813 0821 0829 0837 0845 0853 0853	0478 0486 0494 0502 0510 0510 0714 0782 0790 0798 0806 0814 0822 0830 0848 0846 0854	0487 0495 0503 0511 7 7 7 7 7 7 7 7 7 7 7 7 7 8 7 9 7 9 7 9
0730 0740 0750 0760 0770 1400 1410 1420 1430 1420 1430 1420 1450 1510 1510 1530 1530	0472 0480 0488 0496 0504 0504 0768 0788 0786 0784 0832 0830 0848 0846 0848 0846 0848	0473 0481 0489 0505 1 0769 0777 0785 0793 0801 0809 0817 0825 0833 0841 0849 0857	0474 0482 0490 0506 2 0770 0778 0778 0778 0778 0778 0778 0810 0810	0475 0483 0491 0507 0507 0771 0779 0785 0803 0811 0827 0843 0859 0843	0476 0484 0492 0500 0508 4 0772 0780 0788 0796 0884 0882 0884 0884 0844 0852 0860 0868	0477 0485 0493 0501 0509 5 0773 0781 0787 0797 0805 0805 0805 0805 0845 0853 0861 0869	0478 0486 0494 0502 0510 0714 0782 0790 0798 0806 0814 0822 0830 0888 0846 0854 0854	0487 0495 0503 0511 7 7 7 7 7 7 7 7 8 7 8 7 9 7 9 7 9 7 9 7
0730 0740 0750 0760 0770 1400 1410 1420 1430 1440 1440 1440 1440 1440 1440 1510 151	0472 0480 0496 0504 0504 0768 0776 0784 0784 0784 0784 0832 0840 0816 0824 0832 0840 0848 0856 0864	0473 0481 0489 0497 0505 1 0769 0777 0785 0793 0801 0809 0817 0825 0833 0841 0849 0857 0865	0474 0482 0490 0506 2 0770 0778 0786 0786 0786 0786 0786 0786	0475 0483 0491 0499 0507 0750 0771 0775 0787 0795 0803 0811 0819 0827 0843 0851 0859 0867 0875	0476 0484 0492 0500 0500 0508 0750 0780 0772 0780 0788 0796 0804 0812 0820 0828 0844 0852 0868 0868	0477 0485 0493 0501 0509 5 0773 0781 0789 0797 0805 0813 0829 0813 0821 0829 0837 0845 0853 0869 0869	0478 0486 0494 0502 0510 0510 0714 0782 0790 0798 0806 0814 0822 0830 0818 0846 0854 0854 0870 0878	0487 0495 0503 0511 7 7 0775 0785 0799 0799 0805 0815 0822 0833 0845 0855 08655 08671 0875
0730 0740 0750 0760 0770 1400 1410 1420 1430 1420 1430 1420 1450 1510 1510 1530 1530	0472 0480 0488 0496 0504 0504 0768 0788 0786 0784 0832 0830 0848 0846 0848 0846 0848	0473 0481 0489 0505 1 0769 0777 0785 0793 0801 0809 0817 0825 0833 0841 0849 0857	0474 0482 0490 0506 2 0770 0778 0778 0778 0778 0778 0778 0810 0810	0475 0483 0491 0507 0507 0771 0779 0785 0803 0811 0827 0843 0859 0843	0476 0484 0492 0500 0508 4 0772 0780 0788 0796 0884 0882 0884 0884 0844 0852 0860 0868	0477 0485 0493 0501 0509 5 0773 0781 0787 0797 0805 0805 0805 0805 0845 0853 0861 0869	0478 0486 0494 0502 0510 0714 0782 0790 0798 0806 0814 0822 0830 0888 0846 0854 0854	0487 0495 0503 0511 7 7 7 7 7 7 7 7 8 7 8 7 9 7 9 7 9 7 9 7
0730 0740 0750 0750 0770 1400 1410 1420 1430 1430 1430 1430 1430 1510 1520 1520 1520 1520 1550	0472 0480 0496 0504 0768 0776 0784 0792 0850 0850 0850 0880 0848 0840 0848 0846 0848 0846 0848 0846 0848 0848	0473 0481 0489 0497 0505 1 0769 0777 0785 0793 08801 0809 0817 0825 0831 0849 0841 0849 0857 0845 0849	0474 0482 0490 0506 2 0770 0778 0786 0786 0786 0786 0786 0786	0475 0483 0491 0507 0507 075 0803 0711 0779 0883 0811 0819 0827 0835 0843 0851 0859 0867 0875	0476 0484 0492 0500 0508 4 0772 0780 0796 0804 0812 0820 0828 0844 0820 0828 0844 0850 08684 0866 08684 0866 08684 0866 0884	0477 0485 0493 0501 0509 5 0773 0781 0797 0805 0813 0829 0837 0845 0853 0861 0869 0877 0885	0478 0486 0494 0502 0510 0795 0790 0798 0806 0814 0822 0830 0846 0854 0854 0854 0858 0858 0858	0485 0495 0500 0511 7 7 7 7 7 7 7 7 8 7 8 7 8 7 8 7 9 7 9 7
0730 0740 0750 0770 0770 1400 1410 1420 1430 1440 1430 1440 1440 1510 1550 1550 1550 1550 155	0472 0480 0496 0504 0504 0768 0776 0792 0800 0808 0816 0824 0832 0840 0832 0840 0848 0846 0856 0868 0856 0888 08966	0473 0481 0489 0505 1 0769 0777 0785 0801 0809 0817 0825 0833 0841 08457 0865 0873 0881 0865 0873 0881	0474 0482 0490 0490 0498 0506 0490 0498 0498 0498 0498 0498 0482 0482 0482 0482 0482 0482 0482 048	0475 0483 0491 0499 0507 075 0803 0811 0819 0827 0835 0843 0851 0859 0867 0875 0883 0867 0875	0476 0484 0492 0500 0508 4 0772 0780 0788 0786 0804 0812 0820 0828 0836 0844 0850 0868 0868 0868 0868 0868 0868	0477 0485 0493 0501 0509 0501 0509 0773 0781 0797 0805 0813 0821 0829 0837 0845 0853 0861 0869 0877 0885 0893 0901	0478 0486 0494 0502 0510 07510 0750 0790 0790 0790 0790 0798 0806 0814 0822 0830 0854 0852 0870 0878 0886 0886 0886 0886 0886 0886	0485 0495 0501 0511 0511 0511 0511 077 078 079 079 079 079 080 081 082 083 084 085 0864 085 0867 0871 088 0891
0730 0740 0750 0750 0770 1400 1410 1420 1430 1440 1430 1540 1550 1550 1550 1550 1550 1550 1570 1600 1610	0472 0480 0488 0496 0504 0768 0776 0784 0776 0830 0880 0868 0868 0864 0840 0848 0856 0864 0856 0864 0888 0886 0888 0886 0888 0890 0904 0912	0473 0481 0489 0497 0505 0793 0777 0785 0793 0801 0809 0817 0825 0833 0841 0889 0857 0863 0881 0889 0887 0863 0889	0474 0482 0490 0490 0498 0506 0790 0778 0786 0786 0786 0786 0786 0786 078	0475 0483 0491 0499 0507 0779 0779 0779 0779 0787 0779 0835 0843 0851 0859 0865 0883 0851 0859 0875 0883 0891	0476 0484 0492 0500 0508 4 0792 0780 0780 0788 0796 0804 0812 0820 0828 0836 0844 0852 0868 0884 0852 0868 0884 08892 0900 0908 0916	0477 0485 5 5 0773 0781 0789 0797 0805 0813 0821 0829 0833 0845 0853 0861 08693 0893 0909	0478 0486 0494 0502 0510 0791 0792 0790 0798 0806 0814 0822 0830 0838 0846 0854 0854 0854 0886 0894 0894	0487 0495 0502 0511 7 7 0778 0799 0799 080 081 0822 083 0845 0855 0865 0865 0865 0865 0865 0865 086
0730 0740 0750 0750 0770 1400 1410 1420 1420 1420 1420 1420 142	0472 0480 0488 0496 0504 0504 0768 0776 0784 0792 0830 0808 0840 0840 0840 0840 0846 0856 0864 0856 0880 0888 0896 0912 0920	0473 0489 0489 0505 1 0769 0777 0785 0793 0801 0801 0801 0801 0825 0833 0841 08457 0857 0855 0873 0881 0889 0897	0474 0482 0490 0490 0506 2 0770 0778 0786 0794 0802 0810 0818 0826 0834 0858 0826 0858 0858 0858 0858 0858 0858 0858 085	0475 0483 0491 0499 0507 0771 0775 0795 0803 0811 0819 0827 0843 0851 0843 0851 0843 0859 0867 0883 0889 0907	0476 0484 0492 0500 0508 4 0772 0780 0780 0780 0780 0780 0780 0780	0477 0485 0501 0509 773 0789 0797 0805 0813 0821 0843 0853 0853 0853 0885 0893 0901 0909 09017	0478 0486 0494 0502 0510 0714 0782 0790 0798 0798 0798 0798 0798 0798 0798	0485 0405 0501 0511 077 078 079 079 079 079 080 081 082 083 084 085 086 086 086 086 086 086 087 088 086 086 087 095 095
0730 0740 0750 0750 0750 0770 1400 1410 1420 1430 1430 1430 1510 1550 1550 1550 1550 1550 1550 15	0472 0480 0488 0496 0504 0768 0768 0776 0784 0782 0880 0840 0848 0846 0840 0848 0856 0864 0888 0866 0888 0886 0888	0473 0481 0489 0497 0505 1 0765 0793 0801 0809 0817 0825 0831 0841 0849 0857 0831 0884 0841 0889 0857 0905 0905 0905	0474 0482 0490 0498 0506 0790 0770 0778 0786 0784 0802 0818 0826 0834 0848 0858 0858 0858 0858 0858 0858 085	0475 0483 0491 0499 0507 3 0771 0779 0787 0795 0803 0811 0819 0843 0851 0853 0851 0853 0853 0853 0853 0851 0859 0907 0915	0476 0484 0492 0500 0508 4 0792 0780 0788 0796 0788 0796 0804 0812 0820 0884 0832 0864 0884 0884 0884 0886 0884 0886 0884 0886 0884 0886 0884 0892	0477 0485 5 5 0703 0781 0789 0797 0785 0813 0821 0825 0845 0845 0845 0845 0845 0845 0845 084	0478 0486 0494 0502 0510 0714 0782 0790 0790 0790 0798 0886 0814 0822 0830 08846 0854 0854 0854 0886 0846 0854 0866 0894 0910 0918 0926	0485 0501 0511 0511 0511 0511 071 078 079 080 079 080 083 084 085 086 087 088 086 087 087 087 087 087 087 087 097 097 097
0730 0740 0750 0750 0770 1400 1410 1410 1420 1430 1440 1430 1440 1510 1520 1550 1550 1550 1550 1550 155	0472 0480 0488 0496 0504 0768 0776 0784 0792 0830 0830 0830 0832 0840 0848 0856 0864 0856 0864 0888 0896 09880 09880 09912 0928 0936	0473 0481 0489 0497 0505 0705 0705 0705 0707 0785 0793 0801 0801 0801 0805 0817 0865 0873 0865 0881 0889 0887 0905	0474 0482 0490 0490 0506 0506 0770 0770 0786 0786 0786 0794 0802 0818 0826 0834 0842 0850 0882 0850 0858 0856 0858 0856 0858 0858 0856 0858 0856 0858 0856 0858 0958 09514 09514 09515 09518	0475 0483 0491 0499 0507 0771 0779 0785 0803 0819 0827 0835 0843 0859 0867 0875 0883 0849 0883 0889 0883 0889	0476 0484 0492 0500 0508 4 0772 0780 0786 0786 0786 0786 0786 0786 0884 0884 0884 0884 0884 0884 0884 08	0477 0485 0493 0501 0509 773 0773 0781 0781 0787 0797 0805 0813 0829 0837 0845 0853 0821 0885 0885 0885 0885 0885 0885 0885 088	0478 0486 0494 0502 0510 0714 0774 0782 0790 0798 0805 0814 0822 0830 08854 0822 0830 08854 08854 08854 08854 0886 0886 0886 0886 0886 0886 0886 088	0487 0495 0503 0511 0511 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
0730 0740 0750 0750 0750 0770 1400 1410 1420 1430 1430 1430 1510 1550 1550 1550 1550 1550 1550 15	0472 0480 0488 0496 0504 0768 0768 0776 0784 0782 0880 0840 0848 0846 0840 0848 0856 0864 0888 0866 0888 0886 0888	0473 0481 0489 0497 0505 1 0765 0793 0801 0809 0817 0825 0831 0841 0849 0857 0831 0884 0841 0889 0857 0905 0905 0905	0474 0482 0490 0498 0506 0790 0770 0778 0786 0784 0802 0818 0826 0834 0848 0858 0858 0858 0858 0858 0858 085	0475 0483 0491 0499 0507 779 0779 0787 0795 0803 0811 0819 0843 0851 0853 0843 0851 0853 0853 0867 0875 0883 0899 0907 0915	0476 0484 0492 0500 0508 4 0792 0780 0788 0796 0788 0796 0804 0812 0820 0884 0832 0864 0884 0884 0884 0886 0884 0886 0884 0886 0884 0886 0884 0892	0477 0485 5 5 0703 0781 0789 0797 0785 0813 0821 0825 0845 0845 0845 0845 0845 0845 0845 084	0478 0486 0494 0502 0510 0714 0782 0790 0790 0798 0886 0814 0822 0830 08846 0854 0854 0854 0886 08846 0854 0886 0886 0886 0886 0886 0886 0886 088	0487 0495 0503 0511 7 0777 0783 0799 080 0819 082 0833 0843 0843 0845 0855 0867 0876

 1700
 0960
 0961
 0962
 0963
 0964
 0965
 0966
 0967

 1710
 0968
 0959
 0970
 0971
 0972
 0973
 0974
 0975

 1720
 0976
 0977
 0978
 0979
 0980
 0981
 0982
 0983

 1730
 0984
 0985
 0986
 0987
 0988
 0980
 0991

 1740
 0992
 0993
 0994
 0995
 0996
 0997
 0988
 0980
 0991

 1750
 1000
 1001
 1002
 1003
 1004
 1005
 1006
 1007

 1760
 1008
 1009
 1010
 1011
 1012
 1013
 1014
 1015

 1770
 1016
 1017
 1018
 1019
 1022
 1023
 1023

95118A.F3

NUMERICAL INFORMATION

OCTAL-DECIMAL INTEGER CONVERSION TABLE

	0	1 -	2	3	4	5	6	7
2000	1024	1025	1026	1027	1028	1029	1030	1031
2010	1032	1033	1034	1035	1036	1037	1038	1039
2020	1040	1041	1042	1043	1044	1045	1046	1047
2030	1048	1049 1057	1050 1058	1051	1052 1060	1053 1061	1054	1055
2040 2050	1056	1065	1056	1067	1068	1069	1070	1071
2060	1072	1073	1074	1075	1076	1077	1078	1079
2070	1080	1081	1082	1083	1084	1085	1086	108
2100	1088	1089	1090	1091	1092	1093	1094	1095
2110	1096	1097 1105	1098 1106	1099 1107	1100	1101	1102 1110	1103
2120 2130	11112	1113	1114	1115	1116	1117	1118	1119
2140	1120	1121	1122	1123	1124	1125	1126	1127
2150	1128	1129	1130	1131	1132	1133	1134	1135
2160	1136	1137 1145	1138 1146	1139 1147	1140 1148	1141 1149	1142 1150	1143
2170	1144							1151
2200 2210	1152 1160	1153 1161	1154 1162	1155 1163	1156 1164	1157 1165	1158 1166	1159
2220	1168	1169	1170	1171	1172	1173	1174	1175
2230	1176	1177	1178	1179	1180	1181	1182	118:
2240	1184	1185	1186	1187	1188	1189	1190	1191
2250	1192	1193	1194	1195	1196	1197	1198	1199
2260 2270	1200	1201 1209	1202 1210	1203	1204	1205	1206 1214	1207
2300	1216	1217	1218	1219	1220	1221	1222	
2310	1224	1225	1226	1227	1228	1229	1222	1223
2320	1232	1233	1234	1235	1236	1237	1238	1239
2330	1240	1241	1242	1243	1244	1245	1246	1247
2340		1249	1250	1251	1252	1253	1254	1255
	1248			1000				
2350	1256	1257	1258	1259	1260	1261	1262	1263
2350 2360 2370				1259 1267 1275	1260 1268 1276	1261 1269 1277	1262 1270 1278	126. 1271 1279
2360	1256 1264	1257 1265	1258 1266	1267	1268	1269	1270	1271
2360 2370 3000	1256 1264 1272 0 1536	1257 1265 1273 1 1 1 537	1258 1266 1274 2 1538	1267 1275 3 1539	1268 1276 4 1540	1269 1277 5 1541	1270 1278 6 1542	1271 1279 7 1543
2360 2370 3000 3010	1256 1264 1272 0 1536 1544	1257 1265 1273 1273 1273	1258 1266 1274 2 1538 1546	1267 1275 3 1539 1547	1268 1276 4 1540 1548	1269 1277 5 1541 1549	1270 1278 6 1542 1550	1271 1279 7 1543 1551
2360 2370 3000 3010 3020	1256 1264 1272 0 1536 1544 1552	1257 1265 1273 1273 1273 1273 1273	1258 1266 1274 2 1538 1546 1554	1267 1275 3 1539 1547 1555	1268 1276 4 1540 1548 1556	1269 1277 5 1541 1549 1557	1270 1278 6 1542 1550 1558	1271 1279 7 1543 1551 1559
2360 2370 3000 3010 3020 3030	1256 1264 1272 0 1536 1544 1552 1560	1257 1265 1273 1 1 1537 1545 1553 1561	1258 1266 1274 2 1538 1546 1554 1562	1267 1275 3 1539 1547 1555 1563	1268 1276 4 1540 1548 1556 1564	1269 1277 5 1541 1549 1557 1565	1270 1278 6 1542 1550 1558 1566	1271 1279 7 1543 1551 1559 1567
2360 2370 3000 3010 3020	1256 1264 1272 0 1536 1544 1552	1257 1265 1273 1 1 1537 1545 1553 1561 1569	1258 1266 1274 2 1538 1546 1554 1562 1570	1267 1275 3 1539 1547 1555 1563 1571	1268 1276 4 1540 1548 1556 1564 1572	1269 1277 5 1541 1549 1557 1565 1573	1270 1278 6 1542 1550 1558 1566 1574	1271 1279 7 1543 1551 1559 1567 1575
2360 2370 3000 3010 3020 3030 3040	1256 1264 1272 0 1536 1544 1552 1560 1568	1257 1265 1273 1 1 1537 1545 1553 1561	1258 1266 1274 2 1538 1546 1554 1562	1267 1275 3 1539 1547 1555 1563	1268 1276 4 1540 1548 1556 1564	1269 1277 5 1541 1549 1557 1565	1270 1278 6 1542 1550 1558 1566	1271 1279 7 1543 1551 1559 1567
2360 2370 3000 3010 3020 3030 3040 3050	1256 1264 1272 0 1536 1544 1552 1560 1568 1576	1257 1265 1273 1 1537 1545 1553 1561 1569 1577	1258 1266 1274 2 1538 1546 1554 1562 1570 1578	1267 1275 3 1539 1547 1555 1563 1571 1579	1268 1276 4 1540 1548 1556 1564 1572 1580	1269 1277 5 1541 1549 1557 1565 1573 1581	1270 1278 6 1542 1550 1558 1566 1574 1582	1271 1279 7 1543 1551 1559 1567 1575 1583 1591
2360 2370 3000 3010 3020 3030 3040 3050 3050 3050 3070 3100	1256 1264 1272 0 1536 1544 1552 1560 1568 1576 1584 1592 1600	1257 1265 1273 1 1537 1545 1553 1561 1569 1577 1585 1593 1601	1258 1266 1274 2 1538 1546 1554 1562 1570 1578 1586 1594 1602	1267 1275 3 1539 1547 1555 1563 1571 1575 1587 1595 1603	1268 1276 4 1540 1548 1556 1564 1572 1580 1588 1596 1604	1269 1277 5 1541 1549 1557 1565 1573 1581 1589 1597 1605	1270 1278 6 1542 1550 1558 1556 1574 1582 1590 1598 1606	1271 1279 7 1543 1551 1559 1567 1575 1583 1591 1599 1607
2360 2370 3000 3010 3020 3030 3040 3050 3060 3070 3100 3110	1256 1264 1272 0 1536 1544 1552 1560 1568 1576 1584 1592 1600 1608	1257 1265 1273 1273 1573 1553 1553 1561 1569 1577 1585 1593 1601 1609	1258 1266 1274 2 1538 1546 1554 1562 1570 1578 1586 1594 1602 1610	1267 1275 3 1539 1547 1555 1563 1571 1575 1587 1595 1603 1611	1268 1276 4 1540 1548 1556 1564 1572 1580 1588 1596 1604 1612	1269 1277 5 1541 1549 1557 1565 1573 1581 1589 1597 1605 1613	1270 1278 6 1542 1550 1558 1566 1574 1582 1590 1598 1606 1614	1271 1279 7 1543 1551 1559 1567 1575 1583 1599 1607 1615
2360 2370 3000 3010 3020 3030 3040 3050 3060 3070 3100 3110 3120	1256 1264 1272 0 1536 1544 1552 1560 1568 1576 1584 1592 1600 1608 1616	1257 1265 1273 1273 1537 1545 1553 1561 1569 1577 1585 1593 1601 1609 1617	1258 1266 1274 2 1538 1546 1554 1562 1570 1578 1586 1594 1602 1610 1618	1267 1275 3 1539 1547 1555 1563 1571 1579 1587 1595 1603 1611 1619	1268 1276 4 1540 1548 1556 1564 1572 1580 1588 1596 1604 1612 1620	1269 1277 5 1541 1549 1557 1565 1573 1581 1589 1597 1605 1613 1621	1270 1278 6 1542 1550 1558 1566 1574 1582 1590 1598 1606 1614 1622	1271 1273 7 1543 1551 1559 1567 1575 1583 1599 1607 1615 1623
2360 2370 3000 3010 3020 3030 3040 3050 3060 3070 3110	1256 1264 1272 0 1536 1544 1552 1560 1568 1576 1584 1592 1600 1608	1257 1265 1273 1545 1553 1561 1565 1577 1585 1593 1601 1609 1617 1625	1258 1266 1274 2 1538 1546 1554 1562 1570 1578 1586 1594 1602 1610 1618 1626	1267 1275 3 1539 1547 1555 1563 1579 1587 1595 1603 1611 1619 1627	1268 1276 1276 1540 1548 1556 1564 1572 1580 1588 1596 1604 1612 1628	1269 1277 5 1541 1549 1557 1565 1573 1581 1589 1597 1605 1613 1621 1629	1270 1278 6 1542 1550 1558 1564 1574 1582 1590 1598 1606 1614 1622 1630	1271 1273 7 1543 1551 1559 1567 1575 1583 1591 1599 1607 1615 1623 1631
2360 2370 3000 3010 3020 3030 3040 3050 3050 3050 3100 3110 3120 3130	1256 1264 1272 0 1536 1544 1552 1560 1568 1576 1584 1592 1600 1608 1616 1624	1257 1265 1273 1273 1537 1545 1553 1561 1569 1577 1585 1593 1601 1609 1617	1258 1266 1274 2 1538 1546 1554 1562 1570 1578 1586 1594 1602 1610 1618	1267 1275 3 1539 1547 1555 1563 1571 1579 1587 1595 1603 1611 1619	1268 1276 4 1540 1548 1556 1564 1572 1580 1588 1596 1604 1612 1620	1269 1277 5 1541 1549 1557 1565 1573 1581 1589 1597 1605 1613 1621 1629	1270 1278 6 1542 1550 1558 1566 1574 1582 1590 1598 1606 1614 1622	1271 1273 7 1543 1551 1559 1567 1575 1583 1599 1607 1615 1623
2360 2370 3000 3010 3020 3030 3040 3050 3050 3100 3110 3110 3130 3140 3150 3150	1256 1264 1272 1536 1544 1572 1560 1568 1576 1584 1576 1584 1592 1600 1608 1616 1624 1632 1640	1257 1265 1273 1545 1553 1561 1553 1561 1573 1585 1593 1601 1609 1617 1625 1633 1641	1258 1266 1274 2 1538 1546 1554 1554 1554 1552 1570 1578 1586 1594 1602 1610 1618 1626 1634 1642	1267 1275 3 1539 1547 1555 1563 1571 1579 1587 1595 1603 1611 1619 1627 1633 1643 1643	1268 1276 4 1540 1548 1556 1564 1572 1580 1588 1596 1604 1612 1620 1628 1636 1634 1644	1269 1277 5 1541 1549 1557 1565 1573 1581 1589 1597 1605 1613 1621 1625 1645	1270 1278 6 1542 1550 1558 1566 1574 1582 1590 1598 1606 1614 1622 1630 1638	1271 1273 7 1543 1559 1567 1575 1583 1591 1599 1607 1615 1623 1631 1639 1647 1655
2360 2370 3000 3010 3020 3030 3050 3050 3050 3050 3050 305	1256 1264 1272 0 1536 1544 1552 1560 1568 1576 1588 1576 1588 1576 1600 1608 1616 1622 1640	1257 1265 1273 1 1537 1545 1553 1561 1569 1577 1585 1593 1601 1609 1617 1625 1633 1641	1258 1266 1274 2 1538 1546 1554 1552 1570 1578 1586 1594 1602 1610 1618 1622 1634	1267 1275 3 1539 1547 1555 1563 1571 1579 1587 1595 1603 1611 1619 1625 1643	1268 1276 4 1540 1548 1556 1564 1572 1580 1588 1596 1604 1612 1628 1636 1636 1636	1269 1277 5 1541 1549 1557 1565 1573 1581 1587 1605 1613 1629 1637 1645	1270 1278 6 1542 1550 1558 1566 1574 1582 1590 1598 1606 1614 1622 1638 1648	1271 1273 7 1543 1559 1567 1575 1583 1591 1599 1607 1615 1623 1631 1639 1647 1655
2360 2370 3000 3010 3020 3030 3040 3050 3050 3050 3050 3050 305	1256 1264 1272 0 1536 1544 1552 1560 1584 1592 1600 1608 1616 1624 1648 1656 1648	1257 1265 1273 1 1537 1545 1553 1561 1569 1577 1585 1593 1601 1607 1665 1631 1649 1657	1258 1266 1274 2 1538 1546 1554 1554 1554 1570 1578 1586 1594 1602 1618 1628 1638 1658 1658	1267 1275 3 1539 1547 1555 1563 1571 1575 1603 1611 1619 1627 1633 1651 1659 1667	1268 1276 4 1540 1548 1556 1564 1564 1572 1580 1588 1596 1604 1612 1628 1636 1644 1652 1660	1269 1277 5 1541 1549 1557 1565 1573 1581 1585 1597 1605 1613 1621 1645 1653 1645 1645	1270 1278 6 1542 1550 1558 1566 1574 1582 1598 1606 1578 1698 1638 1646 1634 1646 1634	1271 1273 7 1543 1559 1567 1575 1583 1599 1607 1615 1623 1631 1639 1647 1655 1663
2360 2370 3000 3010 3020 3030 3040 3050 3060 3100 31100 3120 3120 3120 3120 3120 31	1256 1264 1272 0 1536 1544 1552 1560 1568 1576 1584 1592 1600 1608 1616 1622 1640 1648 1656	1257 1265 1273 1553 1553 1561 1569 1601 1609 1617 1625 1633 1641 1649 1657	1258 1266 1274 2 1538 1546 1554 1554 1554 1554 1554 1558 1578 1578 1578 1578 1578 1578 1578	1267 1275 3 1539 1547 1555 1563 1611 1571 1575 1603 1611 1619 1627 1643 1651 1659	1268 1276 4 1540 1548 1556 1564 1572 1588 1596 1604 1612 1620 1628 1628 1628 1628 1644 1652 1660	1269 1277 5 1541 1549 1557 1565 1573 1581 1589 1597 1605 1613 1629 1637 1645 1653 1661	1270 1278 6 1542 1550 1558 1566 1574 1582 1598 1606 1614 1622 1630 1638 1646 1634 1662 1657 1657	1271 1275 7 1543 1551 1559 1567 1575 1583 1591 1699 1607 1615 1647 1655 1663 1671 1679
2360 2370 3000 3010 3020 3020 3020 3020 3020 302	1256 1264 1272 0 1536 1544 1552 1560 1568 1576 1584 1592 1600 1608 1616 1624 1632 1640 1648 1656	1257 1265 1273 1 1557 1545 1553 1569 1577 1585 1593 1601 1609 1617 1625 1633 1641 1649 1657 1649	1258 1266 1274 2 1538 1546 1554 1546 1554 1556 1570 1570 1570 1570 1570 1570 1610 1618 1646 1634 1645 1650 1658	1267 1275 3 1539 1547 1555 1563 1579 1579 1587 1603 1611 1619 1627 1635 1643 1659 1667 1675	1268 1276 1540 1548 1556 1564 1576 1580 1588 1596 1604 1612 1628 1636 1644 1652 1668 1666 1668	1269 1277 5 1541 1549 1557 1565 1573 1581 1585 1597 1605 1613 1621 1645 1653 1645 1645	1270 1278 6 1542 1550 1558 1566 1574 1582 1598 1606 1578 1698 1638 1646 1634 1646 1634	1271 1273 7 1543 1559 1567 1575 1583 1599 1607 1615 1623 1631 1639 1647 1655 1663
2360 2370 3000 3010 3020 3030 3040 3050 3050 3050 3050 3050 305	1256 1264 1272 1536 1544 1552 1560 1568 1576 1584 1576 1600 1608 1616 1624 1632 1640 1648 1656 1664 1652 1660 1688 1696	1257 1265 1273 1573 1573 1545 1553 1561 1569 1577 1585 1593 1601 1609 1617 1625 1633 1641 1649 1657 1655 1653 1665 1667	1258 1266 1274 2 1538 1546 1554 1552 1570 1578 1586 1594 1602 1618 1626 1634 1642 1658 1658 1658 1658	1267 1275 3 1539 1547 1555 1563 1671 1579 1587 1603 1651 1635 1643 1651 1659 1667 1683 1691	1268 1276 1276 1540 1548 1556 1564 1556 1560 1588 1596 1604 1612 1628 1636 1644 1652 1660 1648 1652 1660	1269 1277 5 1541 1549 1557 1565 1573 1581 1587 1605 1613 1621 1669 1637 1645 1653 1661 1669 1677 1685 1693	1270 1278 6 1542 1550 1558 1566 1614 1622 1630 1638 1646 1638 1646 1634 1662 1670 1678 1662 1670	1271 1277 7 1543 1551 1557 1583 1591 1615 1623 1631 1647 1655 1663 1647 1655 1663
2360 2370 3000 3010 3020 3030 3040 3050 3050 3050 3050 3050 305	1256 1264 1272 1272 1572 1576 154 1576 1584 1592 1600 1608 1654 1652 1640 1648 1656 1664 1656 1664 1656	1257 1265 1273 1573 1545 1553 1561 1553 1561 1577 1585 1593 1601 1607 1665 1633 1641 1649 1657 1665 1673 1681 1689 1697 1705	1258 1266 1274 2 1538 1546 1554 1554 1554 1578 1578 1578 1578 1578 1578 1578 1578	1267 1275 3 1539 1547 1555 1563 1571 1575 1603 1661 1655 1643 1655 1663 1655 1663 1659	1268 1276 1540 1548 1556 1564 1572 1588 1596 1604 1612 1620 1628 1636 1644 1652 1660 1668 1676 1676 1678 1679 1708	1269 1277 5 1541 1549 1557 1565 1573 1581 1573 1581 1573 1581 1573 1587 1605 1663 1661 1663 1661 1667 1665 1667 1667 1677 1685 1697	1270 1278 6 1542 1550 1558 1566 1574 1582 1590 1598 1606 1614 1622 1630 1638 1646 1634 1662 1678 1678 1678 1678	1271 1273 1543 1551 1559 1567 1575 1583 1591 1699 1667 1615 1663 1671 1663 1671 1663 1671 1663 1671 1679
2360 2370 3000 3010 3020 3030 3040 3050 3050 3050 3050 3050 305	1256 1264 1272 1536 1544 1552 1560 1568 1576 1584 1576 1600 1608 1616 1624 1632 1640 1648 1656 1664 1656 1668	1257 1265 1273 1573 1573 1545 1553 1561 1569 1577 1585 1593 1601 1609 1617 1625 1633 1641 1649 1657 1655 1653 1665 1667	1258 1266 1274 2 1538 1546 1554 1552 1570 1578 1586 1594 1602 1610 1618 1626 1634 1642 1658 1658 1658 1658	1267 1275 3 1539 1547 1555 1563 1671 1579 1587 1603 1651 1635 1643 1651 1659 1667 1683 1691	1268 1276 1276 1540 1548 1556 1564 1556 1560 1588 1596 1604 1612 1628 1636 1644 1652 1660 1648 1652 1660	1269 1277 5 1541 1549 1557 1565 1573 1581 1587 1605 1613 1621 1669 1637 1645 1653 1661 1669 1677 1685 1693	1270 1278 6 1542 1550 1558 1566 1614 1622 1630 1638 1646 1638 1646 1634 1662 1670 1678 1662 1670	1271 1273 1543 1551 1559 1567 1575 1583 1591 1699 1667 1615 1663 1671 1663 1671 1663 1671 1663 1671 1679
2360 2370 3000 3010 3020 3030 3050 3040 3050 3040 3050 3100 3110 3120 3120 3150 3150 3150 3150 3150 3150 3220 3220 3220	12564 1264 1272 1272 1536 1544 1552 1560 1588 1576 1588 1576 1588 1576 1600 1608 1616 1622 1640 1648 1656 1664 1656 1668 1658 1688 1698 1698	1257 1265 1273 1 1537 1545 1553 1561 1569 1577 1585 1633 1641 1649 1657 1649 1657 1649 1657 1649 1657 1713 1721	1258 1266 1274 2 1538 1546 1554 1554 1556 1557 1570 1578 1580 1610 1618 1630 1658 1666 1634 1650 1658 1666 1674 1690 1698 1706 1714 1722	1267 1275 3 1539 1547 1555 1663 1571 1587 1595 1603 1611 1619 1643 1651 1643 1651 1643 1651 1643 1643 1643 1707 1715	1268 1276 1276 1540 1548 1556 1564 1572 1588 1596 1604 1612 1628 1634 1632 1660 1664 1652 1660 1664 1664 1664 1708 1708	1269 1277 5 1541 1549 1557 1565 1573 1589 1597 1605 1613 1621 1629 1637 1645 1653 1661 1669 1677 1685 1693 1709 1717 1725	1270 1278 6 1542 1550 1558 1566 1574 1590 1598 1606 1614 1622 1630 1638 1646 1634 1662 1678 1668 1664 1710 1718	1271 1273 1543 1551 1559 1567 1575 1583 1599 1607 1615 1663 1639 1647 1655 1663 1679 1687 1697 1687 1697 1703
2360 2370 3010 3020 3030 3030 3040 3050 3050 3050 3050 305	12564 12674 1272 1272 1536 1534 1552 1560 1584 1552 1560 1608 1616 1664 1672 1640 1648 1656 1664 1672 1720	1257 1265 1273 1 1537 1545 1553 1561 1569 1577 1585 1593 1601 1609 1617 1625 1673 1641 1649 1657 1665 1673 1689 1689 1697 1705	1258 1266 1274 2 1538 1546 1554 1554 1554 1554 1554 1554 1554	1267 1275 3 1539 1547 1555 1603 1611 1619 1627 1635 1643 1651 1659 1667 1675 1683 1691 1699 1707	1268 1276 4 1540 1548 1556 1564 1554 1554 1556 1558 1556 1604 1612 1628 1636 1644 1652 1660 1668 1676 1668 1676 1692 1700 1708	1269 1277 5 1541 1549 1557 1565 1573 1581 1589 1597 1605 1613 1629 1637 1645 1653 1661 1669 1677 1685 1653 1701 1709	1270 1278 6 1542 1550 1558 1666 1574 1580 1698 1606 1614 1622 1630 1638 1646 1634 1662 1670 1678 1684 1684 1702 1710	1271 1273 1543 1559 1567 1575 1591 1695 1639 1639 1639 1639 1639 1647 1655 1663 1663 1679 1687 1687 1687 1703
2360 2370 3010 3020 3030 3030 3040 3050 3050 3110 3120 3140 3150 3140 3150 3140 3150 3140 3150 3140 3150 3220 3220 3220 3220 3220 3220 3250 325	12564 12674 1272 1272 1572 1570 1534 1554 1554 1554 1554 1554 1555 1568 1576 1588 1576 1608 1616 1632 1640 1656 1668 1656 1668 1672 1688 1698 1704	1257 1265 1273 1 1537 1545 1553 1561 1569 1577 1585 1633 1601 1649 1657 1665 1673 1649 1657 1673 1649 1657 1705 1773 1775	1258 1266 1274 2 1538 1546 1554 1554 1554 1554 1570 1578 1589 1602 1610 1618 1666 1634 1642 1650 1658 1666 1674 1709 1774	1267 1275 3 1539 1547 1555 1663 1571 1587 1595 1603 1611 1627 1633 1643 1651 1659 1667 1675 1683 1691 1683 1691 1707 1715	1268 1276 1540 1540 1546 1556 1564 1576 1580 1588 1596 1604 1612 1620 1628 1664 1652 1666 1664 1652 1700 1708 1716 1724	1269 1277 5 5 1541 1547 1557 1565 1573 1605 1613 1621 1663 1661 1669 1677 1685 1653 1701 1707 1717 1725	1270 1278 6 1542 1550 1558 1566 1574 1582 1598 1606 1638 1644 1662 1638 1646 1654 1662 1670 1678 1686 1662 1667 1712 1712 1714	1271 1273 1543 1551 1559 1567 1575 1583 1591 1695 1667 1655 1663 1647 1655 1663 1647 1667 1703
2360 2370 3010 3010 3020 3030 3050 3050 3050 3100 3110 3120 3130 3140 3150 3140 3150 3140 3150 3210 3220 3220 3220 3220 3220 3220 322	12564 1264 1272 1272 1572 1570 1576 1584 1592 1568 1576 1584 1592 1600 1648 1616 1622 1640 1648 1656 1664 1657 1688 1696 1771 1712 1728	1257 1265 1273 1 1537 1545 1553 1561 1553 1569 1577 1585 1593 1601 1609 1617 1625 1633 1641 1649 1657 1665 1673 1689 1697 1705 1773 1775	1258 1266 1274 2 1538 1546 1554 1554 1554 1554 1554 1554 1554	1267 1275 3 1539 1547 1555 1563 1571 1575 1579 1587 1579 1587 1579 1603 1611 1619 1627 1663 1651 1655 1663 1651 1675 1683 1771 1773 1771 1773	1268 1276 1540 1540 1546 1556 1556 1556 1556 1580 1588 1596 1604 1612 1620 1628 1669 1668 1668 1668 1668 1666 1668 1676 1682 1669 1700 1708 1716 1772	1269 1277 5 5 1541 1547 1557 1565 1573 1581 1589 1597 1605 1613 16613 1662 1653 1663 16645 1667 16645 1667 1669 1667 1669 1701 1717 1725	1270 1278 6 1542 1558 1558 1566 1574 1582 1598 1606 1614 1622 1638 1646 1638 1646 1638 1646 1638 1646 1638 1662 1670 1718 1726 1772 1770	1271 1273 1273 1543 1551 1557 1567 1575 1583 1631 1639 1647 1655 1663 1671 1679 1685 1663 1711 1779 1775 1743 1751
2360 2370 3010 3020 3030 3030 3040 3050 3050 3050 3050 3100 3120 3120 3120 3120 3120 3120 3220 32	12564 12674 1272 1272 1572 1544 1552 1560 1584 1576 1584 1552 1600 1608 1616 1624 1632 1640 1648 1656 1664 1672 1720 1728 1770 1772 1770	1257 1265 1273 1 1537 1545 1553 1561 1569 1577 1585 1601 1609 1617 1681 1649 1657 1663 16641 1669 1673 1661 1689 1773 1771 1773 1773	1258 1266 1274 2 1538 1546 1554 1554 1554 1570 1578 1570 1578 1602 1618 1638 1636 1633 1650 1658 1662 1650 1658 1662 1650 1658 1662 1706 1774 1772	1267 1275 3 1539 1547 1555 1603 1571 1587 1595 1603 1611 1619 1643 1651 1675 1683 1699 1707 1715 1723 1731 1739	1268 1276 1540 1548 1556 1564 1576 1588 1596 1604 1612 1620 1628 1634 1632 1660 1668 1634 1652 1660 1668 1664 1676 1664 1700 1708 1716	1269 1277 5 1541 1549 1557 1565 1573 1565 1573 1589 1597 1605 1613 1629 1645 1653 1661 1669 1645 16653 1661 1709 1717 1709 1717 1725	1270 1278 6 1542 1550 1558 1566 1574 1558 1576 1578 1582 1590 1598 1606 1614 1622 1630 1638 1646 1634 1662 1702 1710 1718 1742 1742 1750	1271 1273 1273 7 1543 1559 1559 1607 1575 1583 1631 1623 1631 1639 1647 1655 1663 1679 1667 1679 1667 1679 1667 1707 1711 1719 1777 1735
2360 2370 3010 3010 3020 3030 3050 3050 3050 3100 3110 3120 3130 3140 3150 3140 3150 3140 3150 3210 3220 3220 3220 3220 3220 3220 322	12564 1264 1267 1272 1572 1572 1560 1554 1552 1568 1558 1558 1558 1558 1558 1558 1558	1257 1265 1273 1 1537 1545 1553 1561 1553 1569 1577 1585 1593 1601 1609 1617 1625 1633 1641 1649 1657 1665 1673 1689 1697 1705 1773 1775	1258 1266 1274 2 1538 1546 1554 1554 1554 1554 1554 1554 1554	1267 1275 3 1539 1547 1555 1563 1571 1575 1579 1587 1579 1587 1579 1603 1611 1619 1627 1663 1651 1655 1663 1651 1675 1683 1771 1773 1771 1773	1268 1276 1540 1540 1546 1556 1556 1556 1556 1580 1588 1596 1604 1612 1620 1628 1669 1668 1668 1668 1668 1666 1668 1676 1682 1669 1700 1708 1716 1772	1269 1277 5 5 1541 1547 1557 1565 1573 1581 1589 1597 1605 1613 16613 1662 1653 1663 16645 1667 16645 1667 1669 1667 1669 1701 1717 1725	1270 1278 6 1542 1558 1558 1566 1574 1582 1598 1606 1614 1622 1638 1646 1638 1646 1638 1646 1638 1646 1638 1662 1670 1718 1726 1772 1770	1271 1273 1273 1543 1551 1557 1567 1575 1583 1631 1639 1647 1655 1663 1671 1679 1685 1663 1711 1779 1775 1743 1751

		-							
	0	1	2	3	4	5	6	7	
2400	1280			1283					2000 1024
				1291 1299					
2430	1304	1305	1306	1307	1308	1309	1310	1311	(Octal) (Decimal)
2440	1312	1313	1314	1315	1316	1317	1318	1319	
				1323 1331				1327	Out-I Desirel
				1339					Octal Decimal 10000 - 4096
		1 2 4 5	1240	1947	1 2 4 0	1240	1960	1 25 1	20000 9192
				1347 1355					30000 - 12288
2520	1360	1361	1362	1363	1364	1365	1366	1367	40000 - 16384
2530	1368	1369	1370	1371 1379	1372	1373	1374	1375	
				1379					70000 - 28672
2560	1392	1393	1394	1395	1396	1397	1398	1399	
2570	1400	1401	1402	1403	1404	1405	1406	1407	
2600	1408	1409	1410	1411	1412	1413	1414	1415	
				1419					
				1427 1435					
				1443					
				1451				1455	
				1459 1467				1463	
				1475					
	1480			1483 1491					
				1499				1503	
				1507					
				1515 1523				1519	
				1531					
	0	1	2	3	4	5	6	7	
3400									3000 1536
3410	1792 1800	1793 1801	1794 1802	1795 1803	1796 1804	1797 1805	1798 1806	1799 1807	to to 3777 2047
3410	1792 1800 1808	1793 1801 1809	1794 1802 1810	1795 1803 1811	1796 1804 1812	1797 1805 1813	1798 1806 1814	1799 1807 1815	to to 3777 2047
3410 3420 3430	1792 1800 1808 1816	1793 1801 1809 1817	1794 1802 1810 1818	1795 1803	1796 1804 1812 1820	1797 1805 1813 1821	1798 1806 1814 1822	1799 1807 1815 1823	to to 3777 2047
3410 3420 3430 3440 3450	1792 1800 1808 1816 1824 1832	1793 1801 1809 1817 1825 1833	1794 1802 1810 1818 1826 1834	1795 1803 1811 1819 1827 1825	1796 1804 1812 1820 1828 1828	1797 1805 1813 1821 1829 1837	1798 1806 1814 1822 1830 1838	1799 1807 1815 1823 1831 1831	to to 3777 2047
3410 3420 3430 3440 3450 3460	1792 1800 1808 1816 1824 1832 1840	1793 1801 1809 1817 1825 1833 1841	1794 1802 1810 1818 1826 1834 1834	1795 1803 1811 1819 1827 1835 1843	1796 1804 1812 1820 1828 1836 1836	1797 1805 1813 1821 1829 1837 1845	1798 1806 1814 1822 1830 1838 1846	1799 1807 1815 1823 1831 1839 1847	to to 3777 2047
3410 3420 3430 3440 3450 3460 3470	1792 1800 1808 1816 1824 1832 1840 1848	1793 1801 1809 1817 1825 1833 1841 1849	1794 1802 1810 1818 1826 1834 1842 1850	1795 1803 1811 1819 1827 1835 1843 1851	1796 1804 1812 1820 1828 1836 1844 1852	1797 1805 1813 1821 1829 1837 1845 1853	1798 1806 1814 1822 1830 1838 1846 1854	1799 1807 1815 1823 1831 1839 1847 1855	to to 3777 2047
3410 3420 3430 3440 3450 3460 3470 3500	1792 1800 1808 1816 1824 1832 1840 1848 1856	1793 1801 1809 1817 1825 1833 1841 1849 1857	1794 1802 1810 1818 1826 1834 1842 1850 1858	1795 1803 1811 1819 1827 1835 1843 1851 1859	1796 1804 1812 1820 1828 1836 1844 1852 1860	1797 1805 1813 1821 1829 1837 1845 1853 1861	1798 1806 1814 1822 1830 1838 1846 1854 1862	1799 1807 1815 1823 1831 1839 1847 1855 1853	to to 3777 2047
3410 3420 3430 3440 3450 3460 3470 3500 3510	1792 1800 1808 1816 1824 1832 1840 1848 1856 1864	1793 1801 1809 1817 1825 1833 1841 1849 1857 1865	1794 1802 1810 1818 1826 1834 1842 1850 1858 1866	1795 1803 1811 1819 1827 1835 1843 1851 1859 1867	1796 1804 1812 1820 1828 1836 1844 1852 1860 1868	1797 1805 1813 1821 1829 1837 1845 1853 1861 1861	1798 1806 1814 1822 1830 1838 1846 1854 1862 1870	1799 1807 1815 1823 1831 1839 1847 1855 1863 1871	to to 3777 2047
3410 3420 3430 3440 3450 3460 3470 3500 3510 3520 3530	1792 1800 1808 1816 1824 1832 1840 1848 1856 1864 1872 1880	1793 1801 1809 1817 1825 1833 1841 1849 1857 1865 1873 1881	1794 1802 1810 1818 1826 1834 1842 1850 1858 1866 1874 1882	1795 1803 1811 1819 1827 1835 1843 1851 1859 1867 1875 1883	1796 1804 1812 1820 1828 1836 1844 1852 1860 1868 1876 1884	1797 1805 1813 1821 1829 1837 1845 1853 1861 1869 1877 1885	1798 1806 1814 1822 1830 1838 1846 1854 1862 1870 1878 1886	1799 1807 1815 1823 1831 1839 1847 1855 1863 1871 1879 1887	to to 3777 2047
3410 3420 3430 3440 3450 3460 3470 3500 3510 3520 3530 3540	1792 1800 1808 1816 1824 1832 1840 1848 1856 1864 1872 1880 1888	1793 1801 1809 1817 1825 1833 1841 1849 1857 1865 1873 1881 1889	1794 1802 1810 1818 1826 1834 1842 1850 1858 1866 1874 1882 1890	1795 1803 1811 1819 1827 1835 1843 1851 1859 1867 1875 1883 1891	1796 1804 1812 1820 1828 1836 1844 1852 1860 1868 1876 1868 1876 1884 1892	1797 1805 1813 1821 1829 1837 1845 1853 1861 1869 1877 1885 1893	1798 1806 1814 1822 1830 1838 1846 1854 1862 1870 1878 1878 1878 1894	1799 1807 1815 1823 1831 1839 1847 1855 1863 1871 1879 1887 1895	to to 3777 2047
3410 3420 3430 3440 3450 3460 3460 3470 3510 3520 3530 3540 3550	1792 1800 1808 1816 1824 1832 1840 1848 1856 1864 1872 1880 1888 1896	1793 1801 1809 1817 1825 1833 1841 1849 1857 1865 1873 1881 1889 1897	1794 1802 1810 1818 1826 1834 1842 1850 1858 1866 1874 1882 1890 1898	1795 1803 1811 1819 1827 1835 1843 1851 1859 1867 1875 1883 1891 1899	1796 1804 1812 1820 1828 1836 1844 1852 1860 1868 1876 1884 1892 1900	1797 1805 1813 1821 1829 1837 1845 1853 1861 1869 1877 1885 1893 1901	1798 1806 1814 1822 1830 1838 1846 1854 1862 1870 1878 1878 1878 1894	1799 1807 1815 1823 1831 1839 1847 1855 1863 1871 1879 1887 1895 1903	to to 3777 2047
3410 3420 3430 3440 3450 3460 3470 3500 3510 3520 3530 3540 3550 3550 3560	1792 1800 1808 1816 1824 1832 1840 1848 1856 1864 1856 1864 1872 1880 1888 1896 1904	1793 1801 1809 1817 1825 1833 1841 1849 1857 1865 1873 1881 1889 1897 1905	1794 1802 1810 1818 1826 1834 1842 1850 1858 1856 1874 1882 1890 1898 1906	1795 1803 1811 1819 1827 1835 1843 1851 1859 1867 1875 1883 1891 1899 1907	1796 1804 1812 1820 1828 1836 1844 1852 1860 1868 1876 1884 1892 1900 1908	1797 1805 1813 1821 1829 1837 1845 1853 1861 1869 1877 1885 1893 1901 1909	1798 1806 1814 1822 1830 1838 1846 1854 1862 1878 1866 1878 1886 1894 1902 1910	1799 1807 1815 1823 1831 1839 1847 1855 1863 1871 1879 1887 1895 1903	to to 3777 2047
3410 3420 3430 3440 3450 3460 3470 3510 3520 3530 3540 3550 3560 3570	1792 1800 1808 1816 1824 1840 1848 1856 1864 1872 1880 1888 1896 1904 1912	1793 1801 1809 1817 1825 1833 1841 1849 1857 1865 1873 1881 1889 1897 1905 1913	1794 1802 1810 1818 1826 1834 1842 1850 1858 1866 1874 1882 1890 1898 1906 1914	1795 1803 1811 1819 1827 1835 1843 1851 1859 1867 1875 1883 1891 1899 1907 1915	1796 1804 1812 1820 1828 1836 1844 1852 1860 1868 1876 1884 1892 1900 1908 1916	1797 1805 1813 1821 1829 1837 1845 1853 1861 1869 1877 1885 1893 1901 1909 1917	1798 1806 1814 1822 1830 1838 1846 1854 1862 1870 1878 1886 1894 1902 1910 1918	1799 1807 1815 1823 1831 1839 1847 1855 1863 1871 1879 1887 1895 1903 1911 1919	to to 3777 2047
3410 3420 3430 3440 3440 3460 3460 3500 3500 3550 3550 3550 3550 3550 35	1792 1800 1808 1816 1824 1832 1840 1848 1856 1864 1872 1880 1888 1896 1904 1912 1920	1793 1801 1809 1817 1825 1833 1841 1849 1857 1865 1873 1881 1889 1897 1905 1913	1794 1802 1810 1818 1826 1834 1842 1850 1858 1866 1874 1882 1890 1898 1906 1914	1795 1803 1811 1819 1827 1835 1843 1851 1859 1867 1875 1883 1891 1899 1907	1796 1804 1812 1820 1828 1836 1844 1852 1860 1868 1876 1884 1892 1900 1908 1916	1797 1805 1813 1821 1829 1837 1845 1853 1861 1869 1877 1885 1893 1901 1909 1917 1925	1798 1806 1814 1822 1830 1838 1846 1854 1862 1870 1878 1886 1894 1902 1910 1918	1799 1807 1815 1823 1831 1839 1847 1855 1863 1871 1879 1887 1895 1903 1911 1919	to to 3777 2047
3410 3420 3430 3440 3450 3460 3470 3510 3520 3530 3540 3550 3560 3560 3560 3560 3600 3610 3620	1792 1800 1808 1816 1824 1832 1840 1848 1856 1864 1872 1880 1888 1896 1904 1912 1920 1928 1936	1793 1801 1809 1817 1825 1833 1841 1849 1857 1865 1873 1881 1889 1897 1905 1913 1921 1929 1937	1794 1802 1810 1818 1826 1834 1842 1850 1858 1866 1874 1882 1890 1898 1906 1914 1922 1930	1795 1803 1811 1819 1827 1835 1843 1851 1859 1867 1875 1883 1891 1899 1907 1915 1923 1931 1939	1796 1804 1812 1820 1828 1836 1844 1852 1860 1868 1876 1868 1876 1884 1892 1900 1908 1916 1924 1940	1797 1805 1813 1821 1829 1837 1845 1853 1861 1869 1877 1885 1893 1901 1909 1917 1925 1933 1941	1798 1806 1814 1822 1830 1838 1846 1854 1862 1870 1878 1886 1894 1900 1918 1926 1934 1942	1799 1807 1815 1823 1839 1847 1855 1863 1871 1879 1887 1803 1911 1919 1927 1935 1943	to to 3777 2047
3410 3420 3430 3450 3460 3470 3510 3520 3530 3550 3550 3550 3550 3550 3560 3570 3600 3610 3620 3640 3640	1792 1800 1808 1816 1824 1840 1848 1856 1864 1872 1880 1888 1896 1904 1912 1920 1928 1936	1793 1801 1809 1817 1825 1833 1841 1849 1857 1865 1873 1881 1889 1897 1905 1913 1921 1929 1937 1945	1794 1802 1810 1818 1826 1834 1842 1850 1858 1866 1874 1892 1936 1914 1922 1930 1938 1946	1795 1803 1811 1819 1827 1835 1843 1851 1859 1867 1875 1883 1891 1915 1923 1931 1939 1947	1796 1804 1812 1820 1828 1836 1844 1852 1860 1868 1876 1892 1900 1908 1916 1924 1932 1940 1948	1797 1805 1813 1821 1829 1837 1845 1853 1861 1869 1877 1885 1893 1901 1909 1917 1925 1933 1941 1941	1798 1806 1814 1822 1830 1838 1846 1854 1862 1878 1894 1902 1910 1918 1926 1934 1942 1950	1799 1807 1815 1823 1831 1839 1847 1855 1863 1871 1879 1895 1903 1911 1919 1927 1935 1943 1951	to to 3777 2047
3410 3420 3430 3440 3450 3460 3510 3520 3550 3550 3550 3550 3560 3560 3560 3630 3610 3620 3630 3630 3650	1792 1800 1808 1816 1824 1848 1856 1848 1856 1848 1872 1880 1888 1896 1904 1912 1920 1928 1936 1944 1952	1793 1801 1809 1817 1825 1833 1841 1849 1857 1865 1873 1881 1873 1881 1873 1897 1913 1995 1913	1794 1802 1810 1818 1826 1834 1850 1858 1850 1858 1874 1882 1890 1914 1922 1930 1938 1946 1954	1795 1803 1811 1819 1827 1835 1843 1851 1859 1867 1883 1891 1907 1915 1923 1931 1939 1947 1955	1796 1804 1812 1820 1828 1836 1844 1852 1860 1868 1876 1884 1876 1988 1916 1900 1908 1916 1924 1932 1940 1948	1797 1805 1813 1821 1829 1837 1845 1853 1861 1869 1885 1893 1901 1909 1917 1925 1933 1941 1949 1957	1798 1806 1814 1822 1830 1838 1846 1854 1854 1854 1854 1854 1902 1910 1918 1926 1934 1942 1958	1799 1807 1815 1823 1831 1833 1847 1855 1863 1871 1879 1887 1903 1903 1911 1919 1927 1935 1943 1951 1959	to to 3777 2047
3410 3420 3430 3440 3450 3460 3510 3520 3550 3550 3550 3550 3550 3550 355	1792 1800 1808 1816 1824 1832 1840 1848 1856 1864 1872 1880 1888 1896 1904 1912 1920 1928 1936 1944 1952 1960	1793 1801 1809 1817 1825 1833 1841 1849 1857 1865 1873 1881 1873 1881 1873 1881 1897 1905 1913 1921 1929 1945 1953 1961	1794 1802 1810 1818 1826 1834 1850 1858 1850 1858 1850 1858 1874 1850 1898 1906 1914 1922 1930 1914 1922 1930 1946 1954	1795 1803 1811 1819 1827 1835 1843 1851 1851 1855 1867 1875 1883 1891 1915 1923 1931 1931 1931 1939 1947 1955 1963 1971	1796 1804 1812 1820 1828 1836 1844 1852 1860 1868 1876 1884 1876 1884 1890 1908 1916 1924 1932 1932 1940 1948 1956 1964	1797 1805 1813 1821 1829 1837 1845 1853 1861 1869 1887 1885 1893 1901 1917 1925 1933 1941 1949 1957 1965 1973	1798 1806 1814 1822 1830 1838 1846 1854 1854 1854 1854 1870 1978 1926 1910 1918 1926 1934 1942 1958 1966	1799 1807 1815 1823 1831 1839 1847 1855 1903 1871 1879 1895 1903 1911 1919 1927 1943 1951 1955 1967	to to 3777 2047
3410 3420 3430 3440 3450 3460 3510 3520 3550 3550 3550 3550 3550 3550 355	1792 1800 1808 1816 1824 1832 1840 1848 1856 1864 1872 1880 1888 1896 1904 1912 1920 1928 1936 1944 1952 1960	1793 1801 1809 1817 1825 1833 1841 1849 1857 1865 1873 1881 1873 1881 1873 1881 1897 1905 1913 1921 1929 1945 1953 1961	1794 1802 1810 1818 1826 1834 1850 1858 1850 1858 1850 1858 1874 1850 1898 1906 1914 1922 1930 1914 1922 1930 1946 1954	1795 1803 1811 1819 1827 1835 1843 1851 1859 1867 1883 1891 1907 1915 1923 1931 1939 1947 1955	1796 1804 1812 1820 1828 1836 1844 1852 1860 1868 1876 1884 1876 1884 1890 1908 1916 1924 1932 1932 1940 1948 1956 1964	1797 1805 1813 1821 1829 1837 1845 1853 1861 1869 1887 1885 1893 1901 1917 1925 1933 1941 1949 1957 1965 1973	1798 1806 1814 1822 1830 1838 1846 1854 1854 1854 1854 1870 1978 1926 1910 1918 1926 1934 1942 1958 1966	1799 1807 1815 1823 1831 1833 1847 1855 1863 1871 1879 1887 1903 1903 1911 1919 1927 1935 1943 1951 1959	to to 3777 2047
3410 3420 3430 3450 3500 3510 3520 3530 3550 3550 3550 3550 3550 355	1792 1800 1808 1816 1824 1840 1848 1856 1864 1872 1880 1904 1912 1920 1928 1936 1944 1952 1960 1968 1976	1793 1801 1809 1817 1825 1833 1841 1849 1857 1865 1873 1889 1897 1905 1913 1929 1937 1945 1953 1961 1969 1977	1794 1802 1810 1818 1826 1834 1842 1850 1858 1866 1874 1890 1898 1914 1922 1930 1938 1946 1954 1952 1957 1978	1795 1803 1801 1811 1819 1827 1835 1843 1851 1859 1867 1875 1883 1891 1915 1923 1931 1939 1947 1955 1963 1971 1979	1796 1804 1812 1820 1828 1836 1844 1852 1860 1868 1876 1884 1892 1900 1908 1916 1921 1940 1948 1956 1964 1952 1980	1797 1805 1813 1821 1823 1845 1853 1861 1865 1877 1855 1893 1901 1909 1917 1925 1933 1941 1949 1957 1965 1973 1981	1798 1806 1814 1822 1830 1838 1846 1854 1862 1878 1878 1878 1920 1918 1920 1958 1942 1950 1954 1950 1954 1966 1974 1982	1799 1807 1815 1823 1831 1839 1847 1855 1863 1871 1875 1903 1911 1919 1927 1943 1951 1955 1967 1975 1983	to to 3777 2047
3410 3420 3430 3430 3450 3460 3510 3520 3530 3540 3550 3540 3550 3540 3550 3540 3550 3540 3550 3610 3620 3630 3650 3650 3640 3650 3640 3650 3640 3650 3710	1792 1800 1808 1816 1824 1832 1840 1848 1856 1864 1872 1880 1888 1896 1888 1896 1888 1994 1912 1920 1928 1944 1955 1968 1976	1793 1801 1809 1817 1825 1833 1841 1849 1857 1865 1873 1881 1889 1905 1913 1921 1929 1937 1945 1953 1961 1977	1794 1802 1810 1818 1826 1834 1850 1834 1850 1838 1866 1914 1906 1914 1922 1930 1936 1946 1954 1966 1970	1795 1803 1811 1819 1827 1835 1843 1851 1859 1867 1875 1883 1891 1915 1923 1931 1947 1945 1963 1971 1979	1796 1804 1812 1820 1828 1836 1844 1852 1860 1868 1876 1884 1908 1908 1916 1924 1948 1948 1956 1964 1956 1988	1797 1805 1813 1821 1827 1845 1853 1861 1865 1893 1901 1909 1917 1925 1933 1941 1949 1957 1949 1957 1973	1798 1806 1814 1822 1830 1858 1854 1854 1854 1876 1878 1886 1894 1910 1918 1926 1934 1945 1958 1956 1974 1982	1799 1807 1815 1823 1831 1839 1847 1855 1863 1871 1879 1983 1991 1927 1935 1943 1951 1959 1963 1975 1983 1991	to to 3777 2047
3410 3420 3430 3450 3450 3450 3540 3540 3520 3520 3550 3550 3550 3550 3550 355	1792 1800 1808 1816 1824 1832 1840 1848 1856 1864 1872 1880 1888 1896 1904 1912 1920 1928 1944 1952 1968 1976 1984	1793 1801 1809 1817 1825 1833 1841 1849 1857 1865 1913 1873 1881 1873 1881 1905 1913 1921 1945 1953 1961 1969 1977 1985 1993 2001	1794 1802 1810 1818 1826 1834 1850 1858 1866 1874 1882 1890 1914 1906 1914 1922 1930 1938 1946 1954 1970 1978 1986 1994 2002 2010	1795 1803 1811 1819 1827 1843 1851 1855 1843 1851 1875 1883 1891 1975 1947 1955 1947 1979 1947 1979 1987 1995 2003	1796 1804 1812 1820 1828 1836 1844 1852 1860 1868 1876 1884 1892 1908 1916 1924 1938 1948 1956 1948 1972 1980 1988 1996 2004	1797 1805 1813 1821 1827 1845 1853 1861 1867 1885 1893 1909 1917 1925 1933 1941 1949 1957 1941 1949 1957 1941 1949 1957 1941 1949 1957 1973 1981	1798 1806 1814 1822 1830 1838 1846 1854 1862 1874 1878 1896 1991 1910 1918 1926 1934 1942 1950 1958 1956 1954 1954 1954 1958 1954 1954 1954 1954 1954 1954 1954 1954	1799 1807 1815 1823 1831 1839 1847 1855 1963 1871 1919 1927 1935 1943 1951 1975 1983 1991 1995 1995	to to 3777 2047
3410 3420 3430 3450 3450 3450 3540 3540 3520 3520 3550 3550 3550 3550 3550 355	1792 1800 1808 1816 1824 1832 1840 1848 1856 1864 1872 1880 1888 1896 1904 1912 1920 1928 1944 1952 1968 1976 1984	1793 1801 1809 1817 1825 1833 1841 1849 1857 1865 1913 1873 1881 1873 1881 1905 1913 1921 1945 1953 1961 1969 1977 1985 1993 2001	1794 1802 1810 1818 1826 1834 1850 1858 1866 1874 1882 1890 1914 1906 1914 1922 1930 1938 1946 1954 1970 1978 1986 1994 2002 2010	1795 1803 1811 1819 1827 1843 1851 1855 1843 1851 1875 1883 1891 1975 1947 1955 1947 1979 1947 1979 1987 1995 2003	1796 1804 1812 1820 1828 1836 1844 1852 1860 1868 1876 1884 1892 1908 1916 1924 1938 1948 1956 1948 1972 1980 1988 1996 2004	1797 1805 1813 1821 1827 1845 1853 1861 1867 1885 1893 1909 1917 1925 1933 1941 1949 1957 1941 1949 1957 1941 1949 1957 1941 1949 1957 1973 1981	1798 1806 1814 1822 1830 1838 1846 1854 1862 1874 1878 1896 1991 1910 1918 1926 1934 1942 1950 1958 1956 1954 1954 1954 1958 1954 1954 1954 1954 1954 1954 1954 1954	1799 1807 1815 1823 1831 1839 1847 1855 1963 1871 1919 1927 1935 1943 1951 1975 1983 1991 1995 1995	to to 3777 2047
3410 3420 3430 3450 3450 3450 3540 3540 3520 3520 3550 3550 3550 3550 3550 355	1792 1800 1808 1816 1824 1832 1840 1848 1856 1864 1872 1880 1888 1896 1904 1912 1920 1928 1944 1952 1968 1976 1984	1793 1801 1809 1817 1825 1833 1841 1849 1857 1865 1913 1873 1881 1873 1881 1905 1913 1921 1945 1953 1961 1969 1977 1985 1993 2001	1794 1802 1810 1818 1826 1834 1850 1858 1866 1874 1882 1890 1914 1906 1914 1922 1930 1938 1946 1954 1970 1978 1986 1994 2002 2010	1795 1803 1811 1819 1827 1843 1851 1855 1843 1851 1875 1883 1891 1975 1947 1955 1947 1979 1947 1979 1987 1995 2003	1796 1804 1812 1820 1828 1836 1844 1852 1860 1868 1876 1884 1892 1908 1916 1924 1938 1948 1956 1948 1957 1980 1988 1996 2004	1797 1805 1813 1821 1823 1821 1837 1845 1853 1861 1869 1917 1925 1933 1941 1949 1957 1945 1941 1949 1957 1941 1949 1957 1941 1949 1957 1973 1981	1798 1806 1814 1822 1830 1838 1846 1854 1862 1874 1878 1896 1991 1910 1918 1926 1934 1942 1950 1958 1956 1954 1954 1954 1958 1954 1954 1954 1954 1954 1954 1954 1954	1799 1807 1815 1823 1831 1839 1847 1855 1963 1871 1919 1927 1935 1943 1951 1975 1983 1991 1995 1995	to to 3777 2047
3410 3420 3430 3450 3450 3450 3540 3540 3540 3520 3520 3520 3550 3550 3550 3550 355	1792 1800 1808 1816 1824 1832 1848 1856 1864 1872 1880 1888 1896 1904 1912 1920 1928 1936 1944 1952 1960 1968 1976 1984 1996 2008 2008 2008	1793 1801 1809 1817 1825 1833 1841 1849 1857 1865 1873 1849 1897 1905 1913 1921 1929 1937 1945 1953 1961 1953 1961 1953 1961 1993 2001 2009 2017 2025	1794 1802 1810 1818 1826 1834 1842 1850 1858 1866 1874 1890 1898 1914 1922 1930 1938 1946 1954 1954 1954 1954 1958 1994 20010 2018 20218	1795 1803 1811 1819 1827 1835 1843 1851 1855 1883 1851 1875 1883 1937 1915 1939 1947 1955 1939 1947 1959 1947 1979 1987 1995	1796 1804 1812 1820 1828 1836 1848 1852 1860 1968 1876 1884 1892 1900 1908 1916 1924 1948 1956 1954 1956 1954 1950 1958 1996 2004 2012 2012 2020 2028	1797 1805 1813 1821 1829 1837 1845 1853 1861 1869 1877 1885 1931 1901 1909 1957 1949 1957 1949 1957 1949 1957 1949 1957 1949 1997 2005 2013 2021 2022	1798 1798 1806 1814 1822 1830 1838 1846 1854 1854 1962 1910 1978 1941 1942 1950 1958 1956 1974 1982 2006 1998 2006 1998 2001 2014 2022 2030	1799 1807 1815 1823 1831 1839 1847 1855 1903 1911 1959 1943 1951 1959 1967 1955 1983 1991 1999 2007 2015 2023 2031	to to 3777 2047

95118A, F4

NUMERICAL INFORMATION

OCTAL-DECIMAL INTEGER CONVERSION TABLE

0 1 2 3 4 5 6	7
4000 2048 4000 2048 2049 2050 2051 2052 2053 2054	
to to 4010 2056 2057 2058 2059 2060 2061 2062 4020 2064 2065 2066 2067 2068 2069 2070	
4777 2559 4020 2064 2065 2066 2067 2076 2070 (Octol) (Decimal) 4030 2072 2073 2074 2075 2076 2077 2078	
4040 2080 2081 2082 2083 2084 2085 2086	2087
4050 2088 2089 2090 2091 2092 2093 2094	2095
Octal Decimal 4060 2096 2097 2098 2099 2100 2101 2102	2103
10000 - 4096 4070 2104 2105 2106 2107 2108 2109 2110	2111
20000 - 8192 30000 - 12288 4100 2112 2113 2114 2115 2116 2117 2118	2119
40000 16384 4110 2120 2121 2122 2123 2124 2125 2126	2127
50000 - 20480 4120 2128 2129 2130 2131 2132 2133 2134	
60000 - 24576 4130 2136 2137 2138 2139 2140 2141 2142	2143
4140 2144 2145 2146 2147 2148 2149 2150 4150 2152 2153 2154 2155 2156 2157 2158	2151
4160 2160 2161 2162 2163 2164 2165 2166	2167
4170 2168 2169 2170 2171 2172 2173 2174	2175
4200 2176 2177 2178 2179 2180 2181 2182 4210 2184 2185 2186 2187 2188 2189 2190	
4220 2192 2193 2194 2195 2196 2197 2198	2199
4230 2200 2201 2202 2203 2204 2205 2206	2207
A240 2208 2209 2210 2211 2212 2213 2214	2215
4250 2216 2217 2218 2219 2220 2221 2222	2223
4260 2224 2225 2226 2227 2228 2229 2230	2231
4270 2232 2233 2234 2235 2236 2237 2238	2239
4300 2240 2241 2242 2243 2244 2245 2246	2247
4310 2248 2249 2250 2251 2252 2253 2254	
4320 2256 2257 2258 2259 2260 2261 2262	2263
4330 2264 2265 2266 2267 2268 2269 2270 4340 2272 2273 2274 2275 2276 2277 2278	2271
4340 2272 2273 2274 2275 2276 2277 2278 4350 2280 2281 2282 2283 2284 2285 2286	
4360 2288 2289 2290 2291 2292 2293 2294	2295
4370 2296 2297 2298 2299 2300 2301 2302	2303
0 1 2 3 4 5 6	7
5000 2550 2552 2552 2552 2554 2555 2554	
5000 2560 2560 2561 2562 2563 2564 2565 2566 5010 2568 2569 2570 2571 2572 2573 2574	2567 2575
5000 2560 5000 2560 2561 2562 2563 2564 2565 2566 to to 5010 2568 2569 2571 2572 2573 2574 5777 3071 5020 2576 2577 2578 2579 2581 2589	2567 2575 2583
5000 2560 5000 2560 2561 2562 2563 2564 2565 2566 to 5010 2568 2569 2570 2571 2572 2573 2574 5777 3071 5020 2576 2577 2578 2580 2581 2582 (Octab) (Decimal) 5030 2584 2585 2586 2587 2580 2581 2582	2567 2575 2583 2591
5000 2560 5000 2560 2560 2560 2561 2562 2563 2564 2565 2565 2566 2577 2577 2577 2571 2572 2573 2577 2570 2581 2582 2581 2582 2581 2582 2581 2582 2581 2582 2581 2582 2581 2582 2597 2598 2597 <th< td=""><td>2567 2575 2583 2591 2599</td></th<>	2567 2575 2583 2591 2599
5000 to 5777 3071 (Octal) 2560 to 3071 2560 5000 2560 2560 2561 2562 2563 2564 2565 2566 5000 10 5010 2568 2569 2570 2571 2572 2573 2574 5020 2576 2577 2578 2599 2590 2581 2582 5030 2584 2585 2586 2587 2588 2592 2593 2594 2595 2596 2597 2598 5040 2502 2603 2601 2602 2603 2604 2605 2606 5050 2608 2609 2610 2611 2613 2614	2567 2575 2583 2591 2599 2607 2615
5000 to 5777 2560 3071 (Octol) 5000 to 5010 2560 2560 2561 2562 2563 2564 2565 2566 5077 3071 3071 5020 2576 2577 2578 2579 2581 2582 2586 2587 2582 2584 2582 2592 2593 2591 2592 2592 2593 2594 2597 2596 2597 2598 5050 2594 2595 2596 2597 2598 2598 2597 2598 2596 2597 2598 2596 2597 2598 2505 2596 2597 2598 2504 2601 2602 2603 2605 2605 2605 2605 2605 2605 2605 2605 2605 2506 2597 2598 2504 2502 2601 2602 2605 2605 2605 2605 2605 2605 2605 2605 2605 2605 2605 2605 2605 2605 2505	2567 2575 2583 2591 2599 2607 2615
5000 to 5777 2560 5071 5000 5071 2560 5070 2560 5569 2561 5569 2562 2576 2571 2572 2573 2574 (Octol) (Decimol) 5030 2588 2580 2577 2578 2579 2580 2581 2582 5030 2592 2593 2594 2595 2596 2597 2598 2592 2593 2594 2595 2596 2597 2598 2597 2598 2597 2598 5050 2600 2601 2602 2603 2604 2605 2606 5060 2608 2609 2610 2611 2613 2614 5070 2616 2617 2618 2619 2620 2621 2622	2567 2575 2583 2591 2599 2607 2615 2623
5000 2560 5000 2560 2561 2562 2563 2564 2565 2565 5777 3071 5020 2562 2577 2578 2579 2580 2581 2582 (Octal) (Decimal) 5030 2592 2592 2586 2587 2582 2586 2587 2582 2586 2587 2582 2586 2587 2582 2592 2592 2592 2592 2592 2592 2592 2592 2592 2592 2593 2594 2555 2596 2507 2593 2594 2559 2596 2593 2594 2593 2594 2593 2594 2595 2596 2596 2598 2500 2610 2611 2612 <	2567 2575 2583 2591 2599 2607 2615 2623 2631 2631 2639
5000 to 5777 2560 3071 (Octol) 5000 to 5010 2560 2560 2561 2562 2563 2564 2565 2566 5000 2568 2569 2571 2571 2572 2573 2574 (Octol) (Decimol) 5030 2582 2586 2587 2588 2589 2592 2593 2564 2605 2606 5040 2592 2592 2593 2594 2595 2596 2597 2598 2592 2593 2594 2605 2606 2601 2602 2603 2604 2605 2606 2600 2601 2611 2612 2613 2614 5070 2616 2617 2618 2619 2620 2621 2622 5100 2622 2632 2632 2633 2634 2635 2636 2637 2638 5110 2632 2631 2641 2642 2643 2643 2643 2644 2645 <	2567 2575 2583 2591 2599 2607 2615 2623 2631 2631 2639 2647
5000 to 5777 (Octal) 2560 to 3071 (Decimal) 5000 5000 2560 2560 2561 2562 2562 2570 2574 2573 2574 2573 2575 2577 2573 2574 2572 2573 2573 2574 2572 2573 2573 2574 2580 2581 2580 2582 2580 2582 2580 2592 2580 2592 2580 2592 2580 2597 2588 2597 2598 2596 2596 2597 2598 2596 2597 2598 2596 2597 2598 2590 2500 2502 2503 2594 2595 2596 2597 2598 2501 2600 2600 2600 2600 2600 2600 2600 2600 2601 2602 2603 2604 2605 2602 2621 2622 5000 2616 2617 2618 2619 2620 2621 2622 2630 2614 2612 2632 2632 2632 2634 2635 2636 2637 2638 2632 2633 2634 2635 2636 2637 2638 2632 2633	2567 2575 2583 2591 2599 2607 2615 2623 2631 2639 2647 2655
5000 2560 5000 2560 2561 2562 2563 2564 2565 2565 5077 3071 5020 2576 2577 2578 2579 2581 2582 2592 2582 2589 2581 2582 2589 2581 2582 2589 2581 2582 2589 2581 2582 2591 2582 2592 2593 2594 2597 2598 5050 2606 2606 2606 2606 2606 2606 2606 2606 2606 2602 2612 2622 2610 2611 2612 2612 2622 5000 2616 2617 2618 2619 2630 2612 2622 2610 2611 2612 2612 2622 5100 2624 2625 2626 2627 2628 2629 2630 5110 2632 2633 2634 2645 2646 2614 2642 2643 2644	2567 2575 2583 2591 2599 2607 2615 2623 2631 2639 2647 2655 2663
5000 to 5777 2560 3071 5000 5010 2560 2561 2561 2562 2563 2564 2565 2565 2566 2577 2577 2577 2571 2572 2573 2572 2573 2572 2573 2581 2582 2581 2582 2581 2582 2591 2592 2593 2594 2592 2591 2591 2592 2591 2591 2592 2591 2591 2592 2591 2593 2594 2592 2591 2595 2596 2604 2604 2604 2604 2604 2604 2604 2604 2604 2604 2604 2604 2604 2604 2604 2604 2604 2604 2604 261	2567 2575 2583 2591 2599 2607 2615 2623 2631 2639 2647 2655 2663 2671
5000 to 5777 2560 5077 2560 5000 2560 2561 2562 2563 2564 2565 2566 5077 3071 (Octol) 5030 2582 2569 2577 2578 2579 2582 2582 2586 2587 2588 2582 2592 <t< td=""><td>2567 2575 2583 2591 2599 2607 2615 2623 2631 2639 2647 2655 2663 2671 2679</td></t<>	2567 2575 2583 2591 2599 2607 2615 2623 2631 2639 2647 2655 2663 2671 2679
5000 to 5777 (Octol) 2560 to 3071 (Octol) 5000 5000 2560 2561 2562 2563 2564 2565 2565 5777 (Octol) 3071 (Decimol) 5020 2576 2577 2578 2579 2580 2581 2585 5030 2584 2585 2586 2587 2588 2589 2591 2592 5040 2592 2593 2594 2595 2596 2597 2598 2590 2597 2598 2591 2598 2590 2591 2592 2593 2594 2595 2596 2597 2598 2590 2597 2598 2590 2597 2598 2590 2597 2598 2590 2597 2598 2590 2597 2598 2597 2598 2597 2598 2590 2597 2598 2597 2598 2597 2598 2597 2598 2597 2598 2597 2598 2597 2598 2597 2501 <	2567 2575 2583 2591 2599 2607 2615 2623 2631 2639 2647 2655 2663 2671 2679 2687
5000 2560 5000 2560 2561 2562 2563 2564 2565 2565 5777 3071 5020 2576 2577 2578 2579 2580 2581 2582 2589 2581 2582 2589 2581 2582 2589 2581 2582 2589 2581 2582 2581 2582 2581 2582 2581 2582 2581 2582 2581 2582 2581 2581 2582 2591 2598 5591 2598 5501 2502 2502 2502 2592 2591 2598 5501 2502 2502 2502 2502 2591 2598 5501 2502 2502 2502 2501 2502 2501 2502 2501 2502 2501 2502 2501 2502 2501 2501 2502 2501 2502 2501 2501 2501 2501 2501 2501 2501 2501 2501 25	2567 2575 2583 2591 2599 2607 2615 2623 2631 2639 2647 2655 2663 2671 2679 2687 2695
5000 to 5777 (Octol) 2560 to 3071 (Octol) 5500 to 3071 (Octol) 2560 to 5010 2560 2568 2561 2562 2563 2577 2573 2578 2572 2573 2573 2573 2573 2573 2573 2573 2573 2573 2573 2573 2573 2573 2573 2573 2573 2573 2573 2573 2580 2580 2580 2590 2580 2590 2580 2597 2598 2593 2594 2593 2594 2597 2598 2584 2589 2590 5040 2592 2593 2594 2595 2596 2597 2598 2591 2598 2500 2601 2602 2603 2604 2605 2609 2601 2612 2612 2612 2622 5100 2624 2625 2626 2627 2628 2629 2630 5110 2632 2631 2641 2642 2643 2641 2645 2646 2651 2652 2653 2654 2654 2654 2664 2665 2666 2667 2668 2669 2670 2678 2	25677 25755 2583 25919 26077 2615 2623 2631 2637 2647 2647 2647 2647 2647 2647 2647 264
5000 2560 5000 2560 2560 2561 2562 2563 2564 2565 2565 2565 2565 2565 2565 2571 2572 2573 2573 2574 2572 2573 2573 2574 2572 2573 2574 2572 2573 2574 2572 2573 2574 2572 2573 2574 2572 2573 2574 2573 2574 2572 2573 2574 2573 2574 2573 2574 2573 2574 2573 2574 2573 2574 2573 2574 2573 2574 2575 2596 2597 2598 5050 2600 2601 2602 2603 2604 2605 2606 2604 2604 2604 2604 2604 2614 2614 2614 2614 2614 2614 2614 2614 2614 2614 2614 2614 2614 2614 2614 2614 2614 <td< td=""><td>25677 25755 2583 25919 2697 2615 2623 2637 2645 2663 2665 2663 2679 2665 2703 2701 2703 2711 2719</td></td<>	25677 25755 2583 25919 2697 2615 2623 2637 2645 2663 2665 2663 2679 2665 2703 2701 2703 2711 2719
5000 2560 5000 2560 2560 2561 2562 2563 2564 2565 2565 5077 3071 5020 2576 2577 2578 2579 2582 2582 2582 2582 2582 2582 2582 2582 2582 2582 2582 2582 2582 2592 2593 2592 2592 2593 2594 2595 2596 2517 2512 2513 2514 2613 2614 2605 2606 2600 2601 2612 2612 2622 2613 2614 2613 2614 2613 2614 2614 2614 2612 2622 2620 2621 2622 2620 2621 2622 2630 2611 2612 2613 2614 2642 2643 2644 2645 2646 2610 2612 2622 2630 2631 2632 2631 2634 2642 2643 2644 2645 2646 26	25677 25755 25833 25991 26077 26155 26233 2631 2639 26477 26555 26637 26711 26795 26877 26955 27013 2711 2779
5000 2560 2560 2561 2561 2562 2563 2564 2565 2565 5777 3071 5020 2576 2577 2578 2579 2580 2581 2582 2582 2582 2583 2584 2585 2586 2587 2588 2589 2591 25	2567 2575 2583 2591 2599 2607 2615 2623 2631 2639 2647 2655 2663 2671 2679 2687 2695 2703 2711 2719 2727 2727 2727 2727
5000 to 5777 2560 3071 5600 5010 2560 2561 2562 2563 2564 2565 2566 5077 3071 5020 2576 2577 2578 2579 2580 2581 2582 2586 2587 2588 2589 2581 2582 2586 2587 2588 2589 2591 2592 2593 2594 2585 2596 2597 2598 5050 2600 2601 2602 2603 2604 2605 2606 2606 2600 2610 2611 2612 2612 2622 2620 2621 2622 2620 2621 2622 2630 2611 2612 2622 2630 2611 2612 2622 2630 2611 2612 2622 2630 2610 2611 2612 2622 2630 2631 2632 2631 2634 2643 2644 2645 2646 2630 2631 2632 2631 2632 2631 </td <td>2567 2575 2583 2591 2599 2607 2615 2623 2631 2639 2647 2655 2663 2663 2663 2667 2665 2703 2711 2719 2727 2775 27735</td>	2567 2575 2583 2591 2599 2607 2615 2623 2631 2639 2647 2655 2663 2663 2663 2667 2665 2703 2711 2719 2727 2775 27735
5000 to 5777 (Octel) 2560 to 3071 (Octel) 5000 to 5010 2560 2561 2562 2563 2564 2565 2565 5000 2560 2571 2572 2573 2572 2573 2572 2573 2572 2573 2574 2572 2573 2574 2582 2582 2589 2589 2589 2581 2582 2582 2582 2582 2581 2582 2582 2582 2582 2581 2585 2587 2588 2587 2588 2580 2581 2580 2581 2580 2581 2580 2581 2581 2581 2581 2580 2581 2580 2581 </td <td>2567 2575 2583 2591 2599 2607 2615 2623 2631 2637 2665 2663 2671 2679 2687 2695 2703 2719 2727 2719 2727 2735 2743 2751</td>	2567 2575 2583 2591 2599 2607 2615 2623 2631 2637 2665 2663 2671 2679 2687 2695 2703 2719 2727 2719 2727 2735 2743 2751
5000 to 5777 (Octol) 2560 to 3071 (Octol) 5000 to 5010 (Decimal) 2560 5600 5600 5600 5600 5600 5600 5600	2567 25755 2583 2599 2607 2623 2639 2647 2655 2663 2671 2679 2687 2679 2687 2703 2711 2719 2727 2727 27275 2743 2751
5000 to 5777 (Octol) 2560 to 3071 (Octol) 5000 5000 (Decimol) 2560 5000 5000 5000 5000 5026 2561 2562 2560 2570 2577 2578 2579 2580 2587 2588 2580 2587 2588 2580 2587 2588 2580 2587 2588 2580 2597 2598 2597 2598 2597 2598 2597 2598 2597 2598 2597 2598 2597 2598 2597 2598 2500 2600 2600 2600 2600 2600 2600 2600	2567 25755 2583 2591 2607 2607 2605 2663 2663 2663 2663 2663 2663 2703 2711 2727 2735 27751 2751 2751 2759 2765
5000 to 5777 (Octol) 2560 to 3071 (Octol) 2560 to 5010 (Decimol) 2560 to 5000 to 5000 to 5020 to 5030 to 5030 to 5030 to 5030 to 5030 to 5040 to 5030 to 5040 to 5040 to 5040 to 5040 to 505	2567 25755 2583 2591 2599 2667 2665 2663 2663 2671 2669 2667 2669 2703 2711 2719 2727 2735 27753 27751 27759 2763
5000 to 5777 (Octol) 2560 to 3071 (Octol) 5500 (Decimol) 2560 2568 2561 2562 2563 2564 2565 2565 5000 2568 2569 2571 2572 2573 2574 2572 2573 2574 2572 2573 2574 2589 2591 2631 <td< td=""><td>25677 25755 25755 2583 25911 2599 26675 26623 26637 26655 26637 2679 26857 2679 26877 2707 2711 2719 27273 2743 27751 27759 27753 27751</td></td<>	25677 25755 25755 2583 25911 2599 26675 26623 26637 26655 26637 2679 26857 2679 26877 2707 2711 2719 27273 2743 27751 27759 27753 27751
5000 to 5777 (Octol) 2560 to 3071 (Octol) 2560 5000 (Decimel) 2560 5000 5640 5662 5662 5670 5677 577 577 5030 2561 2562 2576 2577 2578 2579 2580 2587 2588 2580 2587 2598 2597 2598 2597 2598 2597 2598 2597 2598 2597 2598 2597 2598 2597 2598 2597 2598 2597 2598 2500 2600 2601 2602 2603 2604 2602 2603 2604 2610 2610 2610 2610 2610 2610 2610 2610	2567 2575 2575 2575 2593 2591 2607 2615 2623 2631 2633 2631 2635 2663 2671 2679 2679 2679 2773 2773 2775 2743 2751 2775 2775 2775 2775
5000 10 5777 (Octol) 2560 10 3071 (Octol) 2560 2568 2561 2562 2563 2564 2565 2565 5020 2576 2577 2578 2579 2580 2581 2582 5020 2576 2577 2578 2579 2580 2581 2582 5030 2584 2585 2586 2587 2588 2589 2591 2598 5040 2592 2593 2594 2595 2596 2597 2598 5050 2600 2601 2601 2602 2603 2604 2605 2602 2621 2622 5100 2624 2625 2627 2628 2637 2638 5100 2642 2643 2643 2643 2644 2645 2644 2645 2644 2645 2646 2657 2658 2659 2660 2661 2652 2653 2645 2646 2645 2646 2642 2645<	25677 25755 2583 25911 25999 2607 26155 2623 2639 2637 2655 2663 26639 26647 2679 2687 2703 2711 2719 2727 2733 2751 2743 27751 27753 27753 27753 27751 27753 27751 27753 27751 27753 27751 27753 27751 27753 27751 27753 27751 27753 27751 27753 27751 27753 27751 27753 27751 27753 27751 27753 27751 27753 27751 27753 2775757 27757575775775777777

	0	1	2	3	4	5	6	7
4400		2305	2306	2307	2308	2309	2310	2311
4410		2313	2314	2315	2316	2317	2318	2319
4420		2321 2329	2322 2330	2323 2331	2324 2332	2325 2333	2326 2334	2323
4430		2329	2338	2339	2332	2333	2342	234
4450		2345	2346	2347	2348	2349	2350	235
4460		2353 2361	2354 2362	2355 2363	2356 2364	2357 2365	2358 2366	235
4500		2369		2371	2372	2373	2374	237
4510	2376	2377	2378	2379	2380	2381	2382	238
4520		2385 2393		2387 2395	2388 2396	2389	2390 2398	239
4540		2401		2403	2404	2405	2406	240
4550	2408	2409		2411	2412	2413	2414	241
4560		2417 2425		2419 2427		2421 2429	2422 2430	242 243
4600		2433		2435	2436	2437	2438	243
4610		2433	2442	2443	2430 2444	2445	2446	244
4620		2449	2450	2451	2452	2453	2454	245
4630		2457		2459	2460	2461 2469	2462 2470	246 247
4640		2465 2473	2466 2474	2467 2475	2468 2476	2409	2478	247
4660		2481	2482	2483	2484	2485	2486	248
4670	2488	2489	2490	2491	2492	2493	2494	249
4700				2499	2500	2501	2502 2510	250
4710		2505 2513		2507 2515	2508 2516	2509	2510	251 251
4730				2523		2517 2525	2526	252
4740	2528	2529		2531	2532	2533		253
4750				2539 2547	2540	2541 2549	2542 2550	254 255
4770				2555	2548 2556	2557	2558	255
	0	1	2	3	4	5	6	7
5400	2816	2817		2819	2820	2821		2823
5410 5420	2824	2825 2833		2827 2835	2828 2836	2829 2837		2831 2839
5430	2840	2841		2843	2844	2845		2847
5440	2848	2849		2851	2852	2853	2854	2855
5450 5460	2856 2864	2857 2865		2859 2867	2860 2868	2861 2869	2862 2870	2863
	2872	2873		2875				2014
5470				2015	2876	2877	2878	2879
5470 5500	2880	2881	2882	2883		2877 2885	2886	2887
5500 5510	2888	2889	2890	2883 2891	2884 2892	2885 2893	2886 2894	2887 2895
5500 5510 5520	2888 2896	2889 2897	2890 2898	2883 2891 2899	2884 2892 2900	2885 2893 2901	2886 2894 2902	2887 2895 2903
5500 5510 5520 5530	2888 2896 2904	2889 2897 2905	2890 2898 2906	2883 2891	2884 2892 2900	2885 2893	2886 2894 2902 2910	2887 2895 2903 2911
5500 5510 5520	2888 2896 2904 2912 2920	2889 2897 2905 2913 2921	2890 2898 2906 2914 2922	2883 2891 2899 2907 2915 2923	2884 2892 2900 2908 2916 2924	2885 2893 2901 2909 2917 2925	2886 2894 2902 2910 2918 2926	2887 2895 2903 2911 2919 2927
5500 5510 5520 5530 5540	2888 2896 2904 2912	2889 2897 2905 291 3	2890 2898 2906 2914 2922 2930	2883 2891 2899 2907 2915 2923 2931	2884 2892 2900 2908 2916 2924 2932	2885 2893 2901 2909 2917	2886 2894 2902 2910 2918 2926 2934	2887 2895 2903 2911 2919 2927 2935
5500 5510 5520 5530 5540 5550 5560 5560 5570	2888 2896 2904 2912 2920 2928 2936	2889 2897 2905 2913 2921 2929 2937	2890 2898 2906 2914 2922 2930 2938	2883 2891 2899 2907 2915 2923 2931 2939	2884 2892 2900 2908 2916 2924 2932 2932 2940	2885 2893 2901 2909 2917 2925 2933 2941	2886 2894 2902 2910 2918 2926 2934 2942	2887 2895 2903 2911 2919 2927 2935 2943
5500 5510 5520 5530 5540 5550 5560 5570	2888 2896 2904 2912 2920 2928 2936 2944 2952	2889 2897 2905 2913 2921 2929	2890 2898 2906 2914 2922 2930 2938 2945	2883 2891 2899 2907 2915 2923 2931 2939 2947	2884 2892 2900 2908 2916 2924 2932 2940 2948	2885 2893 2901 2909 2917 2925 2933	2886 2894 2902 2910 2918 2926 2934 2942 2950	2879 2887 2895 2903 2911 2919 2927 2935 2943 2951 2959
5500 5510 5520 5530 5540 5550 5560 5570 5600 5610 5620	2888 2896 2904 2912 2920 2928 2936 2944 2952 2960	2889 2897 2905 2913 2921 2929 2937 2945 2953 2961	2890 2898 2906 2914 2922 2930 2938 2946 2954 2954	2883 2891 2899 2907 2915 2923 2931 2939 2947 2955 2963	2884 2892 2900 2908 2916 2924 2932 2940 2948 2956	2885 2893 2901 2909 2917 2925 2933 2941 2949 2957 2965	2886 2894 2902 2910 2918 2926 2934 2942 2950 2958 2966	2887 2895 2903 2911 2919 2927 2935 2943 2951 2951 2959 2967
5500 5510 5520 5530 5540 5550 5560 5570 5600 5610 5620 5620 5630	2888 2896 2904 2912 2920 2928 2936 2944 2952 2960 2968	2889 2897 2905 2913 2921 2929 2937 2945 2953 2961 2969	2890 2898 2906 2914 2922 2930 2938 2945 2954 2954 2962 2970	2883 2891 2899 2907 2915 2923 2931 2939 2947 2955 2963 2971	2884 2892 2900 2908 2916 2924 2932 2940 2948 2956 2964 2972	2885 2893 2901 2909 2917 2925 2933 2941 2949 2957 2965 2973	2886 2894 2902 2910 2918 2926 2934 2942 2942 2950 2958 2966 2974	2887 2895 2903 2911 2919 2927 2935 2943 2951 2959 2957 2957
5500 5510 5520 5530 5540 5550 5560 5570 5600 5610 5620 5620 5620 5630 5640	2888 2896 2904 2912 2920 2928 2936 2944 2952 2960 2968 2976	2889 2897 2905 2913 2921 2929 2937 2945 2953 2961 2969 2977	2890 2898 2906 2914 2922 2930 2938 2945 2954 2954 2962 2970 2978	2883 2891 2899 2907 2915 2923 2931 2939 2947 2955 2963 2971 2979	2884 2892 2900 2908 2916 2924 2932 2940 2948 2956 2964 2956 2964 2972 2980	2885 2893 2901 2909 2917 2925 2933 2941 2949 2957 2965 2973 2981	2886 2894 2902 2910 2918 2926 2934 2942 2950 2958 2966 2974 2982	2887 2895 2903 2911 2919 2927 2935 2943 2951 2959 2967 2975 2983
5500 5510 5520 5530 5540 5550 5560 5570 5600 5610 5620 5620 5630	2888 2896 2904 2912 2920 2928 2936 2944 2952 2960 2968 2976 2984	2889 2897 2905 2913 2921 2929 2937 2945 2953 2961 2969	2890 2898 2906 2914 2922 2930 2938 2945 2954 2954 2962 2970 2978 2986	2883 2891 2899 2907 2915 2923 2931 2939 2947 2955 2963 2971 2979 2987	2884 2892 2900 2908 2916 2924 2932 2940 2948 2956 2964 2972 2980 2988	2885 2893 2901 2909 2917 2925 2933 2941 2949 2957 2965 2973	2886 2894 2902 2910 2928 2934 2942 2950 2958 2956 2974 2982 2990 2998	2887 2895 2903 2911 2919 2927 2935 2943 2951 2959 2957 2957
5500 5510 5520 5530 5540 5550 5570 5600 5610 5620 5620 5620 5630 5640 5650	2888 2896 2904 2912 2920 2928 2936 2944 2952 2960 2968 2960 2968 2976 2984 2992	2889 2897 2905 2913 2921 2929 2937 2945 2953 2961 2969 2977 2985	2890 2898 2906 2914 2922 2930 2938 2945 2954 2962 2970 2978 2986 2994	2883 2891 2899 2907 2915 2923 2931 2939 2947 2955 2963 2971 2979 2987 2995	2884 2892 2900 2908 2916 2924 2932 2940 2948 2956 296 2980 2980 2988 2996	2885 2893 2901 2909 2917 2925 2933 2941 2957 2957 2957 2957 2965 2973 2981 2989	2886 2894 2902 2910 2926 2934 2942 2950 2958 2956 2974 2982 2982 2980 2998	2887 2895 2903 2911 2919 2927 2935 2943 2951 2959 2967 2975 2983 2991
5500 5510 5520 5530 5550 5550 5550 5660 5660 5660 566	2888 2896 2904 2912 2920 2928 2936 2944 2952 2960 2968 2976 2948 2976 2949 2958 2976 2949 2930 3000	2889 2897 2905 2913 2921 2929 2937 2945 2953 2969 2977 2985 2993 3001 3009	2890 2898 2906 2914 2922 2930 2938 2945 2954 2954 2954 2954 2954 2954 2954	2883 2891 2899 2907 2915 2923 2931 2939 2947 2955 2963 2971 2979 2987 2995 3003 3011	2884 2892 2900 2908 2916 2924 2932 2940 2948 2956 2956 2956 2956 2958 2996 3004 3012	2885 2893 2901 2909 2917 2925 2931 2941 2949 2957 2965 2973 2981 2981 2981 2997 3005 3013	2886 2894 2902 2910 2926 2934 2942 2950 2958 2958 2958 2958 2996 2974 2982 2990 2998 3006 3014	2887 2895 2903 2911 2919 2927 2935 2943 2951 2959 2967 2975 2983 2991 2999 3007 3015
5500 5510 5520 5530 5550 5550 5550 55570 55570 5660 5660 5	2888 2896 2904 2912 2928 2936 2944 2950 2968 2976 2968 2976 2984 2992 3000 3008	2889 2897 2905 2913 2921 2929 2937 2945 2953 2961 2969 2977 2985 2993 3001 3009 3017	2890 2898 2906 2914 2922 2930 2938 2954 2954 2962 2970 2978 2970 2978 2986 2994 3002 3010 3018	2883 2891 2899 2907 2915 2923 2931 2939 2947 2955 2963 2971 2995 3003 3011 3019	2884 2892 2908 2916 2924 2932 2940 2948 2956 2956 2956 2956 2972 2980 2980 2980 2998 2996 3004 3012	2885 2893 2901 2907 2925 2933 2941 2949 2957 2957 2957 2957 2965 2973 2989 2989 2989 2997 3005 3013 3021	2886 2894 2902 2910 2918 2926 2934 2942 2950 2958 2956 2974 2982 2982 2982 2990 2998 3006 3014 3022	2887 2895 2903 2911 2919 2927 2935 2943 2951 29551 2957 2975 2997 3007 3015 3023
5500 5510 5530 5550 5550 5550 5550 5600 5620 5620 562	2888 2896 2904 2912 2920 2928 2936 2952 2960 2958 2956 2968 2976 2968 2976 2988 2976 2988 2976 3000	2889 2897 2905 2913 2929 2937 2945 2953 2969 2977 2985 2993 3001 3009 3017 3025	2890 2898 2906 2914 2922 2930 2938 2945 2954 2954 2954 2956 2970 2976 2986 2996 29986 3002 3010 3018 3026	2883 2891 28997 2915 2923 2931 2939 2947 2955 2963 2979 2995 3003 3011 3019 3027	2884 2892 2908 2916 2924 2932 2940 2948 2956 2964 2972 2988 2996 3004 3012 3020 3028	2885 2893 2901 2909 2917 2925 2933 2941 2949 2957 2965 2973 2989 2997 3005 3005	2886 2894 2902 2910 2918 2926 2934 2942 2950 2958 2956 2978 2996 2998 3006 3014 3022 3030	2887 2895 2903 2911 2919 2927 2935 2943 2951 29551 2957 2975 2997 3007 3015 3023 3031
5500 5510 5520 5530 5550 5550 5550 5550 5550 5600 5610 5620 5630 5660 5660 5660 5660 5660 5660 5570 5770 57	2888 2896 2904 2912 2920 2928 2936 2952 2960 2958 2956 2968 2976 2968 2976 2988 2976 2988 2976 3000	2889 2897 2905 2913 2921 2929 2937 2945 2953 2961 2969 2977 2985 2993 3001 3009 3017	2890 2898 2906 2914 2922 2930 2938 2945 2954 2954 2954 2956 2970 2976 2976 2976 2986 3002 3010 3018 3026 3034	2883 2891 2907 2915 2923 2931 2939 2947 2955 2971 2955 2971 2995 2997 2995 3003 3011 3012 3027 3035	2884 2892 2900 2908 2916 2924 2940 2940 2956 2956 2956 2956 2958 2996 2958 2996 3004 3012 3028 3028 3036	2885 2893 2901 2907 2925 2933 2941 2949 2957 2957 2957 2957 2965 2973 2989 2989 2989 2997 3005 3013 3021	2886 2894 2902 2910 2918 2926 2934 2942 2950 2958 2956 2974 2980 2998 3006 3014 3022 3030 3038	2887 2895 2903 2911 2919 2927 2935 2943 2951 2959 2967 2975 2983 2991 2999 3007 3015 3023 3031 3039 3047
5500 5510 5520 5530 5550 5550 5550 55570 55570 5660 5660 5	2888 2896 2904 2912 2920 2928 2936 2952 2960 2968 2976 2988 2976 2992 3000 3008 3016 3024 3032	2889 2897 2905 2913 2921 2929 2937 2945 2953 2961 2965 2993 3001 3009 3017 3025 3033	2890 2898 2906 2914 2922 2930 2938 2945 2954 2954 2954 2954 2978 2978 2978 2978 2986 2994 3002 3010 3018 3026 3034 3050	2883 2891 2907 2915 2923 2931 2939 2947 2955 2963 2971 2963 2995 3003 3001 1 3019 3027 3035	2884 2892 2900 2908 2916 2924 2932 2940 2948 2956 2964 2956 2964 2972 2988 2996 3004 3012 3020 3028 3036	2885 2893 2901 2909 2917 2925 2933 2941 2957 2965 2973 2989 2997 3005 3013 3021 3029 3037	2886 2894 2902 2910 2918 2926 2934 2942 2950 2954 2950 2958 2956 2974 2982 2990 3006 3014 3022 3030 3038 3046 3054	2887 2895 2903 2911 2912 2927 2935 2943 2951 2959 2967 2975 2983 2991 3007 3015 3023 3031 3039

95118A.F5

NUMERICAL INFORMATION

OCTAL-DECIMAL INTEGER CONVERSION TABLE

	0	1	2	3	4	5	6	7
6000	3072	3073	3074	3075	3076	3077	3078	3079
6010	3080	3081	3082	3083	3084	3085	3086	3087
6020	3088	3089	3090	3091	3092	3093	3094	3095
6030	3096	3097	3098	3099	3100	3101	3102	3103
6040	3104	3105	3106	3107	3108 3116	3109	3110 3118	3111 3119
6050	3112	3113 3121	3114 3122	3115 3123	3124	3117 3125	3126	3127
6060 6070	3120 3128	3129	3130	3131	3132	3133	3134	3135
6100	3136	3137	3138	3139	3140	3141	3142	3143
6110	3144	3145	3146	3147	3148	3149	3150	3151
6120	3152	3153	3154	3155	3156	3157	3158	3159
6130	3160	3161	3162	3163	3164	3165	3166	3167
6140	3168	3169	3170	3171	3172	3173	3174	3175
6150	3176	3177	3178	3179	3180	3181	3182	3183
6160 6170	3184 3192	3185 3193	3186 3194	3187 3195	3188 3196	3189 3197	3190 3198	3191 3199
6200	3200	3201	3202	3203	3204	32 05	3206	3207
6210	3200	3209	3210	3211	3212	3213	3214	3215
6220	3216	3217	3218	3219	3220	3221	3222	3223
6230	3224	3225	3226	3227	3228	3229	3230	3231
6240	3232	3233	3234	3235	3236	3237	3238	3239
6250	3240	3241	3242	3243	3244	3245	3246	3247
6260	3248	3249	3250	3251	3252	3253	3254	3255
6270	3256	3257	3258	3259	3260	3261	3262	3263
6300	3264	3265	3266	3267	3268	3269	3270	3271
6310	3272	3273	3274	3275	3276	3277 3285	3278 3286	3279 3287
6320	3280	3281 3289	3282 3290	3283 3291	3284 3292	3293	3294	3295
6330 6340	3288 3296	3297	3290	3299	3300	3301	3302	3303
6350					3308	3309	3310	3311
		3305	3306	3307				
	3304	3305 3313	3306 3314	3307 3315	3316	3317	3318	3319
6360 6370	3304	3305 3313 3321						
6360	3304 3312	3313	3314	3315	3316	3317	3318	3319
6360 6370	3304 3312 3320 0	3313 3321 1	3314 3322 2	3315 3323 3	3316 3324	3317 3325	3318 3326	3319 3327
6360	3304 3312 3320	3313 3321	3314 3322	3315 3323	3316 3324 4	3317 3325 5	3318 3326 6 3590 3598	3319 3327 7 3591 3599
6360 6370 7000	3304 3312 3320 0 3584	3313 3321 1 3585	3314 3322 2 3586 3594 3602	3315 3323 3 3 3 3 3587 3595 3603	3316 3324 4 3588 3596 3604	3317 3325 5 3589 3597 3605	3318 3326 6 3590 3598 3606	3319 3327 7 3591 3599 3607
6360 6370 7000 7010	3304 3312 3320 0 3584 3592 3600 3608	3313 3321 1 3585 3593 3601 3609	3314 3322 2 3586 3594 3602 3610	3315 3323 3 3 3587 3595 3603 3611	3316 3324 4 3588 3596 3604 3612	3317 3325 5 3589 3597 3605 3613	3318 3326 6 3590 3598 3606 3614	3319 3327 7 3591 3599 3607 3615
6360 6370 7000 7010 7020 7030 7040	3304 3312 3320 0 3584 3592 3600 3608 3616	3313 3321 1 3585 3593 3601 3609 3617	3314 3322 2 3586 3594 3602 3610 3618	3315 3323 3 3 3587 3595 3603 3611 3619	3316 3324 4 3588 3596 3604 3612 3620	3317 3325 5 3589 3597 3605 3613 3621	3318 3326 6 3590 3598 3606 3614 3622	3319 3327 7 3591 3599 3607 3615 3623
6360 6370 7000 7010 7020 7030 7040 7050	3304 3312 3320 0 3584 3592 3600 3608 3616 3624	3313 3321 1 3585 3593 3601 3609 3617 3625	3314 3322 2 3586 3594 3602 3610 3618 3626	3315 3323 3 3 3587 3595 3603 3611 3619 3627	3316 3324 4 3588 3596 3604 3612 3620 3628	3317 3325 5 3589 3597 3605 3613 3621 3629	3318 3326 6 3590 3598 3606 3614 3622 3630	3319 3327 7 3591 3599 3607 3615 3623 3631
6360 6370 7000 7010 7020 7030 7040	3304 3312 3320 0 3584 3592 3600 3608 3616	3313 3321 1 3585 3593 3601 3609 3617	3314 3322 2 3586 3594 3602 3610 3618	3315 3323 3 3 3587 3595 3603 3611 3619	3316 3324 4 3588 3596 3604 3612 3620	3317 3325 5 3589 3597 3605 3613 3621	3318 3326 6 3590 3598 3606 3614 3622	3319 3327 7 3591 3599 3607 3615 3623
6360 6370 7000 7010 7020 7030 7040 7050 7060 7070	3304 3312 3320 0 3584 3592 3600 3608 3616 3624 3632 3640	3313 3321 1 3585 3593 3609 3617 3625 3633 3641	3314 3322 2 3586 3594 3602 3610 3618 3626 3634 3642	3315 3323 3 3 3587 3595 3603 3611 3619 3627 3635 3643	3316 3324 4 3588 3596 3604 3612 3620 3628 3636 3644	3317 3325 5 3589 3597 3605 3613 3621 3629 3637 3645	3318 3326 6 3590 3598 3606 3614 3622 3630 3638 3646	3319 3327 7 3591 3599 3607 3615 3623 3631 3639 3647
6360 6370 7000 7010 7020 7030 7040 7050 7060	3304 3312 3320 0 3584 3592 3600 3608 3616 3624 3632	3313 3321 1 3585 3593 3601 3609 3617 3625 3633	3314 3322 2 3586 3594 3602 3610 3618 3626 3634	3315 3323 3 3 3587 3595 3603 3611 3619 3627 3635	3316 3324 4 3588 3596 3604 3612 3620 3628 3636	3317 3325 5 3589 3597 3605 3613 3621 3629 3637 3645 3653 3661	3318 3326 6 3590 3598 3606 3614 3622 3630 3638 3646 3654 3654 3662	3319 3327 7 3591 3599 3607 3615 3623 3631 3639 3647 3655 3663
6360 6370 7000 7010 7020 7030 7040 7050 7060 7070 7100 7110 7110	3304 3312 3320 0 3584 3592 3600 3608 3616 3624 3632 3640 3648 3656 3664	3313 3321 1 3585 3593 3609 3617 3625 3633 3641 3649 3657 3665	3314 3322 2 3586 3594 3602 3610 3618 3626 3634 3642 3650 3658 3656	3315 3323 3 3587 3595 3603 3611 3619 3627 3635 3643 3651 3659 3651 3659 3667	3316 3324 4 3588 3596 3604 3612 3620 3628 3636 3644 3652 3660 3668	3317 3325 5 3589 3597 3605 3613 3621 3629 3637 3645 3653 3661 3669	3318 3326 6 3590 3598 3606 3614 3622 3630 3638 3646 3654 3654 3662 3670	3319 3327 7 3591 3599 3607 3615 3623 3631 3639 3647 3655 3663 3671
6360 6370 7000 7010 7020 7030 7040 7050 7060 7070 7100 7110 7110 7120 7130	3304 3312 3320 0 3584 3592 3600 3608 3616 3624 3632 3640 3648 3656 3664 3672	3313 3321 1 3585 3593 3601 3609 3617 3625 3633 3641 3649 3657 3665 3673	3314 3322 2 3586 3594 3600 3618 3626 3634 3642 3650 3658 3666 3674	3315 3323 3 3587 3595 3601 3619 3627 3635 3643 3651 3659 3667 3675	3316 3324 4 3588 3596 3696 3696 3628 3636 3644 3652 3660 3668 3676	3317 3325 5 3589 3597 3605 3613 3621 3629 3637 3645 3653 3661 3669 3677	3318 3326 3590 3598 3608 3614 3622 3630 3638 3646 3654 3654 3654 3670 3678	3319 3327 7 3591 3599 3607 3615 3623 3631 3639 3647 3655 3663 36671 3679
6360 6370 7000 7010 7030 7040 7050 7040 7050 7040 7050 7040 7110 7110 7110 71120 71130 71140	3304 3312 3320 3584 3592 3608 3616 3624 3632 3640 3648 3656 3648 3656 3664 3658	3313 3321 1 3585 3593 3601 3609 3617 3625 3631 3649 3649 3657 3665 36673 3681	3314 3322 2 3586 3594 3602 3610 3618 3626 3634 3642 3650 3658 3658 3658 36674 3682	3315 3323 3 3587 3595 3603 3611 3619 3625 3643 3651 3653 3643 3651 3657 3667 3675 3683	3316 3324 4 3588 3596 3604 3612 3620 3624 3636 3644 3652 3664 3652 3666 3668 3676 3684	3317 3325 5 3589 3597 3605 3613 3621 3627 3645 3653 3661 3661 3667 3665 3657 3685	3318 3326 3590 3598 3606 3614 3622 3630 3638 3646 3654 3654 3654 3654 3678 3678	3319 3327 7 3591 3599 3607 3615 3623 3631 3639 3647 3655 3663 3663 3663 36637 36636 36637 36637 36637 36637
6360 6370 7000 7010 7020 7030 7040 7050 7040 7050 7040 7050 7040 7100 7110 7110 7110 7110 7110 711	3304 3312 3320 0 3584 3592 3600 3608 3616 3624 3648 3656 3664 3656 3664 3656 3664 3680 3680	3313 3321 1 3585 3593 3601 3609 3617 3625 3633 3641 3649 3657 3665 3673 3681 3681 3681	3314 3322 2 3586 3594 3602 3610 3618 3626 3634 3642 3650 3658 3666 3658 3666 3674 3658 3666	3315 3323 3 3587 3595 3603 3611 3619 3627 3643 3651 3659 3667 3675 3663 3663 3659	3316 3324 4 3588 3596 3604 3612 3620 3628 3636 3644 3652 3660 3668 3668 3676 3676 3692	3317 3325 5 3589 3597 3605 3613 3629 3637 3645 3663 3661 3669 3677 3685 3693	3318 3326 3590 3598 3606 3614 3622 3630 3638 3646 3654 3654 3654 3654 3670 3678 3678 3694	3319 3327 7 3591 3599 3607 3615 3623 3631 3639 3647 3655 3663 3647 3657 3667 3657 3667
6360 6370 7000 7010 7030 7040 7050 7040 7050 7040 7050 7040 7110 7110 7110 71120 71130 71140	3304 3312 3320 3584 3592 3608 3616 3624 3632 3640 3648 3656 3648 3656 3664 3658	3313 3321 1 3585 3593 3601 3609 3617 3625 3631 3649 3649 3657 3665 36673 3681	3314 3322 2 3586 3594 3602 3610 3618 3626 3634 3642 3650 3658 3658 3658 36674 3682	3315 3323 3 3587 3595 3603 3611 3619 3625 3643 3651 3653 3643 3651 3657 3667 3675 3683	3316 3324 4 3588 3596 3604 3612 3620 3624 3636 3644 3652 3664 3652 3666 3668 3676 3684	3317 3325 5 3589 3597 3605 3613 3621 3627 3645 3653 3661 3661 3667 3665 3657 3685	3318 3326 3590 3598 3606 3614 3622 3630 3638 3646 3654 3654 3654 3654 3678 3678	3319 3327 7 3591 3599 3607 3615 3623 3631 3639 3647 3655 3663 3671 3679 3687 3687 3695 3703
6360 6370 7000 7010 7020 7030 7040 7050 7060 7070 7100 7110 7120 7110 7120 7110 7120 712	3304 3312 3320 0 3584 3592 3600 3608 3616 3624 3656 3664 3656 3664 3656 3664 3656 3664 3656 3656	1 3313 3321 1 3585 3593 3601 3609 3617 3625 3633 3641 3649 3653 3663 3665 3665 3663 3663 3673 3689 3705	2 3314 3322 2 3586 3594 3610 3618 3626 3634 3642 3650 3658 3664 3664 3664 3664 3664 3664 3664 366	3315 3323 3 3 3587 3595 3603 3611 3659 3643 3651 3659 3667 3675 3683 3691 3699	3316 3324 4 3588 3596 3604 3612 3620 3628 3636 3644 3652 3668 3668 3668 3668 3676 3668 3676 3668 3700 3708	3317 3325 5 3589 3597 3605 3613 3621 3629 3637 3645 3661 3661 3669 3677 3685 3693 3701	3318 3326 6 3590 3598 3606 3614 3622 3630 3638 3646 3654 3654 3654 3658 3654 3658 3654 3658 3694 3702	3319 3327 7 3591 3599 3607 3615 3623 3631 3639 3647 3655 3663 3671 3679 3687 3687 3695 3703
6360 6370 7000 7010 7020 7030 7040 7050 7050 7070 7100 7110 7120 7130 7140 7140 7140	3304 3312 3320 0 3584 3592 3600 3608 3614 3622 3640 3648 3656 3664 3652 3688 3688 3688	3313 3321 1 3585 3593 3601 3609 3617 3665 3633 3641 3649 3657 3665 3673 3689 3689	3314 3322 2 3586 3594 3602 3618 3662 3634 3663 3663 3663 3658 3666 3674 3682 3650 3658 3666 3674 3682 3690 3698	3315 3323 3587 3595 3603 3611 3619 3627 3635 3643 3651 3653 3667 3665 3667 3675 3683 3699 3707	3316 3324 4 3588 3596 3694 3612 3620 3620 3628 3636 3644 3652 3660 3668 3676 3688 3676 3682 3700	3317 3325 5 3589 3597 3605 3613 3613 3613 3613 3613 3613 3663 366	3318 3326 6 3590 3598 3606 3614 3622 3630 3634 3646 3654 3654 3670 3678 3678 3678 3678 3678 3678 3678 3678	3319 3327 7 3591 3599 3607 3615 3623 3631 3639 3647 3655 3663 3663 3663 3663 3663 3663 366
6360 6370 7000 7020 7030 7040 7050 7050 7050 7070 7100 7110 7120 7110 7120 7150 7150 7150 7150 7150 7150 7210 7220	3304 3312 3320 0 3584 3592 3600 3614 3632 3648 3648 3648 3656 3664 3656 3664 3656 3664 3656 3660 3688 3690 3688 3690 3688 3704	1 3585 3593 3601 3609 3617 3625 3633 3641 3649 3657 3665 3663 3661 3669 3673 3673 3673 3673 3705 3713 3721	3314 3322 2 3586 3594 3602 3610 3618 3626 3634 3642 3650 3658 3666 3658 3664 3658 3664 3658 3664 3658 3664 3658 3664 3674 3674 3679 36714	3315 3323 3587 3595 3603 3611 3619 3627 3635 3643 3651 3659 3667 3663 3675 3683 3675 3683 3691 3707 3715	3316 3324 4 3588 3596 3694 3612 3620 3628 3664 3664 3664 3664 3666 3668 3676 3668 3676 3676 3700 3708	3317 3325 5 3589 3597 3605 3613 3621 3629 3637 3645 3653 3661 3669 3667 3667 3667 3667 3709 3717 3775 3733	3318 3326 6 3590 3598 3606 3614 3622 3630 3638 3646 3654 3662 3670 3678 3678 3678 3678 3678 3678 3710 3718	3319 3327 7 3591 3599 3607 3615 3623 3631 3637 3647 3647 3647 3647 3647 3703 3711 3719 3727 3735
6360 6370 7000 7010 7010 7020 7030 7040 7040 7040 7140 7140 7140 7140 714	3304 3312 3320 0 3584 3592 3684 3616 3614 3656 3664 3664 3664 3664 3664 3664 366	3313 3321 1 3585 3593 3601 3607 3617 3625 3633 3641 3649 3657 3657 3657 3665 3673 3689 3689 3689 3689 3737	3314 3322 2 3586 3594 3602 3610 3638 3618 3626 3634 3642 3658 3658 3658 3658 3706 3714 3722 3730 3738	3315 3323 3 3587 3595 3603 3611 3619 3627 3645 3643 3659 36647 3659 3669 3707 3715 3723 3731	3316 3324 4 3588 3596 3604 3612 3620 3628 3634 3644 3652 3660 3668 3664 3668 3676 3669 3700 3708 3716 3724	3317 3325 5 3589 3597 3605 3613 3621 3629 3637 3645 3661 3669 3677 3685 3693 3701 3709 3717 3725 3733 3741	3318 3326 6 3590 3598 3606 3614 3622 3630 3638 3644 3662 3662 3670 3678 3662 3662 3670 3710 3718 3726 3734 3742	3319 3327 7 3591 3607 3615 3623 3631 3639 3647 3655 3663 3671 3687 3687 3687 3703 3711 3719 3727 3735
6360 6370 7000 7010 7020 7030 7050 7050 7050 7050 7050 7110 7110 711	3304 3312 3320 0 0 3584 3592 3600 3608 3616 3616 3648 3656 3664 3656 3664 3656 3668 3656 3668 3656 3688 3704	1 3321 3321 3321 3321 3321 3585 3593 3609 3617 3625 3633 3661 3663 3663 3663 3663 3667 36657 36657 36657 3667 3705 3713 3721 3729 3737 3745	2 33314 3322 2 3586 3594 3602 3610 3618 3626 3634 3662 3650 3658 36642 3650 3658 36642 3650 3658 3664 3674 3682 3690 3678 3798 3714 3712 3730 3738 3746	3315 3323 3587 3595 3603 3611 3619 3635 3643 3651 3655 3663 3691 3659 3663 3691 3699 3697 3707 3715 3723 3731	3316 3324 3324 3324 3324 3324 3588 3596 36012 3612 3620 3644 3652 3664 3652 3664 3656 3676 3670 3708 3716 3724 3743 3748	3317 3325 5 3589 3597 3605 3613 3621 3623 3663 3663 3663 3663 3663	3318 3326 3326 3326 3590 3598 3606 3614 3623 3614 3623 3638 3646 3654 3678 3686 3678 3678 3710 3718 3726 3734 3743 3750	3319 3327 7 3591 3599 3607 3615 3623 3623 3623 3667 3667 3679 3687 3679 3687 3679 3677 3679 3673 3771 3707 3719
6360 6370 7000 7000 7010 7020 7030 7050 7050 7050 7050 7050 7050 705	3304 3312 3320 0 0 3584 3592 3600 3608 3616 3624 3632 3640 3656 3664 3656 3664 3656 3664 3674 3712 3720 3728 3736 3732	1 3313 3321 1 3585 3593 3601 3609 3617 3625 3633 3631 3641 3649 3657 3665 3673 3765 3713 3721 3729 3737 3745 3753	2 2 3586 3594 3602 3610 3618 3626 3634 3642 3650 3658 3662 3674 3652 3658 3662 3678 3768 3714 3722 3730 3738	3315 3323 3587 3595 3603 3611 3619 3627 3635 3643 3651 3643 3663 3663 3663 3663 3663 3683 3693 3705 3715	3316 3324 4 3588 3596 3604 3612 3620 3628 3636 3644 3652 3664 3652 3664 3668 3676 3668 3676 3708 3708 3700 3716 3724 3720 3748 3756	3317 3325 5 3589 3597 3605 3613 3621 3623 3645 3653 3645 3653 3645 3665 3693 3701 3709 3717 3725 3733 3741 3757	3318 3326 6 3590 3598 3606 3614 3654 3654 3654 3654 3658 3654 3658 3654 3670 3678 3670 3678 3670 3718 3718 3716 3774 3774 37750	3319 3327 7 3591 3599 3607 3615 3623 3631 3639 3647 3655 3663 3671 3679 3687 3703 3711 3719 3727 3735 3743 3751
6360 6370 7000 7010 7020 7030 7050 7050 7050 7050 7050 7110 7110 711	3304 3312 3320 0 0 3584 3592 3600 3608 3616 3624 3632 3640 3656 3664 3656 3664 3656 3664 3674 3712 3720 3728 3736 3732	1 3321 3321 3321 3321 3321 3585 3593 3609 3617 3625 3633 3661 3663 3663 3663 3663 3667 36657 36657 36657 3667 3705 3713 3721 3729 3737 3745	2 33314 3322 2 3586 3594 3602 3610 3618 3626 3634 3662 3650 3658 36642 3650 3658 36642 3650 3658 3664 3674 3682 3690 3678 3798 3714 3712 3730 3738 3746	3315 3323 3587 3595 3603 3611 3619 3635 3643 3651 3655 3663 3691 3659 3663 3691 3699 3697 3707 3715 3723 3731	3316 3324 3324 3324 3324 3324 3588 3596 36012 3612 3620 3644 3652 3664 3652 3664 3656 3676 3670 3708 3716 3724 3743 3748	3317 3325 5 3589 3597 3605 3613 3621 3623 3663 3663 3663 3663 3663	3318 3326 3326 3326 3590 3598 3606 3614 3623 3614 3623 3638 3646 3654 3678 3686 3678 3678 3710 3718 3726 3734 3743 3750	3319 3327 7 3591 3599 3607 3615 3623 3623 3623 3667 3667 3679 3687 3679 3687 3679 3677 3679 3673 3771 3707 3719
6360 6370 7000 7010 7020 7030 7030 7030 7030 7030 7030 703	3304 3312 3320 3320 3584 3592 3600 3608 3616 3624 3632 3640 3656 3664 3656 3664 3656 3664 3656 3704 3712 3720 3728 3736 3774 3752 3756	1 3313 3321 1 3585 3593 3609 3617 3625 3633 3641 3649 3657 3665 3663 36657 3765 3775 3775 3775 37761 3769	3314 3322 2 3586 3594 3594 3602 3610 3610 3618 3626 3634 3626 3634 3658 3666 3674 3688 3764 3714 3722 3730 3738 37762 3770	3315 3323 3587 3595 3603 3611 3619 3627 3643 3651 3667 3667 3667 3667 3667 3667 3667 366	3316 3324 4 3588 3596 3604 3612 3620 3628 3636 3644 3652 3668 3668 3668 3668 3668 3668 3668 3700 3708 3716 3772 3740 3774	3317 3325 5 3589 3597 3605 3613 3629 3637 3645 3661 3669 3677 3685 3669 3701 3709 3717 3725 3733 3741 3745 3757 3755 3773	3318 3326 6 3590 3598 3606 3614 3630 3638 3646 3654 3654 3654 3654 3654 3670 3678 3654 3710 3718 3726 3714 3758 3756 3774	3319 3327 7 3591 3599 3607 3655 3633 3631 3655 3663 3667 3665 3703 3711 3719 3717 3727 3735 3743 3751 3759 3767 3775
6360 6370 7000 7010 7030 7030 7030 7030 7030 7	3304 3312 3320 0 3584 3592 3600 3608 3656 3664 3656 3664 3656 3664 3656 3664 3656 3664 3672 3680 3680 3696 3704 3712 3720 3728 3776	1 3383 3321 1 3585 3593 3601 3609 3617 3625 3633 3641 3649 3657 3665 3713 3705 3713 3721 3725 3737 3745 3759 3761 3769 3777	3314 3322 2 3586 3594 3602 3610 3614 3626 3634 3658 3658 3658 3658 3666 3674 3754 3750 3778	3315 3323 3587 3595 3603 3611 3619 3667 3643 3659 3667 3675 3723 3707 3715 3723 3707 3715 3723 3771 3739 3747 3775	3316 3324 4 3588 3596 3604 3612 3620 3660 3663 3663 3660 3668 3660 3668 3660 3668 3660 3668 3660 3668 3670 3700 3770 3770 3770 3774 3772 3780	3317 3325 5 5 3589 3597 3605 3613 3621 3629 3633 3645 3653 3661 3663 3661 3665 3677 3645 3701 3709 3717 3725 3733 3771 3741 3745 3773 3773 3773	3318 3326 6 3590 3598 3606 3614 3638 3646 3654 3662 3670 3678 3686 3694 3702 3718 3712 3718 3742 3750 3778 3774 37782	3319 3327 7 3591 3599 3607 3615 3623 3631 3639 3647 3655 3663 3671 3679 3685 3703 3771 3703 3771 3775 3759 3759 3759 3759 3759 3759 3759
6360 6370 7000 7010 7030 7030 7040 7050 7060 7060 7060 7110 7110 7110 7110 711	3304 3312 3320 3320 3584 3592 3600 3608 3614 3652 3640 3652 3664 3652 3664 3652 3664 3652 3664 3752 3726 3726 3734 3752 3744 3752 3744 3752 3744 3754 3765 3764 3776	1 3313 3321 3321 1 3585 3593 3601 3609 3617 3625 3633 3641 3649 3657 3665 3665 3665 3665 3765 3775 3721 3729 3745 3753 3761 3769 3777 3785	3314 3322 2 3586 3594 3602 3610 3618 3626 3634 3642 3658 3666 3658 3666 3674 3658 3769 3708 3714 3722 3730 3778 3776	3315 3323 3587 3595 3603 3611 3619 3627 3643 3651 3667 36675 3663 3667 36675 36673 3667 3675 3723 3707 3715 3723 37747 3755 3763 3771	3316 3324 4 3588 3596 3604 3612 3620 3664 3663 3664 3664 3664 3666 3664 3666 3664 3660 3708 3716 3724 3720 3724 3758	3317 3325 5 3589 3597 3605 3613 3629 3637 3645 3663 3663 36645 3663 36645 3663 3661 36645 3663 3701 3709 3717 3725 3733 37741 37749 3757 3743 37741 37749	3318 3326 6 3590 3598 3606 3614 3630 3638 3646 3654 3654 3654 3654 3654 3670 3678 3694 3710 3718 3726 3714 3758 3756 3774	3319 3327 7 3591 3599 3607 3655 3633 3631 3655 3663 3667 3665 3703 3711 3719 3717 3727 3735 3743 3751 3759 3767 3775
6360 6370 7000 7010 7030 7030 7030 7030 7030 7	3304 3312 3320 0 3584 3592 3600 3608 3616 3624 3632 3640 3648 3656 3664 3656 3664 3656 3664 3656 3664 3656 3672 3728 3779 3728 3776 3778 3778	1 3383 3321 1 3585 3593 3601 3609 3617 3625 3633 3641 3649 3657 3665 3713 3705 3713 3721 3725 3737 3745 3759 3761 3769 3777	3314 3322 2 3586 3594 3602 3610 3614 3626 3634 3658 3658 3658 3658 3658 3666 3674 3754 3730 3778	3315 3323 3587 3595 3603 3611 3619 3627 3635 3643 3659 3667 3675 3663 3667 3675 3767 3775 3773 3773	3316 3324 4 3588 3596 3604 3612 3620 3660 3663 3663 3660 3668 3660 3668 3660 3668 3660 3668 3660 3668 3670 3700 3770 3770 3770 3774 3772 3780	3317 3325 5 5 3589 3597 3605 3613 3621 3629 3633 3645 3653 3661 3663 3661 3665 3677 3645 3701 3709 3717 3725 3733 3741 3745 3773 3765 3773 3781	3318 3326 6 3590 3598 3606 3614 3623 3630 3638 3646 3654 3652 3670 3678 3662 3670 3718 3726 3710 3718 3726 3774 3725 3750 3758 3766 3774 3782 3790	3319 3327 7 3591 3593 3607 3615 3623 3631 3639 3647 3655 3663 3663 3663 3663 3663 3663 366
6360 6370 7000 7010 7030 7040 7050 7050 7050 7050 7050 7050 705	3304 3312 3320 0 3584 3592 3600 3608 3614 3652 3640 3652 3643 3652 3664 3652 3664 3652 3664 3764 3712 3720 3778 3776 3776 3776 3776	1 3313 3321 3321 3321 3321 3321 3321 3321 3585 3593 3601 3609 3609 3617 3625 3633 3641 3649 3657 3665 3665 3665 3665 3665 3665 3665 3665 3665 3665 3765 3775 3775 3765 3775 3765 3775 3775 3785 3785 3789 3809	3314 3322 2 3586 3594 3586 3594 3602 3610 3610 3618 3626 3634 3642 3658 3666 3674 3658 3666 3674 3658 3666 3674 36698 3706 3714 3722 3730 3778 3774 37754 3776 3778 3778 3778	3315 3323 3587 3595 3603 3611 3619 3627 3635 3643 3659 3767 3675 3683 3691 3707 3715 3723 3771 3755 3747 3775 3767 3777 3775 3767 3777	3316 3324 4 3588 3596 3604 3612 3664 3632 3644 3652 3664 3668 3664 3668 3676 3668 3724 3770 37708 3716 3772 3740 3778 3776 3776 3776	3317 3325 5 5 3589 3597 3605 3613 3621 3629 3637 3645 3653 3645 3653 3645 36653 3661 3769 3771 3725 3773 37741 37757 3755 37773 3781 3781	3318 3326 6 3590 3598 3606 3614 3623 3630 3638 3646 3654 3654 3654 3670 3678 3662 3670 3678 3662 3670 3718 3726 3710 3718 3726 3734 3726 3750 3758 3766 3774 37782 3790 3798	3319 3327 7 3591 3593 3607 3615 3623 3631 3633 3631 3633 3647 3655 3663 36631 3663 37611 3769 3763 37751 37755 37743 37755 37755 37783 37751 37799 38057
6360 6370 7000 7020 7030 7030 7050 7050 7050 7050 7050 7100 7100 7110 711	3304 3312 3320 0 3584 3592 3600 3608 3616 3624 3612 3648 3656 3664 3656 3664 3656 3664 3656 3664 3656 3664 3712 3720 3714 3712 3720 3712 3726 3776 3776 3776 37782 3776	1 3313 3321 3321 3321 3321 3321 3321 3585 3593 3601 3609 3657 3665 3663 3661 3669 3657 3765 3713 3721 3725 3793 3761 3769 3777 3785 3793 3801	3314 3322 2 3586 3594 3602 3610 3610 3634 3642 3658 3658 3658 3658 3668 3674 3642 3658 3668 3674 3768 3776 3778 3778 3778 3778	3315 3323 3587 3595 3603 3611 3619 3627 3635 3643 3659 3667 3675 3663 36691 3659 3667 3707 3715 3707 3715 3723 3771 3779 3777 3775 3779	3316 3324 4 3588 3596 3604 3612 3620 3660 3668 3663 3660 3668 3660 3668 3660 3668 3660 3668 3660 3668 3700 3700 37700 37700 37700 37700 37700 3774 37700 3774 37760 3774 37760 3788 3776	3317 3325 5 5 3589 3597 3605 3613 3621 3629 3637 3645 3663 3661 3669 3667 3665 3663 3761 3709 3717 3725 3733 3741 3749 3757 3773 37741 37781 37781 3789 3797	3318 3326 6 3590 3598 3606 3614 3622 3630 3638 3646 3654 3662 3670 3638 3646 3654 3662 3670 3678 3768 3702 3718 3726 3774 3750 3774 37782 37790 3798	3319 3327 7 7 3591 3599 3607 3615 3623 3663 3663 3663 3663 3663 3663 366

	0	1	2	3	4	5	6	7	
6400	3328	3329	3330	3331	3332	3333	3334	3335	6000 3072
6410	3336	3337	3338	3339	3340	3341	3342	3343	
6420	3344	3345	3346	3347	3348	3349	3350	3351	6777 3583
6430	3352	3353	3354	3355	3356	3357	3358	3359	(Octal) (Decim
6440	3360	3361	3362	3363	3364	3365	3366	3367	
	3368		3370	3371	3372	3373	3374	3375	
6460	3376		3378					3383	Octal Decimo
6470	3384		3386		3388				etter beening
									10000 - 4096
6500	3392	3393	3394	3395	3396	3397	3398	3399	20000 - 8192
	3400			3403			3406		30000 - 12288
	3408			3411			3414		40000 - 16384
	3416								50000 - 20480
			3410	3419		3421		3423	60000 - 24576
	3424			3427		3429		3431	70000 - 28672
	3432			3435		3437		3439	
	3440		3442		3444		3446	3447	
6570	3448	3449	3450	3451	3452	3453	3454	3455	
6600	3456		3458	3459	346 0	3461	3462	3463	
5610	3464	3465	3466	3467	3468	3469	3470	3471	
6620	3472		3474	3475	3476	3477	3478	3479	
6630	3480		3482		3484		3486	3487	
	3488		3490	3491		3493		3495	
	3496			3499			3502	3503	
6660	3504	3505	3506	3507	3508		3510	3511	
6670	3512		3514	3515	3516		3518	3519	
5010	3312	2213	3314	2212	3510	3517	3310	3319	
5700	3520	3521	3522	3523	3524	3525	3526	3527	
5710	3528	3529	3530	3531	3532	3533		3535	
5720		3537		3539	3540	3541	3542	3543	
5730		3545		3547			3550	3551	
5740		3553			3556		3558	3559	
5750				3563		3565		3567	
				9909					
		25.60	3570	3571	3577				
5760	3568			3571 3579		3573 3581		3575	
6760			3570 3578			3573	3574	3583	
6760	3568 3576	3577	3578	3579	3580	3581			
5760	3568 3576 0	3577	3578 2	3579 3	3580	3581 5	3582 6	<u>3583</u> 7	
5760 5770	3568 3576 0 3840	3577 1 3841	3578 2 3842	3579 3 3843	3580 4 3844	3581 5 3845	3582 6 3846	3583 7 3847	7000 3584
5760 5770 7400 7410	3568 3576 0 3840 3848	3577 1 3841 3849	3578 2 3842 3850	3579 3 3843 3851	3580 4 3844 3852	3581 5 3845 3853	3582 6 3846 3854	3583 7 3847 3855	to to
5760 5770 7400 7410	3568 3576 0 3840 3848	3577 1 3841	3578 2 3842 3850 3858	3579 3 3843 3851 3859	3580 4 3844 3852 3860	3581 5 3845 3853 3861	3582 6 3846 3854 3862	3583 7 3847 3855 3863	
5760 5770 7400 7410 7420	3568 3576 0 3840 3848	3577 1 3841 3849	3578 2 3842 3850 3858	3579 3 3843 3851	3580 4 3844 3852 3860	3581 5 3845 3853 3861	3582 6 3846 3854 3862	3583 7 3847 3855 3863	to to
5760 5770 7400 7410 7420	3568 3576 0 3840 3848 3856	3577 1 3841 3849 3857	3578 2 3842 3850 3858 3866	3579 3 3843 3851 3859	3580 4 3844 3852 3860 3868	3581 5 3845 3853 3861 3869	3582 6 3846 3854 3862	3583 7 3847 3855 3863 3871	to to 7777 4095
5760 5770 7400 7410 7420 7430 7440	3568 3576 0 3840 3848 3856 3864 3872	3577 1 3841 3849 3857 3865 3873	3578 2 3842 3850 3858 3866	3579 3 3843 3851 3859 3867 3875	3580 4 3844 3852 3860 3868 3876	3581 5 3845 3853 3861 3869 3877	3582 6 3846 3854 3862 3870	3583 7 3847. 3855 3863 3871 3879	to to 7777 4095
5760 5770 7400 7410 7420 7430 7430 7440 7450	3568 3576 0 3840 3848 3856 3864 3872 3880	3577 1 3841 3849 3857 3865 3873 3881	3578 2 3842 3850 3858 3866 3874 3882	3579 3 3843 3851 3859 3867 3875 3883	3580 4 3844 3852 3860 3868 3876 3884	3581 5 3845 3853 3861 3869 3877 3885	3582 6 3846 3854 3862 3870 3878 3886	3583 7 3847. 3855 3863 3871 3879	to to 7777 4095
5760 5770 7400 7410 7420 7430 7430 7440 7450 7460	3568 3576 0 3840 3848 3856 3864 3872 3880 3888	3577 1 3841 3849 3857 3865 3873 3881	3578 2 3842 3850 3858 3866 3874 3882 3890	3579 3843 3851 3859 3867 3875 3883 3891	3580 4 3844 3852 3860 3868 3876	3581 5 3845 3853 3861 3869 3877 3885 3893	3582 6 3846 3854 3862 3870 3878	7 3 847 3 855 3 863 3 871 3 879 3 887 3 895	to to 7777 4095
7400 7410 7420 7430 7430 7440 7450 7460 7470	3568 3576 0 3840 3848 3856 3864 3872 3880 3888 3896	3577 1 3841 3849 3857 3865 3873 3881 3889 3897	3578 2 3842 3850 3858 3866 3874 3882 3890 3898	3579 3 3843 3851 3859 3867 3875 3883 3891 3899	3580 4 3844 3852 3860 3868 3876 3884 3892 3900	3581 5 3845 3853 3861 3869 3877 3885 3893 3901	3582 6 3846 3854 3854 3870 3878 3886 3894 3902	7 3847. 3855 3863 3871 3879 3887 3895 3903	to to 7777 4095
7400 7400 7410 7420 7430 7430 7440 7450 7450 7460 7470	3568 3576 0 3840 3848 3856 3864 3872 3880 3888 3896 3904	3577 1 3841 3849 3857 3865 3873 3881 3889 3897 3905	3578 2 3842 3850 3858 3856 3874 3882 3890 3898 3996	3579 3 3843 3851 3859 3867 3875 3883 3891 3899 3907	3580 4 3844 3852 3860 3868 3876 3884 3892 3900 3908	3581 5 3845 3853 3861 3869 3877 3885 3893 3901 3909	3582 6 3846 3854 3854 3870 3878 3886 3894 3902 3910	7 3847. 3855 3863 3871 3879 3887 3895 3903 3911	to to 7777 4095
5760 5770 7400 7410 7420 7430 7440 7450 7450 7450 7450 7450 7510	3568 3576 0 3840 3848 3856 3864 3872 3880 3888 3896 3904 3912	1 3841 3849 3857 3865 3873 3881 3889 3897 3905 3913	3578 2 3842 3850 3858 3866 3874 3882 3890 3898 3906 3914	3579 3843 3851 3859 3867 3875 3883 3891 3899 3907 3915	3580 4 3844 3852 3860 3868 3876 3884 3892 3900 3908 3916	3581 5 3845 3853 3861 3869 3877 3885 3893 3901 3909 3917	6 3846 3854 3854 3862 3870 3878 3886 3894 3902 3910 3918	7 3847 . 3855 3863 3871 3879 3887 3895 3903 3911 3919	to to 7777 4095
7400 7410 7410 7410 7420 7430 7440 7450 7450 7450 7470 7510 7520	3568 3576 3576 0 3840 3848 3856 3864 3872 3880 3888 3896 3896 3904 3912 3920	1 3841 3849 3857 3865 3873 3881 3889 3897 3905 3913 3921	2 3842 3850 3858 3866 3874 3882 3890 3898 3906 3914 3922	3 3 3843 3851 3859 3867 3875 3883 3891 3899 3907 3915 3923	4 3844 3852 3860 3868 3876 3884 3892 3900 3908 3916 3924	3581 5 3845 3853 3861 3869 3877 3885 3893 3901 3909 3917 3925	6 3846 3854 3854 3854 3862 3870 3878 3886 3894 3902 3910 3918 3926	7 3847 . 3855 3863 3871 3879 3887 3895 3903 3911 3919 3927	to to 7777 4095
7400 7410 7420 7430 7440 7450 7460 7460 7470 7500 7510 7520 7530	3568 3576 0 3840 3848 3856 3864 3872 3880 3888 3896 3904 3912 3920 3928	3577 1 3841 3849 3857 3865 3873 3889 3897 3905 3913 3921 3929	2 3842 3850 3858 3866 3874 3882 3890 3898 3906 3914 3922 3930	3579 3843 3851 3859 3867 3875 3883 3891 3899 3907 3915 3923 3931	3580 4 3844 3852 3860 3868 3876 3884 3892 3900 3908 3916 3924 3932	3581 5 3845 3853 3861 3869 3877 3885 3893 3901 3909 3917 3925 3933	3582 6 3846 3854 3854 3878 3878 3878 3878 3878 3894 3902 3910 3918 3926 3934	3583 7 3847. 3855 3863 3871 3879 3887 3895 3903 3911 3919 3927 3935	to to 7777 4095
7400 7410 7420 7430 7440 7450 7460 7460 7470 7550 7550 7550 7550 7550	3568 3576 3840 3848 3856 3864 3872 3880 3888 3896 3904 3920 3920 3928 3936	1 3841 3849 3857 3865 3873 3881 3889 3897 3905 3913 3921 3921 3929 3937	2 3842 3850 3858 3856 3874 3890 3898 3906 3914 3922 3930 3938	3 3 3843 3851 3859 3867 3875 3883 3891 3899 3907 3915 3923 3931 3939	4 3844 3852 3860 3868 3876 3884 3892 3900 3908 3916 3924 3932 3940	3581 5 3845 3853 3861 3869 3877 3885 3893 3901 3909 3917 3925 3933 3941	3582 6 3846 3854 3854 3878 3878 3878 3878 3878 3894 3902 3910 3918 3926 3934 3942	3583 7 3847 3855 3863 3871 3879 3887 3895 3903 3911 3919 3927 3935 3943	to to 7777 4095
7400 7410 7410 7430 7440 7440 7440 7450 7460 7440 7450 7460 7470 7510 7550 7550	3568 3576 0 3840 3848 3856 3864 3856 3864 3856 3888 3896 3994 3912 3920 3928 3936 3934	3577 1 3841 3849 3857 3865 3873 3881 3889 3897 3905 3913 3921 3921 3921 3923 3921 3923 3945	3578 2 3842 3850 3858 3866 3874 3882 3890 3898 3906 3914 3922 3930 3938 3946	3 3 3843 3851 3859 3867 3875 3883 3891 3899 3907 3915 3923 3931 3939 3947	3580 4 3844 3852 3860 3868 3876 3884 3892 3900 3908 3916 3924 3924 3932 3940 3948	3581 5 3845 3853 3861 3869 3877 3885 3893 3901 3909 3917 3925 3931 3941 3949	3582 6 3846 3854 3854 3870 3878 3870 3878 3886 3894 3902 3910 3918 3926 3934 3934 3950	3583 7 3847. 3855 3863 3871 3879 3887 3895 3903 3911 3919 3927 3935 3943 3951	to to 7777 4095
7400 7410 7410 7430 7440 7440 7440 7450 7460 7440 7450 7460 7470 7510 7550 7550	3568 3576 0 3840 3848 3856 3864 3856 3864 3872 3880 3888 3896 3904 3912 3920 3928 3934 3934	1 3841 3849 3857 3865 3873 3885 3889 3897 3905 3913 3921 3929 3937 3945 3945	3578 2 3842 3850 3858 3866 3874 3882 3890 3898 3906 3914 3922 3930 3938 3946	3 3 3843 3851 3859 3867 3875 3883 3891 3899 3907 3915 3923 3931 3939	3580 4 3844 3852 3860 3868 3876 3884 3892 3900 3908 3916 3924 3924 3932 3940 3948	3581 5 3845 3853 3861 3869 3877 3885 3893 3901 3909 3917 3925 3933 3941	3582 6 3846 3854 3854 3878 3878 3878 3878 3878 3894 3902 3910 3918 3926 3934 3942	3583 7 3847 3855 3863 3871 3879 3887 3895 3903 3911 3919 3927 3935 3943	to to 7777 4095
7400 7410 7410 7430 7440 7440 7440 7440 7440 7440 744	3568 3576 0 3840 3848 3856 3864 3856 3864 3856 3888 3896 3994 3912 3920 3928 3936 3934	1 3841 3849 3857 3865 3873 3885 3889 3897 3905 3913 3921 3929 3937 3945 3945	2 3842 3850 3858 3858 3866 3874 3890 3996 3914 3922 3930 3938 3946 3954	3 3 3843 3851 3859 3867 3875 3883 3891 3899 3907 3915 3923 3931 3939 3947	4 3844 3852 3860 3868 3876 3884 3892 3900 3908 3916 3924 3932 3948 3948 3956	3581 5 3845 3853 3861 3869 3877 3885 3893 3901 3909 3917 3925 3931 3943	3582 6 3846 3854 3862 3870 3878 3886 3894 3902 3910 3918 3926 3934 3926 3934 3925 3958	3583 7 3847. 3855 3863 3871 3879 3887 3895 3903 3911 3919 3927 3935 3943 3951	to to 7777 4095
7400 7400 7410 7420 7430 7430 7430 7450 7450 7450 7550 7550 7550 7550 7550 7550	3568 3576 0 3840 3848 3856 3864 3856 3864 3872 3880 3888 3896 3904 3912 3920 3928 3934 3934	1 3841 3849 3857 3865 387 3889 3897 3905 3913 3929 3937 3945 3953 3945	2 3842 3850 3858 3850 3858 3866 3874 3898 3906 3914 3922 3930 3922 3930 3945 3954 3954	3 3 3843 3851 3859 3867 3875 3883 3891 3899 3907 3915 3923 3931 39347 3955	4 3844 3852 3860 3868 3876 3884 3892 3900 3908 3916 3924 3932 3948 3948 3956	3581 5 3845 3853 3861 3869 3909 3917 3925 3933 3941 3949 3957 3965	3582 6 3846 3854 3862 3870 3878 3886 3894 3902 3910 3918 3926 3934 3926 3934 3925 3958	7 3847 3855 3863 3871 3895 3903 3911 3919 3927 3935 3943 3951 3959	to to 7777 4095
7400 7410 7410 7420 7430 7440 7430 7440 7450 7550 7550 7550 7550 7550 755	0 0 3840 3840 3848 3856 3864 3856 3864 3872 3860 3904 3912 3920 3928 3934 3934 3952 3960 3968	1 3841 3849 3857 3865 3873 3881 3889 3995 3913 3921 3929 3937 3945 3953 3951 3953	2 3842 3850 3858 3866 3874 3882 3890 3898 3906 3914 3922 3930 3938 3946 3954 3954 3954 3954	3579 3843 3851 3859 3867 3875 3883 3891 3939 3939 3939 3939 3939 3939	4 3844 3852 3860 3868 3876 3984 3876 3984 3990 3908 3916 3924 3940 3948 3956 3964 3956	3581 5 3845 3853 3861 3869 3877 3885 3893 3901 3925 3933 3941 3949 3957 3965 3973	6 3846 3854 3854 3870 3878 3878 3878 3894 3902 3910 3918 3926 3934 3942 3942 3950 3958 3942 3958 3946	7 3847 3847 3855 3863 3871 3879 3887 3903 3911 3919 3927 3943 3943 3943 3943 3943 3943 3943	to to 7777 4095
7400 7410 7410 7430 7430 7440 7430 7440 7430 7440 744	0 3840 3848 3856 3864 3856 3864 3856 3864 3872 3880 3888 3994 3912 3912 3912 3912 3912 3912 3912 3912	1 3841 3849 3857 3865 3873 3865 3873 3885 38873 3887 3905 3913 3921 3929 3937 3945 3945 3945 3945 3945 3945	2 3842 3850 3858 3866 3874 3882 3890 3898 3906 3914 3922 3930 3938 3946 3954 3954 3954 3954	3579 3843 3851 3859 3867 3875 3883 3891 3897 3915 3923 3931 3939 3947 3947 3955 3963 3971	4 3884 3862 3860 3868 3876 3884 3892 3900 3908 3916 3924 3932 3940 3948 3956 3964 3972 3980	3581 5 3845 3853 3869 3877 3885 3893 3909 3917 3925 3933 3941 3949 3957 3955 3957 3981	3582 6 3846 3854 3854 3870 3878 3878 3878 3878 3878 3902 3910 3918 3992 3958 3958 3958 3958 3958	3583 7 7 3847 3855 3863 3871 3879 3887 3893 3903 3903 3911 3927 3935 3943 3951 3951 3951 3953 3951 3953	to to 7777 4095
7400 7410 7410 7410 7420 7440 7440 7440 7450 7440 7450 7440 7550 755	0 3840 3848 3856 3864 3856 3864 3856 3860 3888 3896 3912 3928 3928 3936 3928 3936 3954 3950 3960	1 33841 3849 3857 3865 3873 3889 3995 3913 3921 3921 3921 3937 3945 3953 3965 3997	2 38578 3842 3850 3858 3866 3874 3882 3890 3898 3904 3930 3938 3945 3952 3956 3952 3956 3954 3952 3970 3978 3986	3579 3843 3851 3859 3867 3875 3883 3891 3993 3939 3947 3955 3953 3953 3954 3955	4 3844 3852 3860 3868 3876 3884 3990 3990 39916 3924 39340 3948 3956 3954 3956 3956	3581 5 3845 3853 3861 3869 3877 3909 3917 3925 3933 3941 3949 3957 3941 3949 3957 3981 3989	3582 6 3846 3854 3862 3870 3878 3886 3884 3992 3910 3918 3926 3934 3923 3942 3950 3942 3958 3942 3958	3583 7 7 8847 3855 3863 3863 3871 3879 3993 3991 3911 3919 3923 3943 3951 3953 3951 3957 3967 3983 3991	to to 7777 4095
7400 7410 7410 7420 7440 7450 7450 7450 7460 7470 7550 7550 7550 7550 7550 7550 755	0 3840 3840 3848 3856 3864 3856 3863 3872 3880 3893 3912 3920 3928 3934 3952 3944 3952 3944 3952 3944 3954 3946	1 3841 3849 3857 3857 3857 3865 3873 3889 3889 39913 3921 3923 3937 3945 3953 3953 3951 3953 3951 3953 3951	2 3842 3850 3858 3858 3858 3858 3858 3858 3858	3579 3843 3851 3859 3867 3875 3883 3891 39915 3923 3931 3939 3939 3939 3947 3955 3963 3971 3979	3580 4 3844 3852 3860 3886 3886 38876 3988 39900 3908 3916 3924 3930 3948 3956 3956 3956 3958 3998	3581 5 3845 3853 3861 3869 3877 3885 3893 3901 3909 3917 3925 3933 3941 3949 3957 3945 3949 3957 3945	3582 6 3846 3854 3862 3862 3862 3862 3934 3902 3910 3918 3926 3934 3942 3950 3958 3956 3958 3956 3958 3958	3583 7 7 3847 3855 3863 3863 3887 3887 3903 3911 3919 3927 3943 3943 3951 3959 3967 3959 3967 3991	to to 7777 4095
7400 7410 7420 7440 7440 7440 7440 7450 7440 7450 7460 7550 7550 7550 7550 7550 7550 7550 75	0 0 3840 3845 3856 3864 3856 3864 3872 3880 3904 3912 3920 3912 3920 3912 3923 3934 3952 3956 3968 3976	1 3841 3849 3857 3865 3873 3889 39913 3921 3921 3937 3945 3953 3961 3963 3961 3969 3977 3985 3997	3578 2 3842 3850 3858 3866 3874 3890 3994 3992 3938 3946 3954 3954 3954 3970 3978 3986 3970	3579 3843 3851 3859 3867 3875 3883 3899 3907 3915 3939 3939 3939 3939 3939 3939 3939	4 3844 3852 3860 3868 3876 3990 3916 3924 3940 3948 3956 3956 3956 3958 3956	3581 5 3845 3853 3861 3869 3877 3985 3933 3901 3941 3949 3947 3965 3981 3989 3987	3582 6 3846 3854 3862 3870 3878 3886 3884 3992 3910 3918 3926 3934 3950 3954 3950 3954 3950 3954 3950 3954 3950 3954 3954 3956 39574 3956 39574 395755 39575 39575 39575 395755 39575 39575 395755 395755 395755 3	3583 7 38477 3855 3863 3863 3887 3895 3903 3911 3919 39275 3943 3951 3953 3957 3953 3957 39975 3993 39991 39991	to to 7777 4095
7400 7410 7420 7440 7440 7440 7440 7440 7450 7460 7460 7460 7550 7550 7550 7550 7550 7550 7550 75	3568 3576 0 3840 3848 3856 3864 3864 3888 3886 3888 3904 3912 3920 3928 3936 3944 3952 3956 3954 3956 3968 3976 3968 3976	1 3841 3849 3857 3865 3873 3881 3889 3913 3921 3923 3945 3953 3945 3953 3945 3953 3946 3953 3945 3961 3961 4009	3578 2 3842 3850 3858 3866 3874 3890 3994 3992 3938 3946 3954 3954 3954 3954 3954 3958 3970 3978 3978 3978	3579 3843 3851 3859 3867 3875 3883 3891 3915 3923 3931 3939 3947 3955 3963 3979 3987 3987 3987	3580 4 3844 3852 3860 3868 3876 3990 3916 3923 3940 3948 3956 3954 3956 3956 3988 3996	3581 5 3845 3853 3861 3869 3877 3925 3933 3901 3909 3917 3925 3953 3941 3945 3957 3965 3981 3989 3987	6 3846 3854 3854 3854 3854 3854 3854 3854 3854	7 3847, 3855, 3863, 3871, 3879, 3879, 3903, 3911, 3919, 3927, 3935, 3943, 3951, 3959, 3943, 3959, 3959, 3959, 3943, 3959, 3959, 3943, 3959,	to to 7777 4095
7400 7410 7410 7410 7410 7410 7410 7410	3568 3576 3876 3840 3848 3856 3856 3856 3888 3888 3888 3888 39896 3914 3912 3920 3928 3928 3928 3926 3952 3952 3954 3952 3954 3954 3954 3954 3954 3954 3954 3954	1 3841 3843 3849 3857 3865 3873 3881 3889 3897 3905 3913 3921 3923 3945 3953 3945 3953 3945 3995 3993 4001	2 3842 3850 3858 3856 3858 3882 3890 3914 3922 3930 3938 3946 3954 3954 3954 3954 3954 4002 4010	3579 3843 3851 3859 3867 3875 3883 3891 3937 3915 3923 3931 3939 3947 3955 3963 3971 3979 3987 3987 3987	4 3844 3852 3860 3868 3876 3884 3892 3900 3908 3916 3924 3932 3940 3956 3954 3954 3956 4004 4012	3581 5 3845 3853 3861 3863 3877 3885 3893 3901 3917 3925 3933 3949 3957 3949 3957 3949 3957 3989 3989 3989 3989 39897 4013	6 3846 3854 3854 3854 3854 3854 3870 3957 3918 3926 3934 3918 3926 3934 3950 3958 3956 3956 3956 3950 3958 3956 4014 1022	7 38477 3855 3855 3863 3871 3855 3863 3871 3855 3903 3919 3927 3927 3925 3951 3951 3959 3951 3951 3955 3957 3957 3959 3957 39	to to 7777 4095
7400 7410 7410 7430 7430 7440 7450 7450 7550 7550 7550 7550 755	3568 3576 3876 3840 3848 3856 3856 3864 3872 3888 3896 3904 3912 3928 3936 3944 3952 3954 3954 3954 3954 3954 3954 3954 3954	1 3841 3849 3857 3865 3873 3889 3995 3995 3995 3921 3929 3937 3945 3953 3953 3953 3953 3961 3985 3993 4001 4002	2 3842 3850 3858 3850 3858 3890 3898 3914 3914 3914 3914 3914 3954 3954 3954 3954 3956 3978 3978 3986 3994 4010 4018 4026	3579 3843 3851 3859 3867 3875 3883 3897 3913 3931 3939 3947 3955 3963 3971 3979 3957 3975 3963	4 3844 3852 3860 3868 3876 3888 3892 3900 3918 3924 3932 3940 3956 3954 3956 3954 3956 4024 4020 4028	3581 5 3845 3853 3869 3877 3865 3893 3901 3909 3917 3925 3933 3941 3949 3957 3965 3957 3985 3989 3997 4005	6 3846 3854 3854 3867 3870 3870 3870 3910 3918 3926 3934 3958 3958 3958 3958 3958 3958 3958 3958	7 3847 3847 3855 3863 3871 3879 3887 3903 3911 3919 3927 3953 3953 3953 3959 3957 3959 3957 3959 3959	to to 7777 4095
7400 7410 7420 7430 7450 7450 7450 7550 7550 7550 7550 755	3568 3576 0 3840 3848 3856 3864 3856 3868 3886 3886 3886 3904 3912 3920 3928 3912 3920 3928 3944 3952 3944 3950 3968 3976 3984 4000 4002	1 3841 3849 3857 3865 3913 3887 3929 3937 3945 3953 3953 3953 3953 3953 3953 3954 3959 3957 3953 3954 3953 3954 3953 3954 3953 3953	2 3842 3850 3858 3850 3858 3866 3874 3882 3930 3938 3914 3922 3930 3938 3954 3954 3954 3954 3954 3954 3954 4002 4010 4018 4026	3579 3843 3851 3859 3867 3875 3883 3997 3915 3923 3931 3939 3947 3955 3963 3955 3963 3957 3955 3963 3971 3979 3987 4003 4014 4019	4 3844 3852 3860 3868 3876 3884 3852 3868 3876 3874 3990 3990 3990 3990 3990 3990 3990 3990 3992 3956 3954 3952 3956 3954 3952 3956 3952 3956 3952 3956 3952 3956 3952 3956 3952 3956 3952 3956 3952 3956 3956 3952 3956 3952 3956 3952 3956 3952 3956 3952 3956 3952 3956 3952 3956 4004 4002 4020 4000 400	3581 5 3845 3853 3861 3869 3877 3885 3893 39917 3925 3933 3931 3941 3949 3957 3945 3945 3941 3945 3957 3985 3997 4005 4013	6 3846 3854 3854 3854 3870 3978 3990 3910 3918 3992 3934 3958 3958 3958 3958 3958 3958 3958 4006 4014 4022 4030	7 38847 38477 3855 3863 3871 3879 3873 3873 3903 3911 3919 3927 3943 3959 3959 3959 3959 3959 3959 4007 4015 4023 4023	to to 7777 4095
7400 7410 7410 7410 7420 7440 7440 7450 7440 7450 7550 7550 755	3568 3576 3840 3848 3856 3864 3856 3864 3878 3880 3880 3880 3880 3984 3912 3920 3928 3934 3952 3944 3952 3944 3952 3960 3954 3954 4008 4008 4002	1 3841 3841 3849 3857 3865 3873 3865 3873 3889 3995 3921 3929 3937 3945 3945 3945 3953 3945 3995 3995 3995	2 3842 3850 3856 3874 3866 3874 3886 3890 3898 3906 3914 3922 3930 39386 3954 3954 3954 3954 3954 3986 4002 4010 4018 4022	3579 3843 3851 3857 3867 3887 3887 3887 3899 3907 3915 3923 3931 3939 3947 3955 3963 3947 3955 3963 3971 3979 4003 4003	4 3844 3852 3860 3868 3876 3888 3876 3882 3990 3916 3924 3924 3924 3924 3948 3956 3964 3958 3964 4012 4028 4004	3581 5 3845 3853 3869 3877 3855 3933 3901 3909 3917 3925 3933 3941 3925 3933 3941 3957 3965 3973 3989 3989 3989 3989 4001 3989 4001 4021 4021	6 3846 3854 3854 3870 3870 3870 3878 3894 39902 3910 3918 3926 3934 3942 3934 3945 3958 3956 3958 3956 3954 3950 3958 4066 4014 1022 4038	7 3847; 3855; 3863; 3879; 3879; 3879; 3903; 3911; 3919; 3927; 3935; 3943; 3951; 3959; 3951; 3959; 3951; 3959; 3951; 3959; 3951; 3959; 3951; 3951; 3959; 3951; 3959; 3951; 3959; 3951; 3959; 3951; 4007; 4003; 40047; 4047; 4047; 4055; 4047; 4055; 4047; 4055; 4047; 4055; 4047; 4055; 4047; 4055; 4047; 4055; 4047; 4055; 4047; 4055; 4047; 4055; 4047; 4055; 4047; 4055; 4047; 4055; 4047; 4055;	to to 7777 4095
7400 7410 7420 7430 7430 7450 7450 7450 7550 7550 7550 7550 755	3568 3576 0 3840 3848 3856 3864 3856 3864 3865 3863 3860 3888 3896 3912 3928 3912 3928 3936 3952 3936 3952 3956 3956 3956 3956 3956 4002 4008 4016 4012 4048	1 3841 3841 3849 3857 3865 3873 3889 3905 3913 3921 3929 3937 3945 3953 3953 3953 3953 3953 3961 3969 3977 3985 3993 4009 4017 4025 4034 4049	2 3842 3850 3858 3856 3874 3886 3914 3922 3930 3938 3946 3954 3954 3954 3954 3954 4002 4018 4002 4014 4042 4050	3579 3843 3851 3859 3867 3875 3887 3899 3907 3915 3923 3931 3939 3947 3955 3963 3971 3979 3957 3955 3963 3971 3979 4003 4014 4019 4027 4035	4 3844 3852 3860 3868 3876 3888 3876 3988 3990 3908 3912 3932 3932 3940 3932 3940 3956 3956 3954 3956 3956 4024 4020 4028 4034 4052	3581 5 3845 3853 3861 3869 3877 3865 3893 3901 3917 3925 3933 3941 3949 3957 3965 3957 3965 3997 4005 4013 4021 4029 4037	6 3846 3854 3862 3870 3870 3870 3870 3894 3990 3910 3918 3926 3934 3958 3958 3958 3958 3958 3958 3958 3958	7 3847 3847 3855 3863 3871 3879 3887 3903 3911 3919 3927 3935 3943 3959 3959 3959 3959 3959 3959	to to 7777 4095
7400 7410 7420 7430 7440 7430 7450 7450 7450 7450 7550 7550 7550 755	3568 3576 0 3840 3848 3856 3864 3876 3880 3880 3988 3912 3920 3912 3920 3912 3920 3912 3923 3944 3952 3953 3944 4055 4000 4002	1 3841 3849 3857 3865 3873 3865 3873 3889 3995 3993 3995 3993 3995 3993 3995 3995 3995 3995 3995 3995 3995 3997 3985 3997 3985 3997 4001 4002 4001 4002	2 3842 3850 3858 3850 3858 3866 3874 3882 39906 3914 3922 3930 3934 3930 3934 3954 3954 3954 3954 3954 4002 4010 4012 4012 4014 4026	3579 3 3 3 3 3 3 3 3 3 3 3 3 3	4 3844 3852 3860 3852 3860 3876 3876 3874 3872 3990 4000 4002 4000 4002 40000 4000 4000 4000 40000 4000 4000 4000 4000 4000 4000 4	3581 5 3845 3869 3867 3883 3893 3901 3909 3917 3925 3933 3931 3941 3949 3957 3941 3945 3965 3997 3981 3981 3987 3981 3987 3981 3965 4013 4021 4029 4037 4045 34061	53582 6 3846 3854 3854 3854 3870 3987 3987 3990 3910 3918 3992 3934 3934 3934 3958 3994 3958 3994 3958 3996 3958 39974 4039 4016 4038 4046 4052	3583 7 3847 3855 3863 3871 3873 3873 3895 3903 3911 3919 3927 3935 3943 3951 3953 3951 3953 3953 3953 3953 395	to to 7777 4095
7400 7410 7410 7420 7430 7440 7440 7440 7440 7440 7440 7450 7440 7450 7440 7450 745	3568 3576 3840 3848 3856 3864 3856 3864 3878 3880 3888 3896 3912 3920 3928 3936 3944 3952 3944 3952 3960 3954 3954 4002 4008 4004 4024 4040	1 3841 3841 3849 3857 3865 3873 3865 3873 3889 3995 4001 4009 4007 4003 4041 4049 4057 4065	2 3842 3850 3858 3866 3874 3886 3914 3922 3930 3938 3936 3934 3954 3954 3954 3956 3994 4002 4010 4018 4022 4050 4044 4050	3579 3843 3851 3857 3867 3887 3887 3913 3931 3939 3947 3955 3963 3947 3955 3963 3971 3979 3987 3987 3987 3987 4003 4014 4019 4025	4 3844 3852 3860 3868 3876 3888 3876 3988 3990 3916 3924 3932 3940 3956 3954 3956 3954 3956 4004 4022 4028 4024 4028 4044 4052 4068	3581 5 3845 3853 3869 3877 3885 3893 3901 3909 3917 3925 3933 3941 3925 3933 3941 3957 3957 3965 3973 3989 3989 3989 3989 3989 3989 4005 4021 4025 4045 4053 4069	53582 53846 3854 3854 3854 3870 3870 3870 3873 3990 3910 3918 3926 3934 3926 3934 3945 3934 3945 3934 39588 39588 3958 3958 3958 3958 3958 3958 3958 3958 39	7 3847; 3855; 3863; 3871; 3879; 3887; 3903; 3911; 3919; 3927; 3935; 3953; 3954; 3959; 3954; 3959; 3954; 3954; 3954; 3954; 3955; 3954; 3955; 3954; 3955; 3954; 3955; 3954; 3955; 3957; 3957; 3957; 3959; 4007; 4015; 4047; 4047; 4047; 40	to to 7777 4095
7400 7410 7420 7430 7430 7430 7430 7430 7430 7430 743	3568 3576 0 3840 3848 3856 3864 3856 3864 3856 3868 3856 3860 3888 3896 3912 3920 3912 3920 3912 3920 3914 3952 3946 3952 3956 3952 3956 3952 4000 4002 4000 4002 4016 4024 4016 4024 4048 4056 4072	1 3841 3841 3849 3857 3865 3873 3885 3913 3929 3937 3945 3953 3953 3953 3953 3953 3954 3953 3954 4069 4007 4025 4049 4057 4049 4057 4065 4073 4073 4075 407	2 3842 3850 3858 3856 3874 3886 3914 3922 3930 3938 3946 3934 3954 3954 3954 3954 3954 4026 4010 4018 4026 4034 4042 4058 4058 4074	3579 3 3 3 3 3 3 3 3 3 3 3 3 3	4 3844 3852 3860 3868 3876 3876 3873 3980 3916 3923 3932 3932 3932 3940 3932 3940 3956 3956 3956 3956 4024 4012 4028 4028 4028 4026 4045 4052 4056 407	3581 5 3845 3853 3869 3877 3885 3893 3901 3917 3925 3933 3931 3941 3949 3957 3957 3957 3957 3957 3957 3957 4013 4021 4029 4013 4053 4061 4053	53582 6 3846 3854 3854 3870 3870 3878 39870 3910 3918 3934 3934 3934 3934 3934 3934 3934 3958 3934 3958 4014 4052 4054 4054 4078 4	7 3847 3847 3855 3863 3871 3879 3873 3993 3911 3919 3927 3953 3959 3959 3959 3959 3959 3959 3959 4023 4023 4023 4023 4024 4055 4055 4055 4063	to to 7777 4095
7400 7410 7420 7430 7430 7430 7430 7430 7430 7430 743	3568 3576 0 3840 3848 3856 3864 3856 3864 3856 3868 3856 3860 3888 3896 3912 3920 3912 3920 3912 3920 3914 3952 3946 3952 3956 3952 3956 3952 4000 4002 4000 4002 4016 4024 4016 4024 4048 4056 4072	1 3841 3841 3849 3857 3865 3873 3885 3913 3929 3937 3945 3953 3953 3953 3953 3953 3954 3953 3954 4069 4007 4049 4049 4057 4049 4057 4063	2 3842 3850 3858 3856 3874 3886 3914 3922 3930 3938 3946 3934 3954 3954 3954 3954 3954 4026 4010 4018 4026 4034 4042 4058 4058 4074	3579 3 3 3 3 3 3 3 3 3 3 3 3 3	4 3844 3852 3860 3868 3876 3876 3873 3980 3916 3924 3932 3940 3932 3940 3940 3956 3956 3954 3956 4024 4012 4028 4026 4042 4052 4056 4056 407	3581 5 3845 3853 3869 3877 3885 3893 3901 3917 3925 3933 3931 3941 3949 3957 3957 3957 3957 3957 3957 3957 4013 4021 4029 4013 4053 4061 4053	53582 6 3846 3854 3854 3870 3870 3878 39870 3910 3918 3934 3934 3934 3934 3934 3934 3934 3958 3934 3958 4014 4052 4054 4054 4078 4	7 3847 3847 3855 3863 3871 3879 3873 3993 3911 3919 3927 3953 3959 3959 3959 3959 3959 3959 3959 4023 4023 4023 4023 4045 4055 4063 4079	to to 7777 4095

95118A, Fo

NUMERICAL INFORMATION

OCTAL-DECIMAL FRACTION CONVERSION TABLE

OCTAL	DEC.	OCTAL	DEC.	OCTAL	DEC.	OCTAL	DEC.
.000	.000000	. 100	. 125000	. 200	. 250000	. 300	.375000
.001	.001953	. 101	. 126953	. 201	. 251953	. 301	.376953
.002	. 003906	. 102	128906	. 202	. 253906	. 302	378906
.003	.005859	. 103	130859	. 203	. 255859	. 303	380859
.004	.007812	. 104	. 132812	. 204	.257812	. 304	.382812
.005	.009765	. 105	, 134765	. 205	259765	. 305	.384765
.006	.011718	. 106	. 136718	. 206	.261718	.306	.386718
.007	.013671	. 107	.138671	. 207	.263671	.307	.388671
.010	.015625	. 110	. 140625	.210	. 265625	.310	.390625
.011	.017578	.111	. 142578	.211	.267578	.311	.392578
.012	.019531	.112	. 144531	.212	.269531	.312	.394531
.012	.021484	.113	. 146484	.213	.271484	.313	.396484
.013	.023437	.114	. 148437	.213	. 273437	.314	.398437
.014					-	.315	. 400390
	.025390	.115	. 150390	.215	. 275390		
.016	.027343	.116	. 152343	.216	. 277343	.316	.402343
.017	.029296	. 117	. 154296	. 217	. 279296	.317	.404296
.020	.031250	. 120	.156250	. 220	.281250	. 320	.406250
.021	.033203	. 121	.158203	. 221	.283203	.321	.408203
.022	.035156	. 122	.160156	. 222	.285156	. 322	.410156
.023	.037109	. 123	.162109	. 223	.287109	. 323	.412109
.024	.039062	. 124	. 164062	.224	.289062	. 324	.414062
.025	.041015	. 125	.166015	. 225	.291015	. 325	.416015
.026	.042968	. 126	.167968	. 226	.292968	. 326	.417968
.027	.044921	. 127	.169921	.227	.294921	. 327	.419921
.030	.046875	. 130	. 171875	. 230	. 296875	. 330	.421875
.031	.048828	. 131	.173828	.231	.298828	.331	. 423828
.032	.050781	. 132	. 175781	. 232	.300781	. 332	. 425781
.033	.052734	. 133	. 177734	.233	.302734	.333	. 427734
.034	.054687	. 134	. 179687	.234	. 304687	.334	429687
.035	.056640	. 135	. 181640	.235	.306640	.335	.431640
.036	.058593			.236	.308593	.336	.433593
		. 136	. 183593				
.037	.060546	. 137	. 185546	. 237	.310546	.337	.435546
.040	.062500	. 140	.187500	. 240	.312500	.340	. 437500
.041	.064453	.141	. 189453	.241	.314453	.341	.439453
.042	.066406	. 142	.191406	. 242	.316406	.342	.441406
.043	.068359	. 143	. 193359	.243	.318359	. 343	.443359
.044	.070312	. 144	.195312	.244	.320312	. 344	.445312
.045	.072265	. 145	. 197265	. 245	.322265	. 345	.447265
.046	.074218	. 146	.199218	.246	.324218	.346	.449218
.047	.076171	. 147	.201171	. 247	.326171	.347	.451171
. 050	.078125	. 150	.203125	. 250	.328125	. 350	.453125
.051	.080078	. 151	. 205078	. 251	.330078	. 351	.455078
.052	.082031	. 152	. 207031	. 252	.332031	. 352	457031
.053	.083984	. 153	208984	. 253	.333984	.353	458984
.054	.085937	. 154	.210937	.254	.335937	. 354	460937
.055	.087890	. 155	212890	. 255	. 337890	.355	.462890
.056	.089843	. 156	.214843	.256	.339843	.356	.464843
.057	.091796	. 157	.216796	.257	.341796	.357	.466796
.060		1					
.061	.093750	. 160	.218750	.260	.343750	.360	.468750
	.095703	. 161	.220703	.261	.345703	.361	. 470703
.062	.097656	. 162	.222656	. 262	.347656	. 362	.472656
.063	.099609	. 163	. 224609	. 263	.349609	. 363	.474609
.064	.101562	. 164	.226562	. 264	.351562	. 364	. 476562
.065	.103515	. 165	.228515	. 265	.353515	. 365	.478515
.066	. 105468	. 166	.230468	. 266	.355468	. 366	.480468
.067	.107421	. 167	.232421	.267	.357421	. 367	.482421
.070	. 109375	. 170	.234375	.270	.359375	. 370	.484375
.071	.111328	. 171	.236328	.271	.361328	.371	.486328
.072	.113281	. 172	.238281	.272	.363281	. 372	488281
.073	. 115234	. 173	.240234	.273	.365234	.373	490234
.074	.117187	. 174	.242187	.274	.367187	.374	. 492187
.075	. 119140	. 175	.244140	.275	.369140	.375	.494140
.076	, 121093	. 176	.246093	.276	.371093	.376	.496093
.077	. 123046	. 177	.248046	.277	.373046	.377	.498046

NUMERICAL INFORMATION

OCTAL-DECIMAL FRACTION CONVERSION TABLE

		and the second					
OCTAL	DEC.	OCTAL	DEC.	OCTAL	DEC.	OCTAL	DEC.
.000000	.000000	.000100	.000244	.000200	.000488	.000300	.000732
.000001	.000003	.000101	.000247	.000201	.000492	.000301	.000736
000002	.000007	.000102	.000251	.000202	.000495	.000302	.000740
000003	.000011	.000103	.000255	.000203	.000499	.000303	.000743
000004	.000015	.000104	.000259	.000204	.000503	.000304	.000747
000005	.000019	.000105	.000263	.000205	.000507	.000305	.000751
000006	.000022	.000106	.000267	.000206	.000511	.000306	.000755
000007	.000026	.000107	.000270	.000207	.000514	.000307	.000759
000010	,000030	,000110	.000274	.000210	.000518	.000310	
000011	.000034	.000111	.000274	.000210	.000522	.000310	.000762
	• • • • • •				.000526		
000012	.000038	.000112	.000282	.000212		.000312	.000770
000013	.000041	.000113	.000286	.000213	.000530	.000313	.000774
000014	.000045	.000114	.000289	.000214	.000534	.000314	.000778
000015	.000049	.000115	.000293	.000215	.000537	.000315	.000782
000016	.000053	.000116	.000297	.000216	.000541	.000316	.000785
,000017	.000057	.000117	.000301	.000217	.000545	.000317	.000789
000020	.000061	.000120	.000305	.000220	.000549	.000320	.000793
000021	.000064	.000121	.000308	.000221	.000553	.000321	.000797
000022	.000068	.000122	.000312	.000222	.000556	.000322	.000801
000023	.000072	.000123	.000316	.000223	.000560	.000323	.000805
000024	.000076	.000124	.000320	.000224	.000564	.000324	.000808
000025	.000080	.000125	.000324	.000225	,000568	.000325	.000812
000026	.000083	.000126	.000328	.000226	.000572	.000326	.000816
000027	.000087	.000127	.000331	.000227	.000576	.000327	.000820
						1	
000030 000031	.000091	.000130	.000335 .000339	.000230	.000579	.000330	.000823
		.000131	• • • • • • •	.000231	• • • • • •		
000032	.000099	.000132	.000343	.000232	.000587	.000332	.000831
000033	.000102	.000133	.000347	.000233	.000591	.000333	.000835
000034	.000106	.000134	.000350	.000234	.000595	.000334	.000839
000035	.000110	.000135	.000354	.000235	.000598	.000335	.000843
000036	.000114	.000136	.000358	.000236	.000602	.000336	.000846
000037	.000118	.000137	.000362	.000237	.000606	.000337	.000850
000040	.000122	.000140	.000366	.000240	.000610	.000340	.000854
000041	.000125	.000141	.000370	.000241	.000614	.000341	.000858
000042	.000129	.000142	.000373	.000242	.000617	.000342	,000862
000043	.000133	.000143	.000377	.000243	.000621	.000343	.000865
000044	.000137	.000144	.000381	.000244	.000625	.000344	.000869
000045	.000141	.000145	.000385	.000245	.000629	.000345	.000873
000046	.000144	.000146	.000389	.000246	.000633	.000346	.000877
000047	.000148	.000147	000392	. 000247	.000637	.000347	.000381
000050	.000152	.000150	.000396	.000250	000640	.000350	.000885
000051	.000156	.000151	.000400	.000251	.000644	.000351	.000888
000052	.000160	.000152	.000404	.000252	.000648	.000352	.000892
000052	.000164	.000152	.000404	.000253	.000652	.000353	.000896
000053	.000167	.000154	.000408	.000253	.000656	.000354	.000900
000055				1 .	-	.000355	.000904
	.000171	.000155	.000415	.000255	.000659	.000356	.000904
000056	.000175	.000156	.000419	.000256	.000663		
000057	.000179	.000157	.000423	.000257	.000667	.000357	.000911
000060	.000183	.000160	.000427	.000260	,000671	.000360	.000915
000061	.000186	.000161	.000431	.000261	.000675	.000361	.000919
000062	.000190	.000162	.000434	.000262	.000679	.000362	.000923
000063	.000194	.000163	.000438	.000263	.000682	.000363	.000926
000064	.000198	.000164	.000442	.000264	.000686	.000364	.000930
0 00065	.000202	.000165	.000446	.000265	.000690	.000365	.000934
0 00066	.000205	.000166	.000450	.000266	.000694	.000366	.000938
000067	.000209	.000167	.000453	.000267	.000698	.000367	.000942
000070	.000213	.000170	.000457	.000270	.000701	.000370	.000946
000071	.000217	.000171	.000461	.000271	.000705	.000371	.000949
000072	.000221	.000172	.000465	.000272	.000709	.000372	.000953
000073	.000225	.000173	.000469	.000273	.000713	.000373	.000957
000074	.000228	.000174	.000473	.000274	.000717	.000374	.000961
000075	,000232	.000175	.000476	.000275	.000720	.000375	.000965
000076	.000236	.000176	.000480	.000276	.000724	.000376	.000968
	.000240	.000177	.000484	.000277	.000728	.000377	.000972
000077							

NUMERICAL INFORMATION

OCTAL-DECIMAL FRACTION CONVERSION TABLE

.000400 .000376 .000350 .001220 .000600 .001464 .000700 .001703 .000402 .000384 .005321 .001223 .000602 .001472 .007703 .001703 .004402 .000384 .005322 .000602 .001472 .007703 .001703 .004404 .000391 .00555 .001233 .000605 .001431 .007703 .001733 .004407 .001003 .00557 .001247 .001491 .007707 .001733 .004407 .001010 .00551 .00125 .001413 .007711 .001733 .004417 .001021 .000511 .00125 .001413 .007701 .001733 .004112 .001010 .005511 .001262 .000112 .001701 .001730 .004113 .001026 .000512 .001270 .000154 .007711 .00176 .004114 .001026 .000517 .000152 .001721 .001717 .00176 .001021	OCTAL	DEC.	OCTAL	DEC.	OCTAL	DEC.	OCTAL	DEC.
000402 000984 000521 00122 000463 000984 000502 001176 000702 001702 000604 000981 000505 001232 000605 001476 000702 001704 000604 000991 000505 001233 000605 001483 000706 001714 000605 0010991 000507 001247 000607 001491 000706 001710 000610 001007 0001251 000611 001191 000710 000173 000611 001014 000513 001255 000611 00113 001713 000713 000713 001713 001713 001714 0001724 001714 001714<	.000400							
000402 000988 000502 00122 0001672 000172 000173 001703 001703 001703 001703 001703 001703 001703 001703 001703 001703 001703 001703 001703 001703 001703 001703 001703 001703 001733 0001603 001733 0001603 001733 000173 001733 000173 001733 000173 001733 000173 001733 000173 001733 000173 001733 000173 000173 001733 000171 001733 000171 001733 000171 001733 001741 001733 001741 001733 001741 001733 001741 001733 001741 001733 001741 001733 001733 001714 001743 000171 00173 00173 000173 00173 00173 00173 000173 00173 00173 00173 00173 00173 00173 00173 00173 000171 00173 00173	.000401	.000980	.000501	.001224	.000601	.001468		
000604 000591 000503 001232 000604 0001476 0001764 000174 000604 000599 000505 001233 000605 001483 000706 001711 000604 001007 0001231 000605 001483 000706 001712 000610 001007 001233 000611 001491 000710 001733 000611 001012 000511 001252 000611 0001712 000711 001733 000712 000711 001733 001735 000714 000712 000714 000720 000714 000720 000714 000720 000714 000720 000714 000720 000714 000720 000714 000720 000720 000720 <td< td=""><td></td><td></td><td></td><td>.001228</td><td></td><td>.001472</td><td>.000702</td><td>.001716</td></td<>				.001228		.001472	.000702	.001716
000665 000899 0008057 001231 0008077 001231 0008077 001231 0008077 001231 0008077 001231 0008077 001231 0008077 001231 0008077 001231 0008077 001231 0008011 001193 000710 001733 001733 001733 001733 001733 001733 001733 001733 001733 001733 001733 001733 001733 001733 001733 001733 001733 001733 001735 <		.000988	.000503	.001232	.000603	.001476	.000703	.001720
000406 000999 .000505 .00143 .000705 .001705 .001705 .001705 .001705 .001705 .001705 .001705 .001705 .001705 .001705 .001705 .001705 .001705 .001705 .001705 .001711 .001705 .001711 .001705 .001711 .001705 .001711 .001713 .001711 .001713 .001711 .001735 .000611 .001185 .000711 .001737 .001747 .000173 .001722 .000515 .001714 .001755 .001711 .001752 .000711 .001722 .000151 .001711 .001722 .000171 .001722 .001711 .001722 .00171 .001722 .001711 .001723 .001711 .001723 .001711 .001723 .001711 .001723 .001711 .001723 .001723 .001723 .001723 .001723 .001723 .001723 .001723 .001723 .001723 .001723 .001723 .001723 .001723 .001723 .001723 .001723 .0	000404	.000991	.000504	.001235	.000604	.001480	.000704	.001724
000407 001007 001247 000007 001141 000107 000171 000411 001010 000511 001255 000611 001149 000111 00173 000412 001014 000513 001252 000613 001156 000113 001174 000413 001022 000515 001270 000515 001515 001514 000715 00176 000415 001229 000515 001274 000515 001522 000717 000717 000717 000717 000717 000715 00177 000715 00177 000717<		.000995	.000505	.001239				.001728
000411 001010 000510 001251 000611 001145 000710 00710 000411 001010 000512 001258 000612 001150 0007112 001713 000412 001138 000512 001266 000612 001150 0007114 001755 000413 001256 000615 001514 000714 001755 001755 000414 001025 000515 001271 000616 001512 000717 001755 000420 001037 000521 001282 000622 001529 000721 00177 000422 001041 000523 001283 000622 001529 000721 00177 000423 001052 000223 001283 000623 001544 000723 00172 000424 001052 000253 001300 000525 00157 000723 00172 000425 001050 000723 001300 000524 000723 000733 </th <th></th> <th></th> <th>.000506</th> <th>.001243</th> <th>.000606</th> <th></th> <th>.000706</th> <th>.001731</th>			.000506	.001243	.000606		.000706	.001731
000411 .001014 .000711 .001711 .00711 .007121 .007111 .007121 .007111 .007121 .007121 .007121	.000407	.001003	.000507	.001247	.000607	.001491	.000707	.001735
000411 .001014 .000711 .001711 .00711 .007121 .007111 .007121 .007111 .007121 .007121 .007121	.000410	.001007	.000510	.001251	.000610	.001495	.000710	.001739
000412 .001512 .001525 .00512 .001747 000413 .001032 .000514 .001525 .000714 .001754 000414 .001022 .000515 .001514 .000715 .001755 000415 .001028 .000515 .001517 .000715 .001752 .000417 .001037 .000520 .00177 .000617 .001522 .000717 .00176 .000421 .001041 .000521 .000717 .000720 .00171 .000720 .00171 .000422 .001041 .000521 .00128 .000722 .00171 .000423 .001052 .00153 .000723 .00173 .000723 .00173 .000424 .001056 .000525 .001304 .000622 .001546 .000725 .00178 .000425 .001056 .000537 .001304 .000627 .001556 .00173 .000426 .001646 .000537 .001334 .000531 .001747 .00174			.000511	.001255		.001499	.000711	.001743
000413 .001513 .001522 .005613 .001506 .000713 .001730 000414 .001622 .000515 .001270 .000515 .001755 .001755 .000415 .001023 .000515 .001277 .000515 .001755 .001755 .000417 .001033 .000517 .001277 .000511 .001522 .000717 .001765 .000422 .001041 .000522 .001285 .000621 .001529 .000712 .001771 .000422 .001044 .000522 .001283 .000622 .001337 .000723 .001781 .000423 .001052 .000524 .001285 .001052 .001742 .001782 .001782 .001782 .001782 .001782 .001726 .001726 .001726 .001727 .001786 .001730 .001804 .000623 .001564 .000727 .001786 .001730 .001804 .000633 .001575 .001730 .001804 .000633 .001564 .001730 .001810 <th></th> <th>.001014</th> <th>.000512</th> <th>.001258</th> <th>.000612</th> <th>.001502</th> <th>.000712</th> <th>.001747</th>		.001014	.000512	.001258	.000612	.001502	.000712	.001747
000415 .001515 .001270 .005151 .001755 .001755 000416 .001029 .000516 .001514 .000716 .001752 .000417 .001037 .000520 .001281 .000621 .001522 .000717 .001760 .000421 .001041 .000520 .001283 .000521 .001529 .000712 .001770 .000422 .001045 .000522 .001283 .000523 .001737 .000724 .00172 .001741 .000724 .001785 .000422 .001056 .000525 .001540 .000725 .001744 .000726 .001785 .000423 .001064 .000526 .001540 .000727 .001766 .001730 .001804 .000433 .001075 .000532 .001312 .000633 .001576 .000733 .001804 .000434 .001671 .000535 .001321 .000633 .001575 .001734 .001815 .000435 .001671 .000535 .0015		.001018	.000513	.001262	.000613	.001506	.000713	.001750
000415 .001515 .001270 .005151 .001755 .001755 000416 .001029 .000516 .001514 .000716 .001752 .000417 .001037 .000520 .001281 .000621 .001522 .000717 .001760 .000421 .001041 .000520 .001283 .000521 .001529 .000712 .001770 .000422 .001045 .000522 .001283 .000523 .001737 .000724 .00172 .001741 .000724 .001785 .000422 .001056 .000525 .001540 .000725 .001744 .000726 .001785 .000423 .001064 .000526 .001540 .000727 .001766 .001730 .001804 .000433 .001075 .000532 .001312 .000633 .001576 .000733 .001804 .000434 .001671 .000535 .001321 .000633 .001575 .001734 .001815 .000435 .001671 .000535 .0015	000414	.001022	.000514	.001266	.000614	.001510	.000714	.001754
.000417 .00133 .000517 .001217 .000520 .001211 .001525 .000720 .001766 .000420 .001041 .000520 .001281 .000620 .001525 .000721 .00173 .000422 .001045 .000522 .001283 .000622 .001353 .000723 .00173 .000424 .001052 .000525 .001300 .000622 .001544 .000725 .00173 .000425 .001066 .000526 .001306 .000725 .00175 .000426 .001066 .000527 .001364 .000726 .00175 .000430 .001068 .000530 .001312 .000630 .001556 .000730 .001800 .000431 .001071 .000532 .001319 .000631 .001561 .000734 .001800 .000433 .001071 .000535 .001321 .000634 .001571 .000734 .00181 .000434 .001087 .000535 .001371 .000734 .00181 <th></th> <th>.001026</th> <th></th> <th>.001270</th> <th>.000615</th> <th>.001514</th> <th>.000715</th> <th>.001758</th>		.001026		.001270	.000615	.001514	.000715	.001758
.000420 .001037 .000520 .001281 .000620 .001525 .000720 .00170 .000421 .001041 .000521 .001285 .000622 .001533 .000721 .00171 .000423 .001052 .001052 .000524 .001533 .000724 .00172 .000424 .001056 .000525 .001541 .000724 .00174 .000425 .001056 .000526 .001304 .000622 .001544 .000725 .00172 .000427 .001064 .000527 .001308 .000627 .001556 .000727 .001860 .000430 .001075 .000531 .001312 .000633 .001560 .000731 .001861 .000432 .001075 .000535 .00131 .000633 .001575 .000732 .00181 .000435 .001871 .000536 .00132 .000633 .001575 .000735 .00181 .000435 .00181 .000536 .001328 .000643 .001575	.000416	.001029	.000516	.001274	.000616	001518	.000716	.001762
.000421 .001241 .001251 .001285 .000622 .001533 .000721 .00172 .000422 .001045 .000523 .001293 .000623 .001537 .000723 .001741 .000423 .001045 .000524 .001541 .000724 .001741 .000425 .001056 .000525 .001304 .000725 .00175 .000426 .001060 .000527 .001552 .000726 .00178 .000427 .001064 .000527 .001552 .000727 .00176 .000431 .001075 .000532 .001186 .000721 .00180 .000432 .001075 .000532 .001180 .000731 .00180 .000433 .001075 .000532 .001312 .000633 .001571 .000733 .00180 .000435 .001087 .000535 .001335 .000633 .001571 .000734 .00183 .000437 .001084 .001355 .000174 .00183 .000747	.000417	.001033	.000517	.001277	.000617	.001522	.000717	.001766
.000421 .001241 .001251 .001285 .000622 .001533 .000721 .00172 .000422 .001045 .000523 .001293 .000623 .001537 .000723 .001741 .000423 .001045 .000524 .001541 .000724 .001741 .000425 .001056 .000525 .001304 .000725 .00175 .000426 .001060 .000527 .001552 .000726 .00178 .000427 .001064 .000527 .001552 .000727 .00176 .000431 .001075 .000532 .001186 .000721 .00180 .000432 .001075 .000532 .001180 .000731 .00180 .000433 .001075 .000532 .001312 .000633 .001571 .000733 .00180 .000435 .001087 .000535 .001335 .000633 .001571 .000734 .00183 .000437 .001084 .001355 .000174 .00183 .000747	. 000420	001037	.000520	.001281	. 000620	.001525	.000720	.001770
.000422 .001045 .000522 .001233 .001521 .001722 .001721 .000423 .001052 .001233 .000523 .001531 .000723 .001785 .000424 .001056 .000524 .001296 .000524 .001541 .000724 .001785 .000425 .001066 .005526 .001304 .000627 .001566 .000727 .001785 .000421 .001064 .000531 .001312 .000631 .001566 .000731 .001800 .000431 .001075 .000532 .001323 .000632 .001564 .000731 .001800 .000433 .001075 .000536 .001331 .000633 .001731 .000734 .001811 .000434 .001083 .001327 .000634 .00175 .000736 .001812 .000440 .001098 .000536 .001331 .000636 .001574 .001813 .000440 .001098 .000541 .001356 .0001741 .001834								
.000423 .001049 .000523 .001233 .001531 .000723 .001783 .000424 .001056 .000525 .001030 .000625 .001544 .000724 .001789 .000425 .001060 .000525 .001030 .000625 .001544 .000725 .00179 .000426 .001071 .000530 .001552 .000727 .001808 .000431 .001071 .000532 .001316 .000633 .001556 .000730 .001808 .000433 .001075 .000532 .001312 .000633 .001556 .000730 .001808 .000433 .001075 .000535 .001331 .000633 .001757 .000735 .00173 .001808 .000437 .001094 .000535 .001335 .000637 .001533 .000737 .001831 .000441 .001044 .000541 .001346 .000641 .001586 .000741 .001831 .000441 .001044 .000555 .001354 .000							.000722	
000424 001052 .000524 .001296 .000625 .001306 .000724 .001724 .001724 .001724 .001724 .001724 .001724 .001724 .001724 .001725 .001304 .000625 .001304 .000725 .001736 .000726 .001736 .000726 .001736 .000726 .001736 .000726 .001736 .000726 .001736 .000726 .001736 .000731 .001806 .000532 .0013119 .000633 .001576 .000733 .001806 .000733 .001815 .000733 .001815 .000735 .001835 .001835 .001835 .001835 .001835 .000733 .001815 .000735 .001815 .000735 .001835 .000735 .001835 .000735 .001835 .000735 .001835 .000735 .001835 .000735 .001835 .000737 .001835 .000737 .001835 .000741 .001835 .000741 .001835 .000741 .001835 .000741 .001835 .000741 .001835 <td< td=""><td>000423</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></td<>	000423							
000425 .001366 .000525 .001300 .000625 .001544 .000725 .001755 000426 .001064 .000527 .001308 .000627 .001555 .000727 .001796 .000430 .001064 .000531 .001312 .000630 .001556 .000727 .001796 .000431 .001075 .000531 .001316 .000632 .001564 .000732 .001808 .000432 .001075 .000531 .001321 .000634 .001571 .000733 .001815 .000434 .001087 .000535 .001335 .000637 .001575 .000733 .001815 .000435 .001871 .000734 .001827 .000737 .001827 .000441 .001986 .000537 .001335 .000637 .001583 .000737 .001827 .000441 .001106 .000542 .001346 .000641 .001590 .000741 .001827 .000441 .001106 .000542 .001358 .000	.000424							
.000426 .001060 .000526 .001304 .000626 .001548 .000726 .001792 .000427 .001064 .000537 .001308 .000627 .001552 .000727 .001796 .000430 .001071 .000531 .001312 .000633 .001560 .000733 .001804 .000433 .001079 .000533 .001323 .000633 .001571 .000733 .001811 .000434 .001083 .000535 .001331 .000636 .001579 .000736 .01823 .000436 .001094 .000537 .001338 .000537 .001833 .000747 .001833 .000437 .001094 .000547 .001338 .000637 .001833 .000737 .001833 .000441 .001098 .000541 .001342 .000644 .001894 .000741 .001842 .000444 .00110 .000542 .001358 .000741 .001842 .000444 .00112 .000544 .001358 .0007	000425							
.000427 .001064 .000527 .001308 .000627 .001552 .000727 .001795 .000430 .001071 .000531 .001312 .000630 .001556 .000730 .001800 .000431 .001075 .000531 .00132 .000632 .001564 .000732 .001808 .000433 .001075 .000534 .00132 .000633 .001577 .000733 .001815 .000435 .001087 .000536 .001375 .000737 .001815 .000437 .001094 .000537 .001338 .000637 .001583 .000737 .001831 .000440 .001094 .000541 .001342 .000640 .001586 .000740 .001831 .000441 .001166 .00541 .001345 .000642 .001594 .000743 .001831 .000442 .001166 .00544 .001356 .000744 .001831 .000441 .001125 .000541 .001356 .000744 .001831							.000726	
.000430 .001688 .000530 .001312 .000630 .001556 .000730 .001800 .000431 .001071 .000531 .001560 .00731 .001808 .000433 .001079 .000532 .001319 .000633 .001571 .000734 .001808 .000434 .001687 .000535 .001321 .000633 .001571 .000734 .001815 .000435 .00191 .000535 .001331 .000635 .001579 .001235 .001827 .000436 .001988 .000541 .001342 .000640 .001586 .001741 .001831 .000440 .001988 .000541 .001345 .000641 .001586 .000741 .001831 .000441 .001100 .000541 .001358 .000744 .001831 .000744 .001831 .000444 .001110 .000544 .001358 .000744 .001845 .001846 .001159 .000744 .001846 .000446 .001121 .000								
.000431 .001071 .000531 .001316 .000631 .001560 .000731 .001804 .000432 .001075 .000533 .001319 .000632 .001564 .00733 .001811 .000433 .001683 .000534 .00127 .000634 .001571 .000734 .001811 .000435 .00187 .000535 .001331 .000635 .001575 .000735 .001823 .000436 .001994 .000537 .001338 .000637 .001583 .000737 .001831 .000440 .00198 .000541 .001342 .000641 .001580 .000740 .001834 .000441 .001106 .000542 .001544 .001842 .001844 .001842 .001844 .001842 .001844 .001842 .001844 .001842 .001844 .001842 .001844 .001842 .001844 .001844 .001842 .001844 .001842 .000744 .001842 .000744 .001842 .000744 .001843 .000745					000630	001556	000730	
.000432 .001075 .000532 .001319 .000632 .001564 .000732 .001808 .000433 .001079 .000534 .001327 .000633 .001571 .000734 .001815 .000435 .001867 .000534 .001327 .000635 .001575 .000736 .001815 .000436 .001991 .000536 .001335 .000636 .001579 .000737 .001827 .000437 .001988 .000541 .001342 .000640 .001586 .000741 .001837 .000441 .00102 .000541 .001345 .000641 .001586 .000742 .01838 .000442 .00110 .000541 .001350 .000641 .001588 .000744 .001838 .000444 .001110 .000541 .001358 .000644 .001159 .000744 .001846 .000446 .001121 .000546 .001363 .000645 .001603 .000745 .001857 .000455 .001122 .0005								
000433 001079 000533 001323 000633 001567 000733 00181 000434 001083 000534 001327 000634 001575 000735 00175 000435 001091 000536 001331 000635 001575 000736 001827 000436 001091 000537 001332 000637 001585 000737 001827 000441 001094 000547 001342 000640 001586 000740 00181 000442 001106 000542 001350 000641 001590 000742 001842 000442 001106 000543 001354 000643 001598 000742 001842 000444 001113 000545 001358 000644 0011605 000744 001842 000446 001125 000547 001365 001645 001605 000746 001837 000443 001125 000551 001373 000651 001617						-		
000434 001083 .000534 .001571 .000534 .001571 000435 .001087 .000535 .001331 .000635 .001575 .000735 .001819 000436 .00194 .000537 .001333 .000636 .001575 .000737 .001823 000440 .00198 .000540 .001342 .000641 .001583 .000737 .001831 .000441 .001106 .000542 .001346 .000641 .001594 .000741 .001842 .000443 .001110 .000543 .001358 .000643 .001594 .000743 .001842 .000444 .001113 .000544 .001358 .000644 .001602 .000744 .001842 .000446 .001121 .000545 .001365 .000647 .001613 .000745 .001857 .000450 .001129 .000551 .001377 .000651 .001621 .000750 .001857 .000451 .001129 .000551 .001617 .000755 </td <td>000433</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td>	000433							
.000435 .001087 .000535 .001331 .000635 .001575 .000735 .00131 .000436 .001091 .000537 .001333 .000637 .001583 .000736 .00123 .000441 .00198 .000541 .001342 .000640 .001586 .000740 .001831 .000441 .001102 .000541 .001346 .000641 .001590 .000741 .001834 .000442 .00110 .000543 .001354 .000643 .001598 .000743 .001842 .000443 .001113 .000543 .001358 .000644 .001605 .000745 .001842 .000444 .001122 .000544 .001365 .000645 .001605 .000746 .001842 .000445 .001122 .000546 .001365 .000647 .001613 .000747 .001853 .000450 .001122 .000550 .001373 .000650 .001617 .000750 .001853 .000451 .001122 .00055								
000436 001091 000536 001335 000637 001579 000736 001233 000437 001094 000537 001338 000637 001583 000737 001227 000440 001988 000541 001342 000640 001586 000741 001834 000442 001106 000542 001356 000643 001594 000742 001838 000443 001113 000544 001358 000643 001598 000744 001838 000444 001113 000544 001358 000644 001602 000744 001846 000445 001122 000544 001365 000644 001602 000744 001846 000446 001122 000547 001369 000647 001613 000747 001857 000451 001122 000551 001377 000651 001621 000750 0018181 000452 001135 000551 001377 000651 001								
.000437 .001094 .000537 .001338 .000637 .001583 .000737 .001827 .000440 .001098 .000540 .001342 .000640 .001586 .000740 .001831 .000441 .001102 .000541 .001356 .000641 .001590 .000741 .001838 .000443 .001110 .000542 .001358 .000643 .001598 .000743 .001842 .000444 .001117 .000545 .001655 .000744 .001850 .000745 .001850 .000446 .001121 .000546 .001365 .000646 .001605 .000746 .001853 .000447 .001322 .000551 .001377 .000651 .001617 .001857 .001857 .000451 .00132 .000551 .001373 .000651 .001621 .000750 .001857 .000452 .001353 .001651 .001625 .00175 .001865 .000451 .001328 .0000551 .001625 .00								
000440 00198 000540 00132 000640 001586 000740 001831 000441 001102 000541 001346 000641 001590 000741 001831 000442 001106 000542 001350 000643 001594 000742 001838 000443 001113 000544 001358 000644 001502 000744 001842 000444 001113 000545 001365 000644 001605 000745 001845 000446 001121 000545 001365 000647 001613 000747 001857 000450 001129 000550 001373 000651 001617 000750 001861 000453 001140 000553 001384 000653 001621 000752 001863 000453 001140 000554 001388 000653 001622 000754 001874 000455 001148 000556 001388 000655 001636								.001827
.00041 .001302 .000541 .001346 .001641 .001590 .000741 .001834 .000442 .001106 .000542 .001350 .000642 .001594 .000741 .001834 .000443 .001110 .000544 .001358 .000643 .001594 .000743 .001842 .000444 .001117 .000545 .001361 .000645 .001605 .000745 .001853 .000446 .001121 .000546 .001365 .000645 .001605 .000747 .001853 .000450 .001129 .000550 .001373 .000650 .001613 .000750 .001861 .000451 .001132 .000551 .001377 .000653 .001622 .000751 .001863 .000452 .001136 .000553 .001384 .000653 .001622 .000752 .001873 .000454 .001144 .000555 .001382 .000653 .001632 .000754 .001876 .000455 .001159 .00				-				
.000442 .001106 .000542 .001350 .000642 .001594 .000742 .001842 .000443 .001110 .000543 .001354 .000643 .001598 .000743 .001842 .000445 .001117 .000545 .001361 .000644 .001155 .000745 .001845 .000446 .001121 .000546 .001365 .000646 .001605 .000745 .001850 .000450 .001129 .000551 .001373 .000651 .001621 .000750 .001865 .000452 .001136 .000552 .001825 .000751 .001865 .000453 .001140 .000553 .001386 .000652 .001625 .000754 .001873 .000455 .001144 .000555 .001388 .000655 .001836 .000755 .001886 .000455 .001152 .000556 .001386 .000655 .001836 .000755 .001884 .000456 .001150 .000556 .001856 .0	000440	001102						-
.000443 .001110 .000543 .001354 .000643 .001598 .000743 .001842 .000444 .001113 .000544 .001358 .000644 .001602 .000744 .001842 .000445 .001117 .000545 .001361 .000644 .001605 .000745 .001853 .000447 .001125 .000547 .001369 .000647 .001613 .000747 .001853 .000450 .001129 .000550 .001373 .000650 .001613 .000751 .001861 .000452 .001146 .000552 .001380 .000653 .001828 .000752 .001869 .000454 .001144 .000555 .001392 .000653 .001828 .000755 .001880 .000455 .001152 .000557 .001896 .001632 .000756 .001880 .000456 .001152 .000557 .001399 .000657 .001844 .000756 .001892 .000461 .001153 .000561 .0								
.000444 .001113 .000544 .001358 .000644 .001602 .000744 .001861 .000445 .001117 .000545 .001361 .000645 .001605 .000745 .001853 .000447 .001125 .000550 .001373 .000650 .001613 .000746 .001853 .000450 .001129 .000551 .001373 .000651 .001621 .000751 .001865 .000451 .001140 .000553 .001384 .000653 .001622 .001752 .001869 .000453 .001140 .000554 .001388 .000653 .001628 .000755 .001869 .000454 .001144 .000555 .001392 .000655 .001836 .000755 .001880 .000455 .001148 .000557 .001399 .000657 .001846 .000756 .001892 .000456 .001159 .000566 .001407 .000657 .001844 .000757 .001892 .000461 .001153 .0							.000743	
.000445 .001117 .000545 .001361 .000645 .001605 .000745 .001850 .000446 .001121 .000546 .001365 .000646 .001609 .000745 .001850 .000450 .001129 .000550 .001373 .000651 .001611 .000745 .001861 .000452 .001365 .001377 .000651 .001621 .000751 .001865 .000452 .001366 .000552 .001380 .00652 .001625 .000751 .001865 .000453 .00140 .000553 .001384 .000652 .001625 .000753 .001873 .000455 .001144 .000555 .001386 .000655 .001836 .000755 .001884 .000455 .001152 .000556 .001392 .000655 .001836 .000755 .001884 .000456 .001153 .000557 .001396 .000655 .001844 .000757 .001884 .000461 .001153 .000561 .001							.000744	.001846
.000446 .001121 .000546 .001365 .000646 .001609 .000746 .001853 .000447 .001125 .000550 .001369 .000647 .001813 .000747 .001851 .000450 .001129 .000550 .001373 .000651 .001617 .000750 .001861 .000452 .001136 .000552 .001380 .000653 .001625 .000751 .001873 .000453 .001144 .000554 .001384 .000653 .001628 .000755 .001873 .000455 .001144 .000555 .001392 .000655 .001838 .000755 .001880 .000456 .001152 .000557 .001399 .000655 .001644 .000757 .001880 .000457 .001153 .000560 .001407 .001651 .000761 .001892 .000460 .001171 .000562 .001411 .000662 .001647 .000763 .001892 .000461 .001163 .000561 .0								.001850
.000447 .001125 .000547 .001369 .000647 .001613 .000747 .001857 .000450 .001129 .000550 .001373 .000650 .001817 .000750 .001851 .000451 .001132 .000551 .001377 .000651 .001621 .000751 .001865 .000453 .001140 .000552 .001380 .000653 .001628 .000752 .001869 .000454 .001144 .000554 .001388 .000653 .001628 .000754 .001876 .000455 .001148 .000556 .001392 .000655 .001836 .000755 .001880 .000456 .001152 .000557 .001399 .000657 .001844 .000756 .001893 .000461 .001153 .000562 .001407 .000667 .001844 .000761 .001893 .000462 .001167 .000562 .001411 .000662 .001655 .000762 .001893 .000463 .001174 .0						.001609	.000746	.001853
.000451 .00132 .000551 .001377 .000651 .001621 .000751 .001865 .000452 .001136 .000552 .001380 .000652 .001625 .000753 .001865 .000453 .001140 .000553 .001384 .000653 .001625 .000753 .001873 .000454 .001144 .000555 .001382 .000655 .001832 .000754 .001876 .000455 .001152 .000556 .001392 .000655 .001836 .000755 .001888 .000450 .001155 .000557 .001399 .000657 .001844 .000757 .001888 .000461 .001153 .000561 .001403 .000661 .001647 .000760 .001892 .000461 .001171 .000562 .001411 .000662 .001655 .000763 .001893 .000462 .001174 .000564 .001415 .000663 .001657 .001893 .000464 .001174 .000565 .00				.001369	.000647	.001613	.000747	.001857
.000451 .00132 .000551 .001377 .000651 .001621 .000751 .001865 .000452 .001136 .000552 .001380 .000652 .001625 .000753 .001865 .000453 .001140 .000553 .001384 .000653 .001625 .000753 .001873 .000454 .001144 .000555 .001382 .000655 .001832 .000754 .001876 .000455 .001152 .000556 .001392 .000655 .001836 .000755 .001888 .000450 .001155 .000557 .001399 .000657 .001844 .000757 .001888 .000461 .001153 .000561 .001403 .000661 .001647 .000760 .001892 .000461 .001171 .000562 .001411 .000662 .001655 .000763 .001893 .000462 .001174 .000564 .001415 .000663 .001657 .001893 .000464 .001174 .000565 .00	.000450	001129	000550	.001373	.000650	.001617	.000750	.001861
.000452 .001136 .000552 .001380 .000652 .001625 .000752 .001869 .000453 .001140 .000553 .001384 .000653 .001628 .000754 .001871 .000454 .001144 .000555 .001392 .000653 .001632 .000754 .001880 .000455 .001152 .000556 .001392 .000655 .001644 .000755 .001880 .000456 .001152 .000557 .001399 .000657 .001644 .000756 .001880 .000461 .001153 .000561 .001403 .000661 .001651 .000760 .001892 .000462 .001171 .000562 .001411 .000662 .001655 .000762 .001893 .000464 .001174 .000565 .001422 .000665 .001663 .000764 .001907 .000465 .001178 .000567 .001422 .000665 .001674 .000765 .001910 .000466 .001182 .0								.001865
.000453 .001140 .000553 .001384 .000653 .001628 .000753 .001873 .000454 .001144 .000554 .001388 .000654 .001632 .000755 .00180 .000455 .001152 .000556 .001399 .000655 .001636 .000755 .001884 .000457 .001155 .000577 .001399 .000657 .001644 .000756 .001884 .000460 .001153 .000561 .001647 .000761 .001895 .000462 .001167 .000562 .001411 .000662 .001655 .000761 .001895 .000463 .001171 .000563 .001415 .000663 .000763 .001903 .000464 .001174 .000565 .001415 .000663 .000764 .001903 .000465 .001178 .000565 .001422 .000665 .001677 .001910 .000466 .001178 .000567 .001434 .000667 .001674 .000767 .00						,001625	.000752	.001869
.000455 .001148 .000555 .001392 .000655 .001636 .000755 .001880 .000456 .001152 .000556 .001396 .000655 .001640 .000755 .001880 .000457 .001155 .000557 .001399 .000657 .001644 .000757 .001888 .000461 .001153 .000561 .001403 .000661 .001651 .000760 .001892 .000462 .001117 .000562 .001411 .000662 .001655 .000764 .001939 .000464 .001171 .000562 .001411 .000663 .000763 .001839 .000464 .001174 .000565 .001419 .000663 .000764 .001907 .000465 .001178 .000566 .001422 .000665 .001677 .000765 .001917 .000465 .001182 .000567 .001424 .000665 .001674 .000766 .001911 .000466 .001182 .000567 .001434 .0	.000453	.001140	.000553	.001384		.001628		.001873
.000455 .001148 .000555 .001392 .000655 .001636 .000755 .001830 .000456 .001152 .000556 .001396 .000655 .001640 .000755 .001838 .000457 .001155 .000557 .001399 .000657 .001644 .000757 .001888 .000461 .001153 .000561 .001403 .000661 .001651 .000760 .001892 .000462 .001167 .000562 .001411 .000662 .001655 .000761 .001892 .000464 .001171 .000562 .001415 .000663 .001655 .000762 .001893 .000464 .001174 .000565 .001422 .000665 .001657 .000764 .001907 .000465 .001182 .000566 .001426 .001667 .000765 .001914 .000465 .001182 .000567 .001424 .000665 .001674 .000766 .001914 .000466 .001182 .000577 .0		.001144	.000554		.000654			
.000457 .00155 .000577 .001399 .000657 .001644 .000757 .001888 .000460 .001159 .000560 .001403 .000660 .001647 .000760 .001892 .000461 .001163 .000561 .001407 .000661 .001651 .000760 .001892 .000463 .001171 .000563 .001413 .000663 .001659 .000762 .001893 .000464 .001174 .000564 .001419 .000664 .001663 .000765 .001903 .000465 .001178 .000565 .001422 .000665 .001667 .000765 .001911 .000465 .001182 .000567 .001426 .001670 .000765 .001911 .000467 .001186 .000577 .001430 .000665 .001674 .000767 .001918 .000470 .001190 .000571 .001434 .000671 .001672 .000770 .001926 .000471 .001194 .000571 .00				.001392				
.000460 .001159 .000560 .001403 .000660 .001647 .000760 .001892 .000461 .001163 .000561 .001407 .000661 .001651 .000762 .001892 .000462 .001167 .000562 .001411 .000662 .001655 .000762 .001893 .000464 .001171 .000564 .001115 .000663 .001653 .000764 .001903 .000465 .001174 .000565 .001422 .000665 .001663 .000764 .001917 .000466 .001182 .000565 .001422 .000665 .001670 .000765 .001911 .000467 .001186 .001567 .001674 .000767 .001918 .000470 .001180 .000571 .001434 .000671 .001678 .000770 .001926 .000471 .00197 .000573 .001445 .000673 .001689 .000771 .001930 .000473 .001201 .000573 .001445 .00	.000456	.001152						
.000461 .00163 .000561 .001407 .000661 .001651 .000761 .001895 .000462 .001167 .000562 .001411 .000662 .001655 .000762 .001895 .000464 .001171 .000563 .001415 .000663 .001655 .000763 .001903 .000464 .001174 .000565 .001419 .000665 .001663 .000764 .001907 .000465 .001178 .000565 .001422 .000665 .001667 .000765 .001910 .000466 .00185 .001422 .000665 .001670 .000765 .001911 .000477 .001185 .000570 .001430 .000670 .001674 .000770 .00122 .000471 .001190 .000570 .001434 .000670 .001678 .000770 .00122 .000471 .001197 .000573 .001438 .000671 .001682 .000771 .00132 .000473 .001205 .000573 .001445		-		.001399				-
.000462 .001167 .000562 .001411 .000662 .001655 .000762 .001899 .000463 .001171 .000563 .001415 .000663 .001659 .000763 .001903 .000464 .001174 .000564 .001419 .000663 .001657 .000764 .001907 .000465 .001178 .000565 .001422 .000665 .001667 .000765 .001911 .000466 .001182 .000566 .001426 .000665 .001670 .000765 .001911 .000467 .001186 .000577 .001430 .000667 .001674 .000776 .001922 .000470 .001190 .000571 .001438 .000671 .001872 .000771 .001922 .000471 .001197 .000572 .00141 .000672 .001686 .000772 .001930 .000473 .001201 .000573 .001445 .000673 .001689 .000773 .001931 .000474 .001205 .00	.000460	.001159					.000760	
.000463 .001171 .000563 .001415 .000663 .001659 .000763 .001903 .000464 .001174 .000564 .001419 .000664 .001663 .000765 .001903 .000465 .001178 .000565 .001419 .000665 .001663 .000765 .001903 .000465 .001178 .000565 .001422 .000665 .001667 .000765 .001911 .000467 .001186 .000567 .001426 .000667 .001674 .000767 .001912 .000470 .001190 .000570 .01434 .000670 .001678 .000770 .001926 .000471 .001914 .000571 .001438 .000672 .001678 .000770 .001926 .000473 .00197 .000573 .001438 .000672 .001686 .000772 .00130 .000473 .001205 .000573 .001445 .000673 .001689 .000773 .001334 .000474 .001205 .0005								
.000464 .001174 .000564 .001419 .000664 .001663 .000764 .001907 .000465 .001178 .000555 .001422 .000665 .001667 .000765 .001911 .000465 .001178 .000555 .001422 .000665 .001677 .000765 .001911 .000466 .001182 .000567 .001428 .000665 .001670 .000765 .001911 .000471 .001190 .000571 .001434 .000670 .001678 .000770 .001922 .000471 .001194 .000571 .001438 .000671 .001682 .000770 .001922 .000473 .001201 .000573 .001441 .000673 .001689 .000773 .001934 .000474 .001205 .000574 .001449 .000674 .001693 .000774 .001934 .000475 .001209 .000575 .001453 .000675 .001693 .000775 .001941 .000475 .001209 .0								
.000465 .001178 .000565 .001422 .000665 .001667 .000765 .001911 .000466 .001182 .000566 .001426 .000666 .001670 .000766 .001911 .000467 .001186 .000567 .001430 .000667 .001674 .000766 .001918 .000470 .001190 .000570 .001434 .000670 .001678 .000770 .001922 .000471 .00197 .000572 .001414 .000671 .001682 .000771 .001920 .000473 .001201 .000573 .001441 .000673 .001689 .000773 .001930 .000474 .001205 .000574 .001449 .000674 .001693 .000774 .001937 .000476 .001205 .000575 .001453 .000675 .001693 .000774 .001937 .000476 .001205 .000576 .001453 .000675 .001697 .000775 .001941 .000476 .001205 .00								
.000466 .00182 .000566 .001426 .000666 .001670 .000766 .001914 .000467 .001186 .000567 .001430 .000667 .001674 .000767 .001912 .000470 .001190 .000570 .001434 .000670 .001678 .000770 .001922 .000471 .001194 .000571 .001438 .000671 .001682 .000771 .001922 .000473 .001201 .000573 .001441 .000673 .001689 .000773 .001930 .000474 .001205 .000574 .001645 .000673 .001689 .000773 .001930 .000474 .001205 .000574 .001645 .000673 .001693 .000774 .001937 .000475 .001209 .000575 .001453 .000675 .001697 .000775 .001941 .000476 .001203 .000575 .001453 .000675 .001597 .000775 .001941 .000475 .001203 .00								
.000467 .001186 .000567 .001630 .000667 .001674 .000767 .001918 .000470 .001190 .000570 .001434 .000670 .001678 .000770 .001920 .000471 .001194 .000571 .001438 .000671 .001678 .000771 .001926 .000471 .001194 .000571 .001438 .000672 .001682 .000771 .001926 .000473 .001201 .000573 .001441 .000673 .001689 .000773 .001934 .000474 .001205 .000574 .001449 .000674 .001693 .000774 .001934 .000475 .001209 .000575 .001453 .000675 .001697 .000774 .001934 .000476 .001203 .000575 .001453 .000675 .001697 .000775 .001941 .000476 .001203 .000575 .001457 .000676 .001071 .000776 .001941								
.000470 .001190 .000570 .001434 .000670 .001678 .000770 .001922 .000471 .001194 .000571 .001438 .000671 .001682 .000771 .001922 .000471 .001194 .000571 .001438 .000671 .001682 .000771 .001922 .000473 .001201 .000572 .001441 .000673 .001686 .000773 .00130 .000474 .001205 .000574 .001445 .000674 .001693 .000774 .001334 .000475 .001209 .000575 .001433 .000675 .001693 .000774 .00134 .000476 .001203 .000575 .001453 .000675 .001693 .000775 .001941 .000476 .001203 .000575 .001457 .000676 .00175 .000775 .001941								
.000471 .001194 .000571 .001438 .000671 .001682 .000771 .001926 .000472 .001197 .000572 .001441 .000672 .001866 .000772 .001330 .000473 .001201 .000573 .001445 .000673 .001689 .000773 .001930 .000474 .001205 .000574 .001449 .000674 .001693 .000774 .001937 .000475 .001209 .000574 .001453 .000675 .001697 .000775 .001941 .000476 .001213 .000576 .001457 .001675 .001707 .001941		.001186						
.000472 .001197 .000572 .001411 .000672 .001686 .000772 .001930 .000473 .001201 .000573 .001445 .000673 .001689 .000773 .001930 .000474 .001205 .000574 .001445 .000673 .001689 .000773 .001930 .000474 .001205 .000574 .001449 .000674 .001693 .000774 .001937 .000475 .001209 .000575 .001453 .000675 .001941 .000775 .001941 .000476 .001213 .000576 .001457 .0000776 .001945								
.000473 .001201 .000573 .001445 .000673 .001689 .000773 .001934 .000474 .001205 .000574 .001449 .000674 .001693 .000774 .001934 .000475 .001209 .000575 .001433 .000675 .001693 .000775 .001934 .000476 .001209 .000575 .001453 .000675 .001697 .000775 .001941 .000476 .001213 .000576 .001457 .000676 .001201 .000776 .001945								
.000474 .001205 .000574 .001449 .000674 .001693 .000774 .001937 .000475 .001209 .000575 .001453 .000675 .001697 .000775 .001941 .000476 .001213 .000576 .001457 .000676 .001701 .000776 .001945								
.000475 .001209 .000575 .001453 .000675 .001697 .000775 .001941 .000476 .001213 .000576 .001457 .000676 .001701 .000776 .001945								
.000476 .001213 .000576 .001457 .000676 .001701 .000776 .001945								
.000477 .001216 .000577 .001461 .000677 .001705 .000777 .001949								
	.000477	.001216	.000577	.001461	.000677	.001705	.000777	.001949
					1		1	
	·····		L	·····			1	

NUMERICAL INFORMATION

MATHEMATICAL INFORMATION

POWERS	OF	TEN	$(10^{\pm n})$ I	N OCTAL	r

10 ⁿ	n		10 ⁻ⁿ
	1 0	1. 000 000 0	00 000 000 000 00
	12 1	0. 063 146 3	314 631 463 146 31
1.	14 2	0. 005 075 3	341 217 270 243 66
1 7	50 3	0. 000 406 1	11 564 570 651 77
23 4	20 4	0. 000 032 1	.55 613 530 70 4 15
303 2	40 5	0. 000 002 4	76 132 610 706 64
3 641 1	00 6	0. 000 000 2	206 157 364 055 37
46 113 2)0 7	0. 000 000 0	015 327 745 152 75
575 360 4	00 8	0. 000 000 0	01 257 143 561 06
7 346 545 0)0 9	0. 000 000 0	000 104 560 276 41
112 402 762 0	00 10	0. 000 000 0	00 006 676 337 66
1 351 035 564 0	0 11	0. 000 000 0	00 000 537 657 77
16 432 451 210 0	12	0. 000 000 0	000 000 043 136 32
221 411 634 520 0	0 13	0. 000 000 0	000 000 003 411 35
2 657 142 036 440 0	14	0. 000 000 0	000 000 000 264 11
34 327 724 461 500 0	0 15	0. 000 000 0	000 000 000 022 01
434 157 115 760 200 0	16	0. 000 000 0	000 000 000 001 63
5 432 127 413 542 400 0	00 17	0. 000 000 0	000 000 000 000 014
67 405 553 164 731 000 0	00 18	0. 000 000 0	000 000 000 000 001

VARIOUS CONSTANTS IN OCTAL NOTATION

π = 3.11037552421	e = 2.55760521305
$\pi/2$ = 1.04417665210	1/e = 0.27426530661
$1/\pi = 0.24276301556$	√e = 1.51411230704
$\sqrt{\pi}$ = 1.61337611067	$\log_{10} e = 0.33626754251$
$\ln \pi$ = 1.11206404435	$\sqrt{2}$ = 1.32404746320
$\sqrt{10}$ = 3.12305407267	ln 10 = 2.23273067355

APPENDIX G

SEL 810B INSTRUCTION LIST SUMMARY

CLASS	MNEMONIC	OP CODE	CYCLES 750 NANOSECONDS	FUNCTION	PAGE
ARITHMETIC:	АМА	05	2	Add Memory to A	2-6
	AMB SMA	16 06	2 2	Add Memory to B Subtract Memory from A	2-6 2-6
	MPY	07	2 6 Sec.	Multiply B times Memory	2-0
	DIV	10	11 Sec.	Divide A and B by Memory	2-7
	RNA'	00-01	1	Round A by MSB in B	2 - 8
	OVS	00-37	1	Set Overflow	2-8
LOAD/STORE:	LAA	01	2	Load A from Memory	2-8
	LBA STA	02 03	2 2	Load B from Memory Store Memory from A	2-8 2-9
	STB	03	2	Store Memory from B	2-9
	LCS	00-31	1	Load Control Switches in A	2-9
	LIX	00-45	2	Load Hardware Index Register	2-9
	<u>STX</u>	00-44	2	Store Hardware Index Register	2-9
	BRU	11	1	Unconditional Branch	2-10
BRANCH/SKIP:	SPB	12	2	Store Place and Branch	2-10
	SNS	13-4	1	Skip if Control Switch Not	
	1 (Set	2 - 1 0
	IMS	14	3	Increment Memory and Skip if 0	2-10
	CMA	15	3	Compare Memory and A	
	CMA	15	5	(3 Way)	2-10
	1	1		n+1 if (A)-(M) 0	
	1 1			n+2 if $(A) = (M)$	
	IBS'	00-26	1	n+3 if (A)-(m) 0 Increment B (Index) and	1
	103	00-20	1	Skip if 0	2-11
	SAZ'	00-22	1	Skip if A is Zero	2-11
	SAP'	00-24	1	Skip if A is Positive	2 - 1 2
	SAN'	00-23	1	Skip if A is Negative Skip No Overflow	2-11
	SOF' SAS'	00-25 00-21	1	Skip No Overflow Skip on A Sign (3 Way)	2-12 2-11
	343	00-21	1	n+1(-), n+2(0), n+3(+)	
	SNO	00-32	1	Skip if A is Normalized	2-12
	LOB'	00-36	2	Long branch	2-12
	SXB'	00-50	1	Skip if Index Pointer is Set to B	2.12
	IXS	00-51	1	Increment Index and	2-13
			-	Skip if ≥ 0	2-13
LOGICAL:	AB A'	00-27	1	AND A and B	2-13
	OB A'	00-30	1	OR A and B	2-13
	NEG' ASC'	00-02 00-20	1	Negate A Complement A Sign	2-14 2-14
	CNS'	00-34	1	Convert Number System	2-14
REGISTER	CLA'	00-03	1	Clear A	2-14
CHANGE:	T AB'	00-05	1	Transfer A to B	2-15
	IAB'	00-06	1	Interchange A and B	2 - 1 5
	CSB '	00-07	1	Transfer B sign to Carry and Clear B Sign to Positive	2-15
	TBA'	00-04	1	Transfer B to A	2-15
	TBP	00-40	1	Transfer B-Accumulator to	
		j		Protect Register	2-15
	TPB	00-41	1	Transfer Protect Register	1 2.15
	TAX'	00-52	1	to B-Accumulator Transfer A-Accumulator to	2-15
	1111	00-52	•	Hardware Index Register	2-16
	<u>TXA'</u>	00-53	1	Transfer Hardwarc Index	
	VDV	a	1	Register to A-Accumulator	2-16
	<u>XPX'</u>	00-46	1	Set Index Pointer to Index Register	2-16
	XPB'	00-47	1	Set Index Pointer to B-Ac-	2-10
				cumulator	2-16
	TBV' TVB	00-42 00-43	1	Transfer B-Acc. to VBR Transfer VBR to B-Acc.	2-16 2-16
SHIFT:	RSA	00-10	Time for Shifts	Right Shift A	2-17
	LSA' FRA'	00-11	vary as follows:	Left Shift A	2-17
	FLA'	00-12 00-17	Shifts Time	Right Shift A and B Left Shift A and B	2-17 2-18
	RSL	00-15	1-4 2	Right Logical Shift A	2-17
	FRL	00-14	5-8 3	Full Rotate Logical A and B	
	LSL'	00-16	9-12 4	Left Left Logical Shift A	2 - 18 2 - 18
	FLL'	00-13	13-15 5	Left Logical Shift A and B	2-18
CONTROL:	HLT'	00-00	1	Halt	2-19
	NOP' TO!	00-33	1	No Operation	2-19
	TOI' PIE'	00-35 130600	2	Turn off Interrupt Enable Interrupt	2-19 2-19
	PID'	130600	2	Disable Interrupt	2-19
NDUT/OUTDUT	CEU	10.0004.0	4 + wait	Command External Unit	2.22
INPUT/OUTPUT	CEU' TEU'	13.0IM.0 13.0IM.2	4 + wait 4 + wait	Command External Unit Test External Unit	2-22 2-23
	AOP'	13.01M.2. 1700	4 + wait	Accumulator Word Out to	2-27
	1			Unit	2-23
	AIP'	1702	4 + wait	Accumulator Word In from	
	MOP'	17.0IM.4	4 + wait	Unit Memory Word Out to Unit	2-23 2-24
	MIP'	17,0IM,4 17,0IM,6	4 + wait	Memory Word In from Unit	2-24

a. Underlined instructions require optional hardware.
b. ¹ = Augmented.

98118A.BC