Wallenmeyer

704 Off-line Sub - Routines

MURA - 598
Donald Dickman

LTR: Library Tape Routines

I. Purpose:

With the installation of a 32K core on the 704, it becomes desirable to minimize the output print time. This is accomplished by a 1401 system and routines that write all output on tape. This tape can then be mounted on the 1401 and listed on the high speed printer (1403). The 1401 is scheduled to be delivered in January. The library Tape Routines are an attempt to partially satisfy the concept of all 704 output "off-line". They use logical tape 9 as the output tape and do all writing in BCD. The LTR programs listed here are not assumed to be complete or even the best possible versions. They are merely a first attempt to more efficiently use the 704.

II. Types and Availability:

There are 8 routines, divided into 3 types. Types A & B are available in 2 ways: from a special library tape or on absolute binary cards. In the write-ups which follow whenever these routines are mentioned it is assumed that either version may be used. If the tape version is desired, the "Special Library Tape" must be mounted on Unit 1. These routines may then be pulled off tape by 1 card self-loaders which are available.

Type C Routines are available, at present, only on relocatable binary cards. These cards can be translated by MU RAT 1.

Already in the planning stages are different versions to improve these routines now available. It is hoped that they will all eventually be on tape and part of an overall monitor system.

The 3 types are as follows:

A. Header program - LTR001.
B. Dump programs - LTR002, LTR003, LTR004.
C. Print programs - LTR005, LTR006, LTR007, LTR008.

With the exception of LTR002, these programs are modified and recoded versions of Los Alamos routines. They have been recoded to be used in the SAP system (Los Alamos does not use the SAP system).

Write-ups of the 8 routines follow. It should be re-emphasized that with 1 exception (as noted in the LTR008 write-up), "print" or "dump" in all the write-up is understood to mean WRITING ON THE BCD OUTPUT TAPE (logical tape 9) for reproduction on the 1401 and 1403 printer.

LTR001	:	Print Heading Off Line
Purpose	:	To write 1 BCD record on the output tape, (9), for each alpha-numeric card read on line.
Deck make-up	:	LTR001 N Alpha-numeric cards 1 Blank card Self - Loader or nothing
Description	:	LTR001 pulls itself into memory and writes on the output tape one (1) record for each card following until a blank card is read. It then gives up control by "pushing the load cards button". If no self loader follows the blank card, LTR001 will stop at Loc. 0.
Restrictions	:	Column 1 of each alph-numeric card must be blank. An all blank card must follow the alpha-numeric cards before the self-loader of the users program.

LTR002 : Special Octal Dump

Purpose : To dump, in octal, a portion of memory defined by a control card or setting of the console keys.

Deck make-up : LTR002
N Control cards (if desired)
Blank card, self-loader,
Transfer card, or nothing

Control card format : 1 word per card
9 row right, binary punched:
FWA in decrement
LWA in address

Console keys format : FWA in decrement
LWA in address

Description : LTR002 pulls itself in memory destroying locations $0\text{-}242_8$ ($0\text{-}162_{10}$) and saves the AC and index registers. It then prints out the contents of the AC and index registers. LTR002 then "Halts" at 76_8.

If control cards are used, push "START" for first dump. If keys are desired, enter control word by "enter MQ" and push START. After completing each dump at halt at 76_8 ensues. More than 1 dump may be taken by repeating the the above process. Upon encountering a blank card, LTR002 will restore the console and "Halt". Upon encountering a self-loader or transfer card, it will restore the console (AC, XA, XB, XC) and "push the load cards button". LTR002 skips 2 spaces between dumps.

Output Print format :

Console:
AC = XXXXXXXXXXXX
XA = XXXXX
XB = XXXXX
XC = XXXXX

Memory : address XXXXX word XXXXXXXXXXXX

One word per line. All zero words are omitted. Minus signs print by adding a high order bit to left-most digit. Thus:

- 3777 ---- will be 7777 ----
- 0777 ---- will be 4777 ----

LTR003 : 6/Line Octal Print

Purpose : To dump, in octal, those portions of memory specified by control cards.

Deck make-up : LTR003
N control cards
Blank card, transfer card with col. 1 punched, or nothing.

Control card format : 9 row left punched

columns	contents
1 - 3	no punches
4 - 18	FWA
22 - 36	LWA

Description : LTR003 pulls itself into memory destroying locations $0\text{-}303_8$ ($0\text{-}195_{10}$). It prints the contents of the AC and index registers. Then it selects the card reader and prints the contents of core as specified by the control cards. A blank card or EOF will cause the routine to "Halt" at 73_8. LTR003 will recognize a transfer card if col. 1 is punched (TNX) or any other self-loader that has the 9L sign punched. It will give up control by pushing the "load cards button".

LTR003 does not print mnemonics or restore the console.

Print format : Console

C(AC) C(XA) C(XB) C(XC)

± XXXXXX XXXXXX XXXXX XXXXX XXXXX

Line

1st word loc. 1st word 6th word

XXXXX ±XXXXXX XXXXXX-----±XXXXXX XXXXXX

All zero lines do not print. Plus signs do not print.

Speed : Approximately 2500 lines of non-zero words per minute.

LTR004 : General Dump Program and Binary Card Compare

Purpose : To dump selected protions of memory (in octal or floating decimal), as specified by control cards and/or compare absolute binary cards with their respective locations in core. The console keys may be used instead of control cards for dumping memory.

Options : LTR004 recognizes the following types of cards:

A. No cards: If no cards follow LTR004; after saving and printing the console, it will "Halt" at 242_8. The user then enters the limits of the dump via the MQ and pushes "Start". (FWA in Dec., LWA in address).

B. Blank card: If it is the first card following LTR004, starting at 1750_8 (1000_{10}) the next 8192 words will be printed. Any subsequent blank card will restore the console and give up control by pushing the "load cards" button for the next card.

C. Control cards: Upon encountering a control card, LTR004 will dump that portion of memory as specified by the control card, in the desired mode (as specified on the control card).

Control card format: Punched in binary, 9 row left

Columns		Contents
1		Blank
2,3		print mode
	00 =	octal
	01 =	floating decimal if normalized, otherwise octal.
	10 =	floating decimal, normalized or not.
4-18		FWA
22-36		LWA
37-72		Must be blank

D. Ordinary (non-relocatable) Binary Cards: If LTR004 encounters a binary card, it will compare the words on it with the corresponding locations in memory and print out any differences. These cards can be before, after, or intermixed with the control cards.

E. Self-Loaders: If, upon encountering a binary card, LTR008 sees that the 9 Row Left decrement is greater than 22_{10}; it will give up control by pushing the "load cards" button <u>for this card</u>.

F. Transfer cards: Any card with the 9L1 punched (TNX) will be treated as a transfer card and control will be turned over to it.

Special Features :

A. Any time during operation, press RESET and the 704 will stop at location zero. Press START; the console will be restored (except MQ, instruction counter, and instruction register) and the "load cards" button pushed.

B. LTR004 will print-mnemonics in the octal dump mode with the exception of the sense instruction, which will print as "MSE".

Description : LTR004 pulls itself into memory destroying locations 0 - 1644_8 (0 - 932_{10}). If the binary card compare feature is used, add 75_{10} to the number of locations destroyed. It prints out the status of the console and saves it (except for the MQ, instruction counter, and instruction register). It then attempts to read a card and depending on whether a card is present or what kind of card it is, proceeds as previously noted.

Deck make-up : LTR004
One or more of the cards previously noted.
Another program, if desired.

Restrictions :

A. If LWA = 0, it will be changed to the last location of core.

B. If LWA $<$ FWA, LWA will be changed to the last location of core.

C. If LWA - FWA $>$ 8192, only 8192 words will be printed. Only words actually printed count in this total.

D. A line of printing is deleted if the entire line would be the same as the last number on the last line printed.

E. LTR004 will only recognize transfer cards with 9L1 punched (TNX) and Self-loaders whose 9L decrement is greater than 22_{10}.

Speed : Approximate - non zero words
Floating Decimal : 1350 lines/min.
Octal : 2250 lines/min.
Card compare : Full speed of card reader unless words are found that do not agree, then half speed.
Formats are on next page.
Plus Signs do not print

CONSOLE: 1st line LTR004 Print Formats

```
     ON)                                                          ON)
AC=(OFF)   xxxxxx   xxxxxx   ± . xxxxxxx  ± EE   MQ = (OFF)   con.
  (a)            (b)                  (c)                (d)

                 ON)
          TCK = (OFF)   A = xxxxx   B = xxxxx   C = xxxxx    con.
                (e)                  (f)

                      ON)
                DCK = (OFF)          SL = 1001
                    (g)                  (h)
```

a. accumulator over flow
b. octal accumulator (4 added to first digit if minus)
c. floating decimal accumulator
d. MQ overflow
e. tape check indicator
f. index registers in octal
g. divide check indicator
h. sense lights 1, 2, 3, 4 (0 = OFF, 1 = ON)

OCTAL FORMAT: 6 words per line

```
Rxxxxx   LLL   ±xxxxxx   xxxxxx    LLL   ± xxx - - -   etc.
  (a)    (b)          (c)
```

a. location of first word on line
b. mnemonic
c. octal word

FLOATING DECIMAL FORMAT: 8 numbers per line

```
Rxxxxx   ± . xxxxxxx ± EE    ± . xxxx  - - - - - -    etc.
```

CARD COMPARE FORMAT

```
LOC.        WORD FROM MEMORY          WORD FROM CARD

Rxxxxx      LLL  ±xxxxxx  xxxxxx      LLL  ±xxxxxx   xxxxxx
  (a)       (b)         (c)
```

a. location in core and on card
b. mnemonic
c. octal word

LTR005 : 10/Line Floating Point Print

Purpose : To print a single line of floating point data, 10 numbers per line, each number consisting of a signed 5 digit fraction and a signed 2 digit exponent. Routine also prints a 1 digit ID.

Format : ID 1st word - - - - - - - - - - 10th word

x ± x.xxxx ± xx ± x.xxxx ± xx

Plus signs do not print

Calling sequence : Words are assumed to be in locations COMMON + 30 through COMMON + 39.

Loc.	Op.	Addr.	Tag.	Dec.
L	TSX	LTR005	4	0
L+1	PZE	I	b	M
L+2	Control	returns	here	

Unormalized data is not normalized by routine.

I = ID, an integer whose range is zero through 9. If I is greater than 9 it will be printed modulo 10 to assure being only 1 digit.

b = Skip indicator

b = 0, no skip before printing (single spacing)
b = 1, ½ page skip before printing
b = 2, 1 line skip before spacing (double spacing)
b = 3, skip to new page before printing

M = No print indicator; a one (1) in any position indicates that corresponding word on the line is to be left blank.

Example: Sap. coding

Col. 8	12
TSX	LTR005, 4, 0
PZE	9, 2, 341

This will result in an ID of 9, double spacing and only words 1, 3, 5, 7, 9 printed.
($341_{10} = 525_8 = 0101010101$)
M = 0, Print all 10 numbers

Print Wheel Positions:

Number	Print Wheels		
First	3	through	12
Second	15	"	24
Third	27	"	36
Fourth	39	"	48
Fifth	51	"	60
Sixth	63	"	72
Seventh	75	"	84
Eighth	87	"	96
Ninth	99	"	108
Tenth	111	"	120

Speed : Non-zero words, approximatly 1250 lines/minute.

Storage : Routines occupies 195 locations and uses COMMON through COMMON + 39. (words to be printed are stored in COMMON + 30 through COMMON + 39).

Routine stores and restores XR1 and XR2 before exit to return.

LTR006 : 6/Line Floating Point Print

Purpose : To print a line of floating point data consisting of an ID word, ID number, and 6 floating point numbers, each having a signed 8 digit fraction and a signed 2 digit exponent.

Print format :

ID word	ID no.	1st word	6th word
LLLLL	xxx	±x.xxxxxxx±EE-----	±x.xxxxxxx±EE

Plus signs do not print.

Calling sequence : Words assumed to be in COMMON + 30 through COMMON + 35.

Loc.	Op.	Address	Tag	Decrement
L	TSX	LTR006	4	0
L+1	←ID	word	→	
L+2	PZE	ID no.	b	M
L+3	Control	returns	here	

Unormalized data is not normalized by routine.

ID word : Must be in BCD (octal) in calling sequence. First 6 bits must be zero.

ID no. : An integer in the range 0 - 999, a number greater than 999 will be printed modulo 1000.

b : Skip indicator

b = 0, no skip before printing (single spacing)
b = 1, ½ page skip before printing
b = 2, 1 line skip before printing (double spacing)
b = 3, restores page before printing

M : No print indicator
A one (1) in any position indicates that corresponding number is to be left blank.

Example : Sap. Coding

Col. →	8	12
L	TSX	LTR006, 4, 0
L+1	OCT	006521474651
L+2	PZE	97, 2, 42
L+3		Control returns here

This will result in an ID word of "VAPOR", an ID number of 97, double spacing, and only the 2nd, 4th, and 6th words being printed, ($42_{10} = 52_8 = 101010_2$)
M = zero, print all 6 numbers.

Print wheel positions:

Number	Print wheels		
First	17	through	29
Second	35	"	47
Third	53	"	65
Fourth	71	"	83
Fifth	89	"	101
Sixth	107	"	119

Storage : LTR006 occupies 215 locations in memory and uses locations COMMON through COMMON + 35 (numbers to be printed are stored in COMMON + 30 through COMMON + 35)

Speed : Approximately 1600 lines per minute for non-zero numbers.

LTR007 : 6/Line Octal Print

Purpose : To print 1 line of octal data consisting of 6 words, each 12 digits long.

Print format :

```
1st word                                  6th word
± xxxxxx    xxxxxx -------  ±xxxxxx    xxxxxx
```

Plus signs do not print.

Calling sequence : Words assumed to be in COMMON through COMMON + 5

loc.	op.	Address	Tag	Dec.
L	TSX	LTR007	4	0
L+1	PZE	M	0	0
L+2	Control	returns here		

M : No print indicator

A zero in any position indicates the corresponding word is to be blank.

Example: Sap. Coding

Col.	8	12
	TSX	LTR007, 4, 0
	PZE	43, 0, 0

This will print the first, third, fifth, and sixth words in a line.

$(43_{10} = 53_8 = 101011_2)$

M = zero, no words printed.

Storage : LTR007 occupies 60 locations and uses COMMON through COMMON + 27.

Speed : Approximately 2500 lines per minute.

LTR008 : General Print Program

Purpose : To set up and print 1 line of data. Any desired format may be used and is specified by the programmer. The following types of conversions and print outs are possible:

Floating binary to floating decimal
Floating binary to fixed decimal
Fixed binary to fixed decimal
Binary integer to decimal integer
Binary integer to octal integer
BCD to Hollerith

Special note : This program is the exception to the rule that all LTR write "off-line" via the output tape (logical tape 9).

LTR008 offers the option of off line printing or on line printing or both. The desired method is specified in the calling sequence. For "on line" printing, the SHARE # 2 print panel is used.

Calling sequence : Because of the flexibility of the program there is no standard calling sequence, except for the first line (TSX LTR008, 4, 0).

A calling sequence is then formed with one calling sequence line required for each word to be printed in the line. Blocks of data may be printed out as indexing can be used in the calling sequence. Table 1 lists the type of conversions possible in the calling sequence.

OP	ADDR.	TAG	DEC.	CONVERSION
SVN	L+C(X)	X	1000D+PP	FLOATING TO FLOATING
SIX	"	"	"	FLOATING TO FIXED
FOR	"	"	PP	INTEGER TO INTEGER
PTH	"	"	1000N+PP	BCD TO HOLLERITH
FVE	"	"	1000D+PP	FIXED TO FIXED
PZE	T	0	0	BINARY POINT FOR FIXED TO FIXED
PTW	0	0	INST	OFF LINE PRINTING
PON	INST 1	0	INST 2	ON LINE PRINTING

Examples are given later in the write-up.

Explanation of symbols:

L : Location of word to be printed. L+C(X) is used to indicate that L is the effective address when indexing is used to print out a block of data. This is illustrated in the examples.

X : Index register used when printing out a block of data. May be 1, 2, or 3. If no indexing is desired, it should be zero.

D : Floating point numbers; the number of digits wanted in the fraction. For example: D = 8 would yield $\pm$ X.XXXXXXX $\pm$ EE.

D : Fixed point numbers; the number of places to the right of the decimal point.

In either case of "D", D $\leq$ 8.

N : Number of 36 bit BCD words to be picked up for out-put.

PP : Right most print position to be used ($1 \leq PP \leq 120$).

T : Binary point placement for fixed to fixed conversion (explained more fully under "FVE" in "Types of Conversions").

INST : Spacing control for off line print. Explained in Table 2.

INST 1 & INST 2 : Spacing control for on line print. Explained in Table 2.

Note : In the case of off line printing, the spacing control must be in the decrement. In the case of on line printing, the spacing control can be either in the decrement or address (but not both).

TABLE 2: SPACING CONTROL

OP	ADDR.	TAG	DEC.	SPACING
PTW	0	0	1	Restore Paper (off line)
PON	241	0	0	Restore Paper (on line)
PTW	0	0	0	Double Spacing (off line)
PON	0	0	244	Double Spacing (on line)
PTW	0	0	48	Single Spacing (off line)
PON	0	0	0	Single Spacing (on line)
PTW	0	0	16	Suppress Spacing (off line)
PON	0	0	245	Suppress Spacing (on line)
PTW	0	0	2	Half-Page skip (off line)
PON	242	0	0	Half-Page skip (on line)

All controls are effective before printing. If printing off line print position 1 should not be used for data as it is not reproduced, but is used for carriage control.

Routine does not echo check when printing on line.

Storage : LTR008 occupies 395 locations in memory and uses COMMON through COMMON + 51.

Speed : On line - 150 lines per minute for 72 columns, 75 lines per minute for more then 72. Off line-variable, but as an example, seven 8 place floating decimal numbers runs at 750 lines per minute.

TYPES OF CONVERSION

SVN : Floating binary to floating decimal, converts floating binary to floating decimal with "D" digits in the fraction and a 2 digit exponent (both are signed) Plus signs do not print. Programmer should allow D + 5 spaces for a floating point number.

Format: $\pm$ X. XX - - - X $\pm$ EE

SIX : Floating binary to fixed decimal
Converts floating binary to fixed decimal rounded "D" places to the right of the decimal point. Places to the left will vary with the size of the number (11 is maximum). If the absolute value of a number exceeds 34, 359, 738, 367; it will be printed as in "SVN". Lead zeroes to the left of the decimal point are omitted.

FVE : Fixed binary to fixed decimal
The binary number is converted as a fixed point fraction. The decimal point is considered to be at the extreme left of the number UNLESS the next calling sequence word contains a "PZE" in the operation and "T" in the address. ($1 \leq T \leq 35$).
If T is present, the first "T" bits will be converted as an integer and the rest of the word converted as a fraction. If T = 35 (an integer), then the conversion "FOR" should be used as only 1 calling sequence word is needed. If a T is given that shifts non-zero bits too for to the left, a blank is inserted for that number.

FOR : Binary integer to octal or decimal integer.
Binary number is converted as a binary integer to an octal or decimal integer with sign. If D = Zero, the number will be converted to a decimal integer with lead zeroes omitted. If D $\neq$ Zero, the number will be converted as a 12 digit signed octal integer with the following format:

$\pm$ XXXXXX XXXXXX

PTH : BCD to Hollerith
The "N" BCD words starting in "L" through "L + N - 1" (in that order) will be printed out as Hollerith text for headings, ID^{S}, etc.

The following are examples of how to use LTR008:

Ex. 1 : 1000 words of floating print data to be printed in sequence, 10 numbers per line, starting at "BLOCK" + 1. Off line only, single spacing.

SAP CODING

LABEL	OP	VARIABLE FIELD
	LXD	ONE, 1
START	TSX	LTR008, 4, 0
	SVN	BLOCK, 1, 5012
	SVN	BLOCK + 1, 1, 5024
	"	" + 2, 1, 5036
	"	" + 3, 1, 5048
	"	" + 4, 1, 5060
	"	" + 5, 1, 5072
	"	" + 6, 1, 5084
	"	" + 7, 1, 5096
	"	" + 8, 1, 5108
	"	" + 9, 1, 5120
	PTW	0, 0, 48
	TXH	* + 2, 1, - 1000
DONE	TRA	OUT
	TXI	START, 1, -10
ONE	PZE	0, 0, -1

Ex. 2 : You have 10 word blocks of data of various types and want to print 1 word from each block per line, until all ten words are finished. Also you want to print both on line and off line and desire double spacing.

The data in the various blocks requires the following conversion s:

LWA + 1	Conversion
FLOAT	Floating to floating
FLFIX	Floating to Fixed
INT	Integer to Octal integer
FIXED	Fixed to Fixed (Point 6 bits from left, not counting sign)
HOLL	BCD to hollerith (1 word per line)

This example was done and the results will be shown. However, the author wrote the Hollerith text backwards so it has to be read from bottom to top.

SAP CODING

LABEL	OP	VARIABLE FIELD
	LXD	TEN , 1
PRINT	TSX	LTR008, 4
	SVN	FLOAT, 1, 8017
	SIX	FLFIX, 1, 4034
	FOR	INT, 1, 8051
	FVE	FIXED, 1, 6068
	PZE	6, 0, 0
	PTH	HOLL, 1, 1078
	PTW	0, 0, 0
	PON	244, 0, 0
	TIX	PRINT, 1, 1
DONE	TRA	OUT
TEN	PZE	0, 0, 10

RESULTING PRINT

1.0000000	00	1.0000	-377777 777777	2.988281	MODES
2.5700000	09	2.0000	-275676 567656	5.612930	PRINT
-6.3499999-	04	2.2563	-043442 441440	7.000000	LTR008
0.0000000	00	-3.0000	001002 003004	9.140625	OF
6.2736999	18	-6.7275	123456 765432	10.607117	FORMS
-1.7562999	18	98.7620	111111 111111	12.571428	ALL
2.7769999	00	103.7100	223224 225226	14.797620	TESTIN
3.3333330-	01	.0000	234524 625724	16.750000	ARE
7.5999999	00	7.6736	-266605 555044	19.253914	LINES
1.0399999	01	97.5600	222222 333333	21.250000	THESE
FL. TO FL.		FL. TO FIX.	INT.	FIXED TO FIXED	HOLL.