32K FOCAL UPDATE - June 20, 1977

Significant changes have been made to the 32K Focal System. The current version of 32K FOCAL (version 77B) is described in Lick Observatory Technical Report #21, "32K FOCAL User's Guide" (Feb. '77). Effective June 20, 1977, version 77B of 32K FOCAL was replaced by version 77C*. This update describes the differences between the two versions, and the updates to be made to LOTR #21 so that it will correspond to version 77C.

The differences between version 77B and 77C fall into the following areas:

I. Modifications/corrections of resident functions

II. Corrections of overlay functions

III. Addition of new resident functions

IV. Addition of new overlay functions

V. Relocation of resident functions

VI. Correction of inherited bugs.

Robert Kibrick

* The version number of any 32K FOCAL System DECtape or floppy disk appears in the title line that is printed whenever a FOCAL WRITE command is executed.

ALPHABETICAL INDEX OF 32K FUNCTIONS

NAME	PAGE	DESCRIPTION
ADDQ*	U10	ADD ITEM TO END OF 5 ELEMENT QUEUE
ADDR*	34	OCTAL ADDRESS OF LAST WORD USED BY X ZAP
ADV	21.3	ADVANCE IBM TAPE ONE RECORD
AND	26.1	AND TWO 24-BIT NUMBERS TOGETHER
ASK	14	READ A FOCAL FLOATING PT. VARIABLE FROM DECTAPE
BAK	21.3	BACKSPACE IBM TAPE ONE RECORD
BIG	26.1	CONVERT 12-BIT SIGNED INTEGER TO 24 BITS
BRK	U6	BREAK OUT OF A FOCAL DO GROUP OR FOR LOOP
CADO	U7	DO A LINE OR GROUP IN ANOTHER PROGRAM
CALL	12	CALL AND OPTIONALLY START ANOTHER PROGRAM
CHAN	17	RETURN VALUE OF A SINGLE CHANNEL
CHEK	C3	RETURN SCANNER/PUSHBUTTON STATUS; UPDATE LEDS
CHOP*	*2	MOVES 120" TELESCOPE
CLOS	15	CLOSE OUT DATA-CHAINED DECTAPE BUFFER
CNT*	*1	START 3-PHOTOMULTIPLIER COUDE DEVICE
COMB*	*2	COMBINE ADJACENT 12-BIT WORDS IN A BUFFER
COMP	18	DRAW STRAIGHT LINES ON CRT OR CALCOMP
CORL*	33	CROSS-CORRELATION FUNCTION ON A BUFFER
COTY	22	PRINT TEXT FROM I.D. BUFFERS ON CRT OR TELETYPE
CPEN	11	RAISE/LOWER CALCOMP PEN
CRT	18	PLOT SCANS ON CRT OR CALCOMP
CT*	*2	READ AND ZERO SCANNER MEMORY COUNTING TIMERS
DEC*	34	CONVERT OCTAL TO DECIMAL
DIS	11	DISPLAY A DOT ON CRT
DIVD	25	DIVIDE BUFFER BY ANOTHER BUFFER OR A CONSTANT
DLTQ*	U10	RETURN AND DELETE ITEM FROM END OF QUEUE
DMUL	25	MULTIPLY BUFFER BY ANOTHER BUFFER OR A CONSTANT
DO	11	COMPUTED FOCAL DO COMMAND
DONE*	U10	RETURN STATUS OF QUICK IBM COMMANDS
DTIM*	29	APPLY DEADTIME CORRECTION TO A BUFFER
EDIT	17	SET VALUE OF A SINGLE CHANNEL
END	11	RETURN TO CALLING PROGRAM
EOF	21.1	WRITE AN END OF FILE MARK ON IBM TAPE
ERAS	17	SET ALL OR PART OF A BUFFER TO ZERO
EXCR	20	EXCHANGE/CIRCULAR SHIFT ROUTINE
FILE	13	SAVE A PROGRAM ON DECTAPE
FLIP*	33	REVERSE ORDERING OF CHANNELS IN A BUFFER
GO	11	COMPUTED FOCAL GO COMMAND
HDTS*	*2	RUN HIGH SPEED DATA TAKING SYSTEM
HI	26.1	RETURN HI-ORDER 12 BITS OF 24 BIT INTEGER
HUNT	21.2	HUNT FOR DOUBLE END OF FILE ON IBM TAPE

NOTES: * IN NAME FIELD INDICATES COMMAND IS IN AN OVERLAY

*1 SEE DOCUMENTATION FOR 3-PHOTOMULTIPLIER COUDE DEVICE

*2 SEE DOCUMENTATION FOR 32K DATA TAKING SYSTEMS

ALPHABETICAL INDEX OF 32K FUNCTIONS (CONT.)

NAME	PAGE	DESCRIPTION
IBM	21.2	RETURN IBM TAPE STATUS
IBME	21.2	ERASE 4 FEET OF IBM TAPE
IELD*	34	RETURNS FIELD OF LAST WORD USED BY X ZAP
IN	24	ADD 2 BUFFERS TOGETHER
IND*	30	FIRST MOMENT PEAK FINDER
INTP	C5	RETURN AND SET MASK OF P/S MONITOR
IPHO*	U13	DISPLAY VARIABLE DENSITY PIXELS ON CRT
IPUT	22	WRITE A 12 OR 24 BIT WORD TO I.D. BUFFER
ITAK	22	RETURN 12 OR 24 BIT WORD FROM I.D. BUFFER
JMP*	35	SIMULATES A PDP-8 JMP INSTRUCTION
JMS*	35	SIMULATES A PDP-8 JMS INSTRUCTION
LED	22	DISPLAY NUMBERS ON LEDS
LO	26.1	RETURN LO-ORDER 12 BITS OF A 24-BIT INTEGER
LOGB*	29	TAKE 10000*LOG BASE 10 OF A BUFFER
LOOK*	36	PLOT INTENSITY MAP ON CRT (FOR MAPPING TUBE)
LREC	21.3	RETURN RECORD LENGTH/LOG # OF LAST IBM RECORD
MCEN*	32	SET SWEEP CENTER
MCQ*	U11	START SCANNER COUNTING WITH EXTERNAL CLOCK PULSE
MEMC	23	START SCANNER COUNTING OR RETURN REMAINING TIME
MEME	23	ERASE ALL OF SCANNER MEMORY
MEMR	23	READ SCANNER MEMORY WITHOUT ERASING IT
MEMW	23	WRITE SCANNER MEMORY
MEMX*	32	LOAD/READ X SWEEP
MEMY*	32	LOAD/READ Y SWEEP
MGET	19	READ A 2048-CHANNEL SCAN AND I.D. FROM DECTAPE
MOVE	24	COPY FROM BUFFER TO BUFFER
MPX*	35	TRANSMIT/RECEIVE FROM/TO SERIAL MULTIPLEXER
MRQ*	U12	READ SCANNER MEMORY, ERASING WHAT IS READ
MSAV	19	WRITE A 2048 CHANNEL SCAN AND I.D. TO DECTAPE
NAME	12	READ AND ACTIVATE FOCAL COMMANDS FROM OVERLAY
OCT*	34	CONVERT FROM DECIMAL TO OCTAL
OUT	24	SUBTRACT A BUFFER FROM ANOTHER
PAK*	U12	PACK LO PART OF N CHANNELS IN N WORDS
PAUS	23	START/STOP SCANNER FROM COUNTING; NO TIME CHANGE
PEAK	17	FIND PEAK IN A BUFFER
PIXL*	U14	SET SCREEN PARAMETERS FOR X IPHO
POLY*	28	GENERATE POLYNOMIAL IN A BUFFER
POSN*	32	RETURN POSITION COORDINATES OF 120" TELESCOPE
PUT	14	WRITE A 12-BIT WORD TO DECTAPE
PUTN	17	LOAD A BUFFER WITH LINEAR DATA

NOTES: * IN NAME FIELD INDICATES COMMAND IS IN AN OVERLAY

*1 SEE DOCUMENTATION FOR 3-PHOTOMULTIPLIER COUDE DEVICE

*2 SEE DOCUMENTATION FOR 32K DATA TAKING SYSTEMS

ALPHABETICAL INDEX OF 32K FUNCTIONS (CONT.)

NOTES: * IN NAME FIELD INDICATES COMMAND IS IN AN OVERLAY
*1 SEE DOCUMENTATION FOR 3-PHOTOMULTIPLIER COUDE DEVICE
*2 SEE DOCUMENTATION FOR 32K DATA TAKING SYSTEMS

I. MODIFICATIONS/CORRECTIONS OF RESIDENT FUNCTIONS

A. Corrections

1. SET D = FMEMR (A,B,H) (See p. 23, LOTR #21)

   The "read only low 12 bits" option (H ≠ 0) now works correctly. In version 77B, use of this option would give correct results, but would actually run slower than if all 24 bits were read.

2. SET D = FCHEK (0) (See p. C-3, LOTR #21)

   Counting time display now counts down correctly to 0. In version 77B, counting time display would count down past 0 and leave a final display of 4095. Counting time now counts down in minutes/seconds; in version 77B countdown was in seconds only.

B. Modifications

1. X CALL (N,S,Q) (See pgs. 12-13, LOTR #21)

   a) Calls can no longer be nested to 10 levels as could be done in version 77B. Attempts to use the nesting option (Q ≠ 0) now generate a ?5422?00.00 error message. Nested calls can now be done using the new CALL/DO function (See X CADO command, pg. U7 of this update)

   Changes to text of LOTR #21

   1) p. 12, delete these 2 sentences under X CALL
      "If Q = 1, calls can be nested to 10 levels.
      Nesting cleared for Q = 0."

   2) p. B2, insert error code ?5422?00.00
      "5422 X CALL NESTING NO LONGER SUPPORTED. USE X CADO INSTEAD."

   3) p. B1, delete error code ?22.24.

b) Using the X CALL (or X END) command from inside a DO group or FOR loop has always been illegal (see LOTR #1, pg. Z9, paragraph C). However, such illegal uses were never detected in either 8K FOCAL or 32K FOCAL version 77B, and would cause unpredictable results. In version 77C, such illegal uses of X CALL or X END are detected, and a ?5440?00.00 error message is printed.

Changes to text of LOTR #21

1) p. B2, insert error code ?5440?00.00:

"5440 X CALL or X END FROM WITHIN A DO GROUP OR FOR LOOP"

c) In both 8K and 32K FOCAL version 77B, one could easily wipe out many minutes worth of keypunching FOCAL text if one accidentally started a new (or modified) program that contained an X CALL (or X END) command, before one had filed the new program on DECtape or floppy disk. In version 77C:

If FOCAL's text area contains new or modified text that has not yet been filed, then execution of X CALL (or X END) will cause FOCAL to type

REALLY?

on the teletype, and then to wait for a reply. Any reply other than 'Y' will cause the X CALL (or X END) command to be supressed, and the contents of the text area to be preserved. A reply of 'Y' will cause the X CALL (or X END) to proceed, and the contents of the text area to be lost.

Changes to text of LOTR #21:

1) Cut off bottom half of this page; paste on pg. 12, LOTR #21 over X CALL (N,S,Q)

2. X END (0) (See pg. 11, LOTR #21)

Modifications b) and c) as described under X CALL above apply also to X END.

3. X FILE (N) (See pg. 13, LOTR #21)

The contents of FOCAL's text area is written to DECtape or floppy disk as program N; then the program N just written out is read back into FOCAL's text area, as if an X CALL (N) had been used. In version 77B, the text area is written out but not read back. The user should not notice any real difference between the two versions, except that the version 77C X FILE command will take slightly longer.

Place over bottom half of page 12, LOTR #21

Cut here

X CALL(N,S)

(For greater versatility in calling, see the X CADO command, pg. U7)

- Call Program N from Dectape. If S>0, starts program N at subroutine S. (Code S*128 + L to start program N at line L of subroutine S). Use X END(0) to return to the line following the X CALL command in the calling program.

Important Notes:

A) It is illegal to use either X CALL or X END from inside a DO group or FOR loop.

B) If FOCAL's text area contains new or modified text that has not yet been filed (See X FILE command below), then execution of X CALL or X END will cause FOCAL to type "REALLY?" on the teletype, and then to wait for a reply. Any reply other than "Y" will cause the X CALL or X END command to be supressed, and the contents of the text area to be preserved. A reply of "Y" will cause the command to proceed and the contents of the text area to be lost.

C) If X CALL is used to address a program area on the Dectape that does not contain a FOCAL program, the following action is taken:

II. CORRECTIONS TO OVERLAY FUNCTIONS

A. Sweep, Time, Position Overlay (See p. 32, LOTR #21)

The commands in this overlay have been changed so that they work correctly in either NAME buffer. This new overlay (dated 5/16/77) will not work if used with version 77B FOCAL. The old Sweep, Time, Position overlay (dated 2/14/77) will not work with version 77C FOCAL. The old Sweep, Time, Position overlay worked correctly if used in NAME buffer 1, but caused occasional MUX errors if used in NAME buffer 2. When copying version 77C to your FOCAL tape, be sure to copy an updated version of this overlay.

Changes to text of LOTR #21

p. B3: Change error code 6171 to 6165, 6174 to 6170.

B. DebuggingUtility Overlay (See p. 34, LOTR #21)

1. The X MPX command in this overlay has been changed so that it works correctly in either NAME buffer. (The X MPX command had the same problem as the commands in the Sweep, Time, Position overlay, and the same considerations apply. The new Debugging Utility Overlay is dated 5/4/77; the old one was dated 2/23/77).
2. The X MPX command no longer hangs up the spectrograph control panel at the 120" readout room; the previous version did.

III. ADDITION OF NEW RESIDENT FUNCTIONS

A. X BRK(0)

One of the problems with FOCAL is that:

> "If a GOTO or IF command that is inside a DO group transfers control to a line outside the DO group, that line is executed and control then returns to the command following the DO." (see DEC. FOCAL manual)

A similar problem exists with GOTO or IF statements that are used inside of FOR loops. This feature of FOCAL makes it awkward to bail out of a DO group or FOR loop, as is illustrated by the following example program.

```
*01.10 FOR J=1,3;DO 2
*01.20 TYPE !"L 1.2"
*01.30 TYPE !"L 1.3";QUIT

*02.10 TYPE %2,J;DO 3
*02.20 TYPE !"L 2.2"

*03.10 IF (J-2)3.3,3.2,3.3
*03.20 TYPE !"L 3.2"
*03.25 GO 1.2
*03.30 TYPE !"L 3.3"
*GO
  1
L 3.3
L 2.2    2
L 3.2
L 1.2
L 2.2    3
L 3.3
L 2.2
L 1.2
L 1.3*
```

The X BRK command solves this problem by purging FOCAL's pushdown list, which is where FOCAL stores information relating to the nesting of DO groups and the control of FOR loops.

The effect of the X BRK command is to break one out of the inside of any DO groups or FOR loops, as illustrated by the example below. Note that any commands on the same line following an X BRK command are ignored, and execution continues with the next sequential line.

```
*01.10 FOR J=1,3;DO 2
*01.20 TYPE !"L 1.2"
*01.30 TYPE !"L 1.3";QUIT

*02.10 TYPE %2,J;DO 3
*02.20 TYPE !"L 2.2"

*03.10 IF (J-2)3.3,3.2,3.3
*03.20 TYPE !"L 3.2";X BRK(0);TYPE "ABC"
*03.25 TYPE !"L 3.25";GO 1.2
*03.30 TYPE !"L 3.3"
*GO
  1
L 3.3
L 2.2    2
L 3.2
L 3.25
L 1.2
L 1.3*
```

Program

Group

Step

B. X CADO (P,G,S)

This command functions exactly like a FOCAL DO command, except it allows one to "DO" text that is in another FOCAL program. Program P is called, then FOCAL 'DO'es group G, step S, as if it were part of the calling program. If S = 0, all of group G is "done." If G = S = 0, all of program P is "done". When the specified text has been "done," control returns to the statement following X CADO in the calling program. The X CADO command can be used anywhere a FOCAL DO command can be used, and X CADO's can be nested. The degree of nesting allowed depends upon space available in the pushdown list; X CADO requires 1 more word of pushdown list space than does a regular FOCAL DO statement.

Note - If FOCAL's text area contains new or modified text that has not yet been filed, then execution of an X CADO command will result in a ?3336?00.00 error.

Changes to text of LOTR #21

p. B2, insert error code 3336

"3336 X CADO EXECUTED FROM MODIFIED TEXT BUFFER. USE X FILE FIRST."

IV. ADDITION OF NEW OVERLAY FUNCTIONS

A. Quick Data Overlay

```
                    ______Buffer (BUFFER SPANNING NOT ALLOWED)
                   /
                  /   ____Starting channel
                 /   /    (-1 to read 31 words into I.D. area A)
                /   /
               /   /   ___Number of channels (ignored if C = -1)
              /   /   /
             /   /   /   _Wait option
            /   /   /   /
SET D = FRDQ (A,C,N,W)
```

Starts reading N channels from IBM tape to buffer A. Reads to end of buffer if N = 0. Works like READ command except:

1) Control returns to FOCAL as soon as the IBM tape starts moving. Data transfer can go on in parallel with computing. (One must be careful not to use the portion of buffer A into which the data is being transferred until the transfer is completed! Use DONE* command to check for completion.)

2) D returns the status of any previous IBM read or write operation, not the status of this command. See DONE command for description of status value returned.

3) If the tape drive is still busy processing a previous RDQ or WRQ command when the current command is executed, three options are available

W = 0	A ?6062?00.00 error is printed and the FOCAL program is terminated
W < 0	The current command is ignored and control returns immediately to the FOCAL program. D is set -1

*The DONE command is described on page U10

W > 0 The current command waits for the previous IBM tape operation to complete. Then the current command is carried out.

4) If the IBM tape is at END OF TAPE (EOT), the tape drive will appear "busy". The same actions are possible as described in 3) above, except that in the case of W < 0, D will be set to -2.

Buffer (BUFFER SPANNING NOT ALLOWED)
Starting channel
(-1 to write 31 words from I.D. area A)
Number of channels (ignored if C = -1)
Wait Option

SET D = FWRQ (A,C,N,W)

Starts writing N channels to IBM tape from buffer A, starting in channel C. Writes to end of buffer if N = 0. Works like WRIT command except for the same 4 differences described under the RDQ command above. Also, if a WRQ command is directed to a write protected IBM tape, a ?6273?00.00 error is given.

Notes: The RDQ and WRQ commands can be used interchangeably with READ and WRIT, and are fully compatible with all other resident IBM tape commands, with the exception that one should allow approximately a 20 millisec delay after using X BAK before using X RDQ or X WRQ.

Changes to text of LOTR #21:

p. B3 - add error codes 6062 and 6273

"6062 RDQ or WRQ USING 'NO WAIT' OPTION FOUND IBM TAPE BUSY
6273 WRQ FOUND IBM TAPE WITHOUT WRITE RING"

SET D = FDONE(0)

Returns status of last READ or WRITE operation.

D = 0 if last RDQ or WRQ operation completed successfully

D < 0 if last RDQ or WRQ operation has not yet completed

D > 0 if tape error on last RDQ or WRQ, or if last record was a file mark.

The low 12 bits of D contain the remaining word count

The high 12 bits of D contain:

BIT 11 is 1 if last record was file mark (4096 bit)

BIT 10 is 1 is parity error on last record (8192 bit)

BITS 1-9 unused

BIT 0 is 0

Number to be queued. (0 ≤ A ≤ 4095)

SET D = FADDQ (A)

Appends A to the end of a 5 element queue.*

D = 0 if successful.

If queue is already full, D is set < 0 and queue remains unchanged.

SET D = FDLTQ (0)

Sets D to the value of the front element of a 5 element queue*, then removes this element from the queue.

Sets D < 0 if queue already empty.

Notes: 1. ADDQ and DLTQ are useful for maintaining a queue of buffers to be read or written using the RDQ and WRQ commands. By queueing data, one can handle instantaneous data rates

* A queue works on the basis of first in, first out.

that exceed that average data transfer rate of the IBM tape drive. Such situations occur in high speed data taking applications.

2. The storage area used for the queue is contained within the overlay. The queue is initially empty when the overlay is brought in with X NAME. No X NAME commands should address the buffer containing this overlay in between saving queue elements with ADDQ and retrieving them with DLTQ.

Number of scanner cycles to count

Switch mask

SET D = FMCQ (N, MASK)

If scanner is already counting or if N < 0, returns remaining counting time, and leaves scanner unchanged. If scanner is not counting and N $\geq$ 0, then

1) Waits for a pulse to be detected on the switch(es) in group 3 selected by MASK. (External clock pulses can be brought in across switches 3.8, 3.9, and 3.10. Pulse width should be at least 30 μsec)

2) Waits to synchronize with scanner memory (max. 4.4 millisec). D returns time spent waiting to synchronize with memory in units of 13.25 μsec.

3) Once synchronized, sets counting time of N scanner cycles and starts counting. 1 scanner cycle is ≈ 4.4 μsec. (N = 0 gives a counting time of ≈10 hours).

```
              Buffer for 1st 2048 channels (right slit)
              Buffer for 2nd 2048 channels (left slit)
X MRQ (A,B)
```

Wait for current scan to complete, then read scanner memory to core.

First 2048 channels read to buffer A, (starting in channel 0) and erased from scanner memory as they are read (Not read or erased if A = -1). Similarly for 2nd 2048 channels and buffer B. Reads all 24 bits normally.

To read only the low 12 bits of each scanner channel, and to pack 2 of these 12 bit scanner channels per 24 bit buffer channel, add 4096 to the buffer number.

B. Data Compaction Overlay

```
                        Output Buffer
                        Input Buffer
                        Starting word in A (0 ≤ WDA ≤ 4095)
                        Starting word in B (0 ≤ WDB ≤ 4095)
                        Number of channels in B
X PAK (A,B,WDA,WDB,N)   Packs N channels into N words
```

Each channel in B consists of 2 12-bit words. Starting at WDB in buffer B and processing N such channels, the low order word of each channel is stored into consecutive words in buffer A, starting at WDA. The high order words of each channel in B are ignored.

```
Output Buffer
Input Buffer
Starting word in A
Starting word in B
Number of channels in A
X UNPK (A,B,WDA,WDB,N)
```

Reverse of PAK.

Starting at WDB in buffer B and processing N words, appends a high order word of 0 to each word and stores into successive channels in buffer A, starting at WDA.

Changes to text of LOTR #21:

p. B3 - add error code 6255

"6255 starting word(s) <0 or >4095 in PAK or UNPK"

C. Isophote Display Overlay (Insert as pg. 38 of LOTR #21)

```
Buffer
Starting channel
Number of channels
SET D = FIPHO (A,C,N)
```

Takes data values in the range 0, 1, or 2 and displays them as PIXELS of 3 different densities on CRT.

0 displays as a blank PIXEL.

1 displays as a lined PIXEL.

2 displays as a solid PIXEL.

Display consumes 1 channel of data per PIXEL, sweeping the CRT in a raster pattern. PIXEL size and CRT screen parameters are set using the X PIXL command. D is set to number of PIXELS displayed by this command.

<u>Changes to text of LOTR #21</u>:

p. B3 insert error code 6344.

"6344 screen full in XIPHO or XPIXL not called first."

PIXEL size (CRT dots on a side)

PIXELS per line

Lines per screen

X origin for picture on CRT

Y origin for picture on CRT

0 to raster from bottom to top
1 to raster from top to bottom

X PIXL (PS,PL,LS,XØ,YØ,YSWEP)

Defines PIXEL size and screen parameters used by IPHO command.

The input arguments must pass the following tests:

1) PS must be 2 or a multiple of 4

2) PL, LS, XØ, YØ must all be ≥ 0

3) PL*PS+XØ must be ≤ 1024

4) LS*PS+YØ must be ≤ 1024

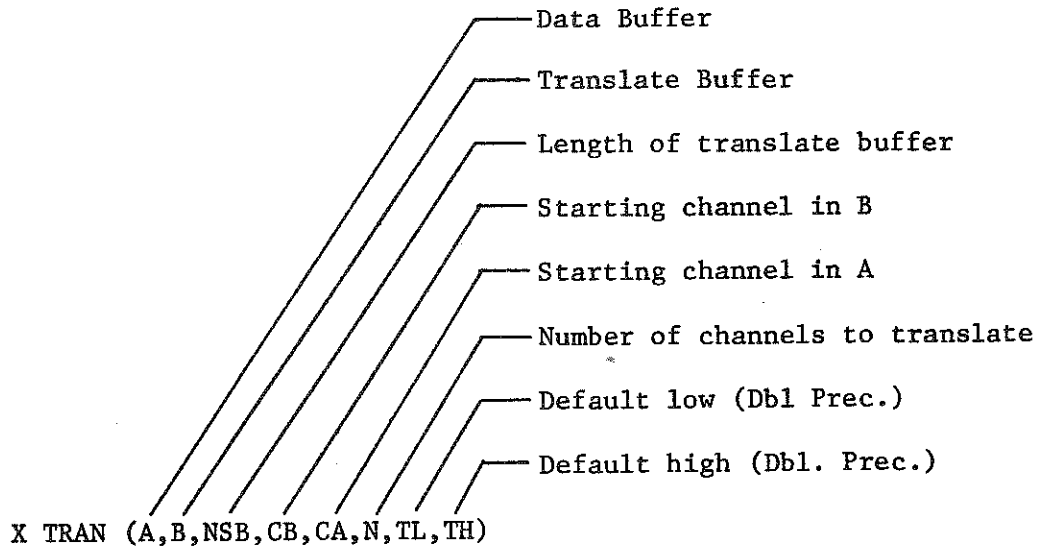
<u>Changes to LOTR #21</u>:

p. B3 insert error codes 6301, 6302, 6303

"6301 Bad PIXEL size in X PIXL

6302 Bad combination of PL, PS, XØ in X PIXL

6303 Bad combination of LS, PS, YØ in X PIXL"



Data Buffer
Translate Buffer
Length of translate buffer
Starting channel in B
Starting channel in A
Number of channels to translate
Default low (Dbl Prec.)
Default high (Dbl. Prec.)

X TRAN (A,B,NSB,CB,CA,N,TL,TH)

This command allows one to apply an arbitrarily defined function to a buffer of data. The function values are assumed to have been previously computed and stored into the translate buffer. The function value stored into a given channel of the translate buffer is simply the value obtained by applying the function to the number of the channel. Specifically, the TRAN command operates as follows:

> Translates buffer A, starting at channel CA and translating N channels.
>
> Buffer B contains the translate table, starting in channel CB and running NSB channels long. The following operation is performed on each indicated channel:
>
> > The value of the channel in buffer A is used as an index into the translate table in buffer B. If this index value is >0 and <NSB, the value of the indexed channel in buffer B is used to replace the value of the indexing channel in buffer A. If the value of the indexing channel in buffer A is <0, then it is replaced with the double precision value TL. If the value is > = NSB, then it is replaced with the double precision value TH.

V. RELOCATION OF RESIDENT FUNCTIONS

The starting addresses of the following resident functions have been moved. In some cases, this has caused a corresponding change in the error messages produced by these functions. Appendices A1-A3, and B2 should be updated accordingly.

NAME	OLD ADDR.	NEW ADDR.	OLD ERROR	NEW ERROR
ADV	5257	5256	--	--
AND	3730	4770	--	--
CHEK	3722	3732	--	--
END	5537	5550	--	--
FILE	5473	5501	--	--
PAUS	3630	3627	--	--
RWND	5121	5120	--	--
VAR	5260	5255	5273	5270
			5277	5274

VI. CORRECTION OF INHERITED BUGS

A. If one used a DO statement to invoke a single line that contained an error, FOCAL diagnosed the error as occuring on the line containing the DO, and not on the line in which the error actually occurred.

Example

```
*E   A
*1.2   DO 2.1
*1.3   C
*2.1   TYPE A/Ø
*G
 ?28.73 @ 1.20  error should be diagnosed on line 2.1
```

B. The Lick FOCAL functions X DO and X GO did not behave the same as the standard FOCAL DO and GO commands. In particular, these commands had the same effect as the new X BRK command. That is, X DO and X GO would purge FOCAL's pushdown list and cause the program to break out of any DO groups or FOR loops in which it was nested. Also, any text following X DO on the same line was ignored. In 32K FOCAL version 77C, X DO and X GO now work exactly like FOCAL's DO and GO commands.

C. If the X EOF(0) command was immediately followed by an X BAK(1,1) command, one would fail to backspace behind the file mark. Now fixed.

D. If an IBM tape were backspaced to beginning of tape (BOT) using the X BAK command, or rewound using the X RWND command, the IBM tape drive was erroneously left in the "start" state. If another IBM tape command immediately followed, it could possibly start too soon and cause I/O errors. This problem fixed in version 77C.