RECOMP II USERS' PROGRAM NO. 1116

PROGRAM TITLE:	LAMP, A Programming Aid for Machine-Language Programming
PROGRAM CLASSIFICATION:	Service
AUTHOR:	Major Wm. F. Luebbert USMA
PURPOSE:	This technical memorandum provides a technical description of the LAMP programming aid developed at the United States Military Academy to simplify the preparation and debugging of machine-language programs for the RECOMP II computer. The version presented here is that available 1 September 1961.
DATE:	1 September 1961

Published by

RECOMP Users' Library
at
AUTONETICS INDUSTRIAL PRODUCTS

A DIVISION OF NORTH AMERICAN AVIATION, INC.
3400 E. 70th Street, Long Beach 5, California

COMPUTER LABORATORY
DEPARTMENT OF ELECTRICITY
UNITED STATES MILITARY ACADEMY
WEST POINT, N. Y.

TECHNICAL
MEMO NO. 4

SUBJECT: LAMP, A Programming Aid for Machine-Language Programming

This technical memorandum provides a technical description of the LAMP programming aid developed at the United States Military Academy to simplify the preparation and debugging of machine-language programs for the RECOMP II computer. The version presented here is that available 1 September 1961.

WILLIAM F. LUEBBERT
Major, Sig C

TABLE OF CONTENTS

1. INTRODUCTION

LAMP (Luebbert's Aid for Machine Programming) is a hybrid program consisting of some of the simplified data entry features now being developed for programming system being developed at the United States Military Academy grafted onto the existing features of PPP-2. If desired LAMP may be used identically to PPP-2, for it retains all of the subroutines and calling sequences, both normal and single instruction, of that service program. However the utility features are consolidated and simplified as follows:

Entry Location	Service
0001 (START 1)	Alphanumeric Input-Output (8 characters/word) (Automatic entry of control word optional) (Automatic verification typeback optional)
0002 (START 2)	Octal Input-Output
0003 (START 3)	Decimal Input-Cutput, Fixed or Floating Point
0004	AN-049 and AN-051 6 character/word alphanumeric input-output (Used in PPP-2) (Automatic Entry of Control Word Cptional) (Automatic Verification Typeback Optional)
0005	Program Trace
0006	Memory Dump, Typewriter or Punch
0007	Zonal Memory Clear

In all cases utility control information may be given from the console in the form

L XXXX START

for those operations which require specification of only one memory location and

L XXXX C

L XXXX START

for those operations which require specification of two memory locations. A "Clear" after the "c" is optional but will give a convertent console display if used.

Alternately, if a typed record of utility entries is desired they may be typed in:

"FILL" L XXXX S

for those operations which require specification of only one memory location and

"FILL" L XXXX C L XXXX S

for those operations which require specification of two memory locations.

2. CONTENTS

Locations	Contents
0001	Utility entry for alphanumeric (8 char/word) input and output
0002	Utility entry for octal input
0003	Utility entry for decimal (fixed or floating point) input and output
0004	Utility entry for PPP-2 AN-049 and AN-051 special alphanumeric (6 char/word) input and output
0005	Utility entry for Program Trace
0006	Utility entry for Typewriter and Punch Memory Dump
0007	Utility entry for Zonal Memory Clear
0010-0147	AN-002 Fixed Point Input Subroutine
0150-0427	AN-015.1 Floating to Fixed Point Output Subroutine (0230-0427 is AN-016 Fixed Point Output)
0530-0767	AN-014 Floating Point Output Subroutine
0770-1047	AN-037 Logarithm Subroutine (Bases 2, e, or 10)
1050-1207	AN-044 Exponential Subroutine (Bases 2, e, or 10)
1210-1327	AN-017.2 Sine Cosine Subroutine
1330-1427	AN-018.1 Arc Tangent Subroutine
1430-1457	AN-020 Arc Sine-Cosine Subroutine
1460-1537	AN-019 Arc Tangent Y/X Subroutine
1540-1567	AN-038 Angle Reduction Subroutine
1570-1617	AN-049 Alphabetic Input Subroutine (6 chars/word)

Locations	Contents
1620-1647	AN-051 Alphabetic Output Subroutine (6 chars/word)
1650-1757	PPP-2 Decimal Number Input-Output Control
1770-2107	AN-008 Octal Input Subroutine
2110-2165	RP-025 Typewriter Memory Dump
2166-2207	RP-026 Punch Memory Dump
2210-2257	PPP-2 Calling Sequences
2260-2307	AN-001 Float Subroutine
2310-2507	Variable Format Floating Point Input
2510-2637	AN-003 Angle Output Subroutine
2640-2667	PPP-2 Angle Input Control
2670-2777	PPP-2 Calling Sequences
3000-3060	Alphanumeric (8 char/word) Utility Input-Output Control
3060-3076	SAIL alphanumeric Input Subroutine (8 chars/word)
3100-3122	LAOS Alphanumeric Output Subroutine (8 chars/word)
3130-3154	Special Alphanumeric (6 char/word) utility Input-Output Control
3155-3163	Memory Dump Utility Control
3170-3217	Memory Clear Control, Typeout and Subroutine
3220-3247	SELECTIVE PUNCH DUMP

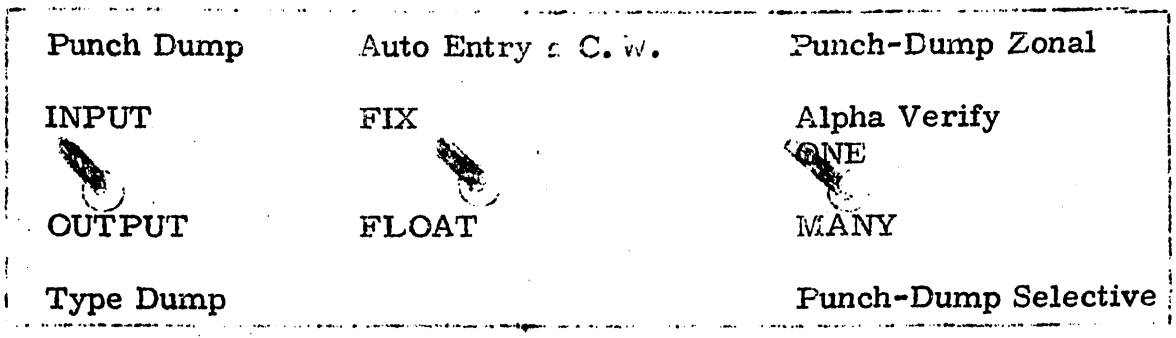
3. USAGE

3.1 UTILITY AND SERVICE

3.1.1 General Information About Input-Output Control. The input-output control features of the LAMP program which are designed to provide utility and service functions have the following basic modes of operation:

Method of Entry	Service Performed
START 1 (or L00010 ENTER START	Alphanumeric Input-Output
START 2 (or L00020 ENTER START	Octal Input
START 3 (or L00030 ENTER START)	Decimal Input (Fixed or Floating Point)

Within these basic modes subordinate modes are determined by the setting of the sense switches. The following template for the sense switch area of the control console provides and abbreviated summary of the subordinate modes:



There are also the following additional modes which are less commonly used and which are not affected by the sense switch settings:

Method of Entry	Service Performed
L00040 ENTER START	AN-049 AN-051 Special Alphanumeric Input-Output
L00050 ENTER START	Program Trace
L00060 ENTER START	Typewriter or Punch Memory Dump
L00070 ENTER START	Zonal Memory Clear

Because the entry of numbers and control symbols from the console keyboard is usually more convenient than entry from the typewriter instructions are written to use this form of entry wherever possible. However if one desires use the typewriter instead one may do so by following these rules:

(a) For Typewriter entry always depress the "FILL" button on the typewriter and make sure the "fill source keyboard" light on the console is extinguished.

(b) Make entries from the keyboard using the following table of equivalencies:

Console Keyboard	Typewriter Keyboard
ENTER	Carriage Return or Tab
CLEAR	X or /
L (location)	L or)
C (command)	C or :
START	S or s
0 through 9	0 through 9

When making entries from the typewriter keyboard one may shift back and forth from letters to figures at will without affecting the data entered into the computer; however, one cannot enter spaces into the data without ruining the entry.

3.1.2 Alphanumeric Input (SAIL Subroutine)

a. SWITCH B OFF (INPUT) Depress START 1. The machine will halt at L3000.

b. Alphanumeric print-out routines use a control word of the form +00XXXXO +OOYYYYO to specify the extent of typeout desired (i.e., all locations from XXXX to YYYY). If at the end of data entry one desires the control word specifing the extent of the alphanumeric data to be automatically inserted into an alphanumeric printout routine calling sequence, turn SWITCH C OFF (Auto Entry Alpha C.W.). Key in to the console LWWWWC where L is the L key just to the left of the key field, WWWW is the four digit address of the location into which the control word is to be inserted, and C is the C key also just to the left of the key field. If auto entry of the control word is not desired turn SWITCH C ON and omit this key-in.

c. If at the end of data entry automatic verification typeout of the data entered and the control word produced is desired turn SWITCH D OFF. If not turn SWITCH D ON.

d. To specify the location where the first word of alphanumeric data is to be stored key in LXXXX START on the console keyboard, where L is the L key just to the left of the key field XXXX is the four digit address of the memory location where the first word is to be packed and START is the start button just to the left of the key field.

e. The typewriter will be set to the figures case and will execute a carriage return. Type in alphabetic information which will be stored in SAIL format.

f. Type blanks slowly until the first and last memory locations used are displayed and the typewriter carriage returns and types out the control word. Then it returns to condition 3.1.2b.

3.1.3 Alphanumeric Output (LAOS Subroutine)

a. SWITCH B ON (Output) Depress START 1. The machine will halt at L3000.

b. To specify the memory location of the start of alphanumeric data key into the console keyboard LXXXXC where L is the L key just to the left of the ten position key field, XXXX is the four digit address of the memory location where the alphanumeric data starts, and C is the C key just to the left of the ten position key field.

c. To specify the memory location of the end of the alphanumeric data, next key into the console keyboard LYYYY START where L is the L key just to the left of the ten position keyboard, YYYY is the four digit address of the last word (the right address of the control word), and START is the start key just to the left of the key field.

d. The typewriter goes to letters shift, executes a carriage return, types out the data between the specified limits, the control word which specified these limits, and returns to condition 3.1.3b.

3.1.4 Octal Input (AN-008 Subroutine)

a. SWITCH B OFF (Input). Depress START 2. The machine will halt at L1752.

b. If only a single word is to be input SWITCH D OFF (one). If more than one word is to be input to successive locations SWITCH D ON (Many).

c. To specify the memory location where the octal data is to be stored key into the console keyboard LXXXX START, where L is the L key just to the left of the ten position key field, XXXX is the four digit address of the memory location, and START is the start key just to the left of the ten position key field.

d. The machine will carriage return and type XXXX.

e. Type on the typewriter a sign (optional) 1 to 13 octal digits, and a carriage return. DO NOT INPUT MORE THAN 16 SYMBOLS INCLUDING SIGN UNDER ANY CIRCUMSTANCES OR PART OF THE MEMORY DUMP PORTION OF LAMP WILL BE DESTROYED.

f. If SWITCH D is OFF (One) the machine returns to condition 3.1.4b. If SWITCH D is ON (Many) the machine will type the next location (XXXX+1) and wait at condition 3.1.4e.

g. SWITCH D may be changed during input, usually while inputting the last number.

h. If an error is made at step 3.1.4e, type X and then carriage return. If this happens unexpectedly the probable cause is typing an 8 or 9.

3.1.5 Octal Output

a. If the combination SWITCH B ON (Output) and START 2 are used the computer will type out the following: "OCTAL OUTPUT NOT AVAILABLE".

b. An octal display of any location may be obtained by depressing the octal display button on the console.

3.1.6 Decimal Input, Fixed Point (AN-0002 Subroutine)

a. SWITCHES B and C OFF, Depress START 3. The computer typewriter types "FILL".

b. If more than one word is to be input to successive locations in memory SWITCH D ON (Many). If individual entries are to be made to various locations or if only a single entry is to be made SWITCH D OFF (One).

c. To specify the memory location where the fixed point data is to be stored key into the console keyboard LXXXX START, where L is the L key just to the left of the ten position key field, XXXX is the four digit address of the memory location, and START is the start key just to the left of the ten position key field.

d. The computer typewriter will carriage return and type "B". This indicates that the binary point of the data should be entered from the typewriter as follows:

(1) Sign (optional)
(2) The number (always an integer smaller than 40), decimal point optional.
(3) Enter

e. Type on the computer typewriter the location in which the decimal data is to be stored. Type in the number as follows:

(1) Sign (optional)
(2) The number (decimal point optional for integers)
(3) Enter

f. If SWITCH D is OFF (One) the computer will return to condition 3.1.6b. If SWITCH D is ON (Many) the computer will type the next location and return to condition 3.1.6e. When inputting the last number of a sequence move SWITCH D to OFF (One) so as to return to location 3.1.6b.

3.1.7 Decimal Output, Fixed Point (Subroutine AN-016)

a. SWITCH B ON, SWITCH C OFF, Depress START 3. The computer typewriter types "FILL" and stops at L1653.

b. If more than one word is to be output from successive locations in memory SWITCH D ON (Many). If individual words from scattered memory locations or only a single word output is required SWITCH D OFF (One).

c. To specify the memory location from which the fixed point data is to be output, key into the ten position keyboard LXXXX START, where L is the L key just to the left of the ten position keyboard, XXXX is the four digit address of the memory location and START is the start key just to the left of the ten position keyboard.

d. The computer typewriter will carriage return and type B to signify that the binary point of the data should be keyed into the ten position keyboard as follows:

(1) Sign (Optional)
(2) The number (Always an integer less than 40, decimal point optional)
(3) Enter

e. The typewriter will type out the decimal number. If the binary point is in the range -13 B 39, the typeout will be in fixed point format. If not it will be output in floating point form.

f. If SWITCH D is OFF (One) the computer will return to condition 3.1.7b. If SWITCH D is ON it will type successive locations until SWITCH D is turned OFF.

3.1.8 Decimal Input, Floating Point (Subroutine AN-031)

a. SWITCH B OFF, SWITCH C ON, Depress START 3. The computer typewriter types "FILL".

b. If more than one word is to be input to successive locations in memory SWITCH D ON(Many). If individual words are to be input to scattered locations or only a single word input is required SWITCH D OFF (One).

c. To specify the memory location to which the data is to be input, key into the console keyboard LXXXX START, where L is the L key just to the left of the ten position key field, XXXX is the four digit address of the memory location, and START is the start key just to the left of the ten position key field.

d. The computer typewriter will carriage return and type the memory location. Type the number to be stored into the computer typewriter as follows:

(1) Sign (optional if plus)
(2) Integral part of number (if any)
(3) Decimal point followed by fractional part of number(if any)
(4) Sign and power of ten exponent. NOTE: EXPCNENT, IF PRESENT, MUST BE AN INTEGER.
(5) ENTER

Almost any logically reasonable abbreviation of this complete form which terminates in an ENTER may be used. Specifically, the following combinations are acceptable: (2, 5) (3, 5) (1, 2, 5) (1, 3, 5) (2, 4, 5) (3, 4, 5) (1, 2, 4, 5) (1, 3, 4, 5) (5 - interpreted as +0) (1, 5 - interpreted as a signed 0), or the complete (1, 2, 3, 4, 5).

e. If SWITCH D is OFF the computer will return to condition 3.1.8b. If SWITCH D is ON a new location two higher than the last one will be typed (a floating point number requires two words for storage) and the computer will wait at condition 3.1.8d. When inputting the last number of a sequence to be stored in successive locations move SWITCH D to OFF (One) so as to return to condition 3.1.8b.

3.1.9 Decimal Output, Floating Point (Subroutine AN-014)

a. SWITCH B ON, SWITCH C ON, Depress START 3. The computer typewriter types "FILL".

b. If more than one word is to be input to successive locations in memory SWITCH D ON (Many). If individual words are to be input to scattered locations or only a single word input is required SWITCH D OFF (One).

c. To specify the memory location to which the data is to be output, key into the console keyboard LXXXX START, where L is the L key just to the left of the ten position key field, XXXX is the four digit address of the memory location, and START is the start key just to the left of the ten position key field.

d. The machine types the location and its contents. If SWITCH D is OFF it will return to condition 3.1.9b. If SWITCH D is ON it will continue typing until SWITCH D is turned OFF, then on the completion of the word being typed revert to condition 3.1.9b.

3.1.10 Special Alphanumeric Input (PPP-2 AN-049 6 chars/word format)

a. Set program counter to L0004 (Depress L 00040 ENTER)

b. SWITCH B OFF, Depress START. Computer will halt at L3130

c. Alphanumeric print out routines use a control word of the form +00XXXX0 +00YYYY0 to specify the extent of typeout desired (i.e., all locations from XXXX to YYYY). If at the end of data entry one desires the control word specifing the extent of the alphanumeric data to be automatically inserted into an alphanumeric printout routine calling sequence, turn SWITCH C OFF (Auto Entry Alpha C.W.). Key in to the console LWWWWC where L is the L key just to the left of the key field, WWWW is the four digit address of the location into which the control word is to be inserted, and C is the C key also just to the left of the key field. If auto entry of the control word is not desired turn SWITCH C ON and omit this key-in.

d. If at the end of data entry automatic verification typeout of the data entered and the control word produced is desired turn SWITCH D OFF. If not turn SWITCH D ON.

e. To specify the location where the first word of alphanumeric data is to be stored key-in LXXXX START on the console keyboard, where L is the L key just to the left of the key field XXXX is the four digit address of the memory location where the first word is to be packed and START is the start button just to the left of the key field.

f. The typewriter will be set to the figures case and will execute a carriage return. Type in alphanumeric information which will be stored in the AN-049 (PPP-2) format.

g. Type blanks slowly until the first and last memory locations used are displayed and the typewriter carriage returns and types out the control word. If a verification printout is required the computer will halt before the control word is typed out. Depress START and both will type out.

3.1.11 Special Alphanumeric Output (PPP-2 AN-051 6 chars/word format)

a. Set program counter to L0004 (Depress L00040 ENTER)

b. SWITCH B ON, Depress START. Computer will halt at L3130

c. To specify the memory location of the start of alphanumeric data key into the console keyboard LXXXXC where L is the L key just to the left of the ten position key field, XXXX is the four digit address of the memory location where the alphanumeric data starts, and C is the C key just to the left of the ten position key field.

d. To specify the memory location of the end of the alphanumeric data, next key into the console keyboard LYYYY START where L is the L key just to the left of the ten position keyboard, YYYY is the four digit address of the last word (the right address of the control word), and START is the start key just to the left of the console key field.

e. The typewriter goes to letters shift, executes a carriage return, and halts. Depress START again. The computer types out the data between the specified limits, and the control word which specified these limits.

3.1.12 Program Trace

a. Set program counter to L0005 (Depress L00050 ENTER).

b. Depress START. Early LAMP tapes do not contain the trace and will type out NO TRACE repeatedly.

3.1.13 Punch Memory Dump

a. Set program counter to L0006 (Depress L00060 ENTER).

b. SWITCH B OFF, Depress START. The computer will halt at L3155. Feed out about 18 inches of tape by depressing the tape advance button at the tape punch.

c. To specify the first memory location to be punched onto tape, key into the console keyboard L XXXX C where L is the L key just to the left of the ten position keyfield, XXXX is the four digit address of the memory location where the dump should start, and C is the C key just to the left of the ten position key field.

d. To specify the last memory location to be punched key into the console keyboard L YYYY START where L is the L key just to the left of the ten position keyfield, YYYY is the four digit address of the final memory location to be punched, and START is the start key just to the left of the ten position keyfield.

e. The required memory locations will be punched out in alphanumeric format.

3.1.14 Typewriter Memory Dump

a. Set program counter to L0006 (Depress L00060 ENTER).

b. SWITCH B ON, Depress START. The computer will halt at L3155.

c. To specify the first memory location to be typed out, key into the console L XXXX C where L is the L key just to the left of the 10 position keyfield, XXXX is the four digit address of the first memory location to be typed out, and C is the C key just to the left of the ten position keyfield.

d. To specify the last memory location to be typed out, key into the console L YYYY START, where L is the L key just to the left of the ten position keyfield, YYYY is the four digit address of the final location to be typed out, and START is the start key just to the left of the ten position keyfield.

e. The computer will carriage return and begin typing the contents of the memory locations in command format. Every 8 locations it will double space and type the address of the first of the next block of 8 locations which are typed in command format.

3.2 SUBROUTINES

A subroutine is a set of instructions necessary to direct a computer to carry out a well-defined mathematical or logical operation, e.g., a sine-cosine subroutine.

A subroutine may be open. That is, it is inserted directly into the linear operational sequence and is not entered by a jump. Such a subroutine must be recopied at each point that it is needed in a routine. Since this is extremely wasteful of space and coding time, most subroutines are closed.

A closed subroutine is not stored in its proper place in the linear operational sequence, but is stored away from the routine which refers to it. Such a subroutine is entered by a jump, and provision is made to return, i.e., to jump back to the proper point in the main routine at the end of the subroutine.

In RECOMP, the return is made possible by using the SAX instruction to see where the jump to the subroutine was and usually jumping back to a point two or three instructions past the jump. The exact return defined as "NORMAL RETURN" when no errors occur, or as "ERROR RETURN" when an error occurs, is specified in the calling sequence.

The LAMP program contains all the PPP-2 closed subroutines plus a few more. Two means of entry are provided as in PPP-2:

(a) Normal subroutine entries as described for the individual subroutine in the subroutine manual. This method of entry allows one to take full advantage of all the flexibility of the subroutine, select printouts or other action for error returns, etc.

(b) Simplified entry by means of a single transfer command. Such entries generally require the argument to be used to be stored in the A and R registers before entry and leave the result in the A and R registers. These entries require more computer running time.

3.2.1 EXPLANATION OF PSEUDO OPERATIONS AND CALLING SEQUENCES

3.2.1.1 Explanation of Notation

A pseudo-operation is an operation which, ordinarily, does not affect the state of any of the arithmetic registers and leaves the bit configuration of words in the main memory and high speed loops unchanged. These operations are used frequently during the preparation of symbolic programs and to indicate information to be supplied when control is transferred to a subroutine.

The following pseudo-operations are used in the subroutine write-ups in this manual:

MZE (Minus Zero) - This pseudo-operation, when appearing in the program, is synonomous with -00. Since pseudo-operations are not executed, the minus sign does not cause trapping to take place. The half-word containing this operation is examined by the subroutine and could indicate, for example, that a negative quantity is to be computed by the routine. The 00 usually indicates that no information will be gained by inspection of the first six bits of the half word. In most cases, the 00 could be replaced by any other combination of two octal numbers, but for standardization and convenience, the zeroes are used.

PZE (Plus Zero) - This is essentially the same as MZE, except that the plus sign on +00 usually indicates that the sign is either not interpreted or a positive quantity is to be computed.

SLL (Set Location Left) - This pseudo-operation indicates that the next meaningful operation must be contained in the left half of the next full word. During assembly of the program this operation can be replaced by any operation which will not affect the state of the arithmetic registers or memory locations. For example, +40 0000.0, which shifts the contents of the accumulator zero places. Note that, in some instances, it will not be necessary to replace this code by an absolute code.

The following example will illustrate the latter case:

```
2021.0 + 60     STØ  A
2021.1 + 00     CLA  B
                SLL
2022.0 + 57     TRA  ARC
2022.1 + 60     STØ  C
```

The SLL in the symbolic coding indicated that the instruction TRA ARC was to be contained in the left half of a word. Since the CLA B instruction occurred in the right half of word 2021, the TRA ARC occurred in the left half of word 2022 without the use of a forcing instruction.

SLR (Set Location Right) - This pseudo-operation is similar to SLL except that it is an indication that the next meaningful instruction must be contained in the right half of a word.

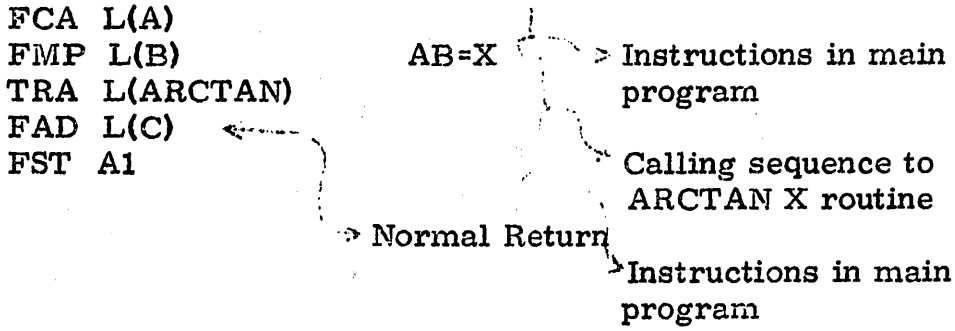
3.2.1.2 Example, Normal Entry

A calling sequence is an instruction or sequence of instructions and pseudo-instructions which interrupts a program with a transfer to a subroutine, provides information concerning parameters needed for the execution of a subroutine, allows flexibility of correction procedures in case of error during the execution of a subroutine.

When it is necessary to include a subroutine in a given program, usually the next available location is used. Into this location, and perhaps the next one, the calling sequence is inserted. (See example 1)

The number of instructions in a calling sequence will vary depending on the information which must be supplied by the programmer in order that the purpose of a subroutine may be accomplished.

Example 1:



In order to compute the Arctangent of a quantity X, it is necessary to have the floating point argument in the A and R registers. The FCA L(A) and FMP L(B) instructions left X=AB in A and R so that no other instructions were necessary to meet this requirement. (If X had been computed previously, one instruction (FCA L(X)) would have satisfied the requirement). The TRA L(ARCTAN) instruction interrupts execution of the instruction in the main program, allows computation of Arctangent X, which is left in the A and R registers, and the next command which is executed is FAD L(C). The address of the FAD command is called the NORMAL RETURN address, which simply means that when computation of Arctangent X is complete, control will automatically be transferred to that address in the main program.

The actualccalling sequence of the Arctangent X subroutine, since it must pertain to any program rather than a specific program, is as follows:

```
α       FCA   L(X)
        TRA   L(ARCTAN)
α + 1   NORMAL RETURN
```

where α is any symbolic location and α+1 is one full word beyond α. Note that the instruction at location α could be replaced by any sequence of instructions which could leave the argument X in the A and R registers, and the instruction at location α+1 could be any instruction that the programmer wishes to place in that position.

Example 2:

In order to compute a quantity e^X using the Exponential Subroutine, the following calling sequence must be used:

1) (X in A and R registers)

2) SLR

3) TRA L(EXP)

4) PZE Code

Code = 7772 for base 2
7774 for base e
7770 for base 10

5) ERROR RETURN

6) NORMAL RETURN

The symbolic coding in a program which would correspond to this calling sequence could be as follows:

```
Main Program        FST   BAN
                    FCA   L(X)
                    SLR

Calling Sequence    TRA   L(EXP)
                    PZE   7774
                    HTR   E1

Main Program        FST   ETOX
                    FCA   Q
```

It can be easily seen that the coding in this sample program meets the requirements imposed by the calling sequence, since:

1) FCA L(X) places X in the A and R registers.

2) SLR will insure that the next instruction is executed from the right half of a word.

3) TRA L(EXP) provides for control to be transferred from the main program to the beginning of the subroutine.

4) PZE 7774 is the code which means that the base e is to be used rather than base 2 or base 10.

5) HTR E1 will cause computation to cease with the location counter set at location $E1_x$ in case an error is detected in computing e^x. In this case an ERROR RETURN would indicate that the exponent of X is greater than 35 and X is greater than zero.

6) FST ETOX is the NORMAL RETURN. This instruction will be executed automatically immediately following the correct computation of e^x.

3.2.1.3 Example, Simplified Entry

To input an angle in degrees, minutes and seconds take its tangent; take the natural logarithm of its tangent; output the answer; carriage return and repeat, the following short program will be all that is needed:

```
L   +5726400 + 5722301
    +5727001 + 5722531
    +7200100 + 57  L
```

3.2.2 List of One-Instruction Calling Sequences

1.	Read Z (Used by SALT and SCRAP only)	+57 22100
2.	Input (AN-007.1)	+57 22140
3.	CXP (Used for debugging. Prints full word location and contents of A and R in floating point format; restores A and R. Salt generates TSB CXP)	+ 57 22171
4.	Tangent (Error prints "TAN")	+57 22301
5.	Power (a^b for a o. a in A and R, b in 1764, 5)	+57 22411

6.	Output (Typeout in fixed format if possible, if not possible, typeout in floating point.)	+57 22531
7.	Angle Input (Degrees, minutes, seconds using AN-007.1 3 times. Sign of the angle is same as sign of degrees)	+57 26400
8.	Angle Output (AN-003)	+57 25100
9.	Logarithm Base 2 (Error prints "LOG")	+ 57 26700
10.	Logarithm Base e (Error prints "LOG")	+57 27001
11.	Logarithm Base 10 (Error Prints "LOG")	+57 27030
12.	Exponential Base 2 (Error Prints "EXP")	+57 27100
13.	Exponential Base e (Error prints "EXP")	+57 27201
14.	Exponential Base 10 (Error Prints "EXP")	+57 27230
15.	Sine (Angle reduction automatic, error prints "SIN")	+57 27261
16.	Cosine (Angle reduction Automatic, error prints "COS")	+57 27361
17.	Arc Sine (Error prints "ARCSIN")	+57 27461
18.	Arc Cosine (Error prints "ARCCOS")	+57 27600
19.	Arc Tangent	+57 13300
20.	Angle Reduction	+57 15400

3.2.3 List of Subroutines With Normal Calling Sequences

The calling sequences for the list of subroutine contained in the package are as follows(the argument is assumed to be in A and R unless otherwise mentioned--see write-ups for more detailed explanations);

3.2.3.1 AN-007.1 Variable Format Floating Point Input

+57 23100
ERROR RETURN
NORMAL RETURN

3.2.3.2 AN-002 Fixed Point Input

```
+00   L(B)
+57   00100
ERROR RETURN
NORMAL RETURN
```

L(B) is the location of B, the Binary Point, at b39.

3.2.3.3 AN-015.1 Floating to Fixed Point Output

MODE 1	MODE 2
SLR	SLR
+57 01500	+57 01500
+00 LLRRC	-00 00000
ERROR RETURN	ERROR RETURN
NORMAL RETURN	NORMAL RETURN

LL and RR are 2-digit octal numbers denoting the number of digits to be printed to the left and right, respectively, if the decimal point.

3.2.3.4 AN-016 Fixed Point Output (the number must be in A, and its binary point in R at b39)

```
SLR
+57 02300
+00 LLRRC
NORMAL RETURN
```

LL and RR are 2-digit octal numbers denoting the number of digits to be printed to the left and right, respectively, of the decimal point.

3.2.3.5 AN-031 Floating Point Input

TYPEWRITER	TAPE
+57 04320	+57 04300
NORMAL RETURN	NORMAL RETURN

3.2.3.6 AN-014 Floating Point Output

```
SLR
+57 07700
+00 CODE

ERROR RETURN CONTROL
NORMAL RETURN
```

BASE	CODE
2	77720
ee	77740
10	77700

3.2.3.8 AN-044 Exponential, 2, e, 10 (See AN-037 for Code)

```
SLR
+57 10500
+00 CODE
ERROR RETURN
NORMAL RETURN
```

3.2.3.9 AN-017.2 Sine, Cosine

```
SLL
+57 10500
+00 L(Sin)
ERROR RETURN
NORMAL RETURN
```

The sine is stored in L(Sin) and the Cosine in L(Sin) +2.

3.2.3.10 AN-018.1 Arc Tangent

```
+57 13300
NORMAL RETURN
```

3.2.3.11 AN-020 Arc, Sine, Cosine

```
SLL
+57 14300
+00 L(Arc Sin)
ERROR RETURN
NORMAL RETURN
```

The Arc Sine is stored in L(Arc Sin) and the Arc Cosine is in the A and R registers.

3.2.3.12 AN-019 Arc Tangent Y/X (Arguments in L(Y) and L(X))

```
SLR
+57 14600
+00 L(Y)
+00 L(X)
ERROR RETURN
NORMAL RETURN
```

L(Y) is the location of Y. L(X) is the location of X.

3.2.3.13 AN-038 Angle Reduction

```
+57 15400
NORMAL RETURN
```

3.2.3.14 AN-049 Special Alphabetic Input (6 chars/word)

```
SLL
+57 16100
+00 LOC
NORMAL RETURN
```

LOC is the location of the first packed word.

3.2.3.15 AN-051 Special Alphabetic Output (6 chars/word)

```
SLR
+57 16450
+00 L(S)
+00 L(F)
NORMAL RETURN
```

3.2.3.16 SAIL Alphanumeric Input (8 chars/word)

```
SLL
+57 30600          XXXX = Location of start of alphanumeric
+00 XXXXO                 data
NORMAL RETURN
```

3.2.3.17 LAOS Alphanumeric Output

```
SLR
+57 31000
+00 XXXXO          XXXX = location of first word of printout
+00 YYYYO          YYYY = location of last word of printout
NORMAL RETURN
```

REPRESENTATIVE FLOW CHART
UTILITY & SERVICE PORTION
LAMP