APPENDIX

The TRANSCODE system of routines consists of several parts:

I One section, involving decimal-binary conversion and arithmetic operations on floating numbers, could actually be used independently of the remainder of the TRANSCODE system. This comprises the following routines, and full details are given under these titles elsewhere in the FERUT Library:- TC: DECINPUT, TC: ARITHMETIC and TC: DECOUT. However, these routines are automatically utilized in their TRANSCODE setting.

II Another section consists of a pair of TC: INPUT routines, which interpret the tape controls, transferring the instructions and numbers, with some modification, to predetermined locations within the machine. CNST and NUMB controls automatically employ TC: DECINPUT.

III A large section of the TRANSCODE system consits of routines which are called into play by ENTR. They translate the instructions as stored by TC: INPUT into the real machine code, and make all other necessary preparations for the final execution of the programme under the automatic control of the machine. It is actually in the translation of the QUIT instruction that all anticipated addresses in control transfers are inserted, and all other final preparations are made for switching over from the translating to the executing stage of the work. A ~~88~~/L stop separates the two stages.

The FNTN input facility resorts to INPUT/T as a subroutine, and it is assumed that this is stored on tracks 4 and 5. All other necessary routines are peculiar to TRANSCODE.

The reader may be interested to know that, in the final analysis, the standard Toronto Scheme for changing routines and allocating data is employed. During the execution of the calculation, the electronic store is utilized as follows:-

S0 and S_1 contain the current problem routines, and these assume that the FLOATER routines of TC: ARITHMETIC are on S_4 and S_5. PERM/T is available on S_2, and column I of S_3 is used for the RCS/T link-list as well as for working space. The remainder of S_3 is used for the Z-type numerical working space, S6 for the X-type, and S_7 for the Y-type pages of numerical data.

Magnetic Storage

The TRANSCODE routines occupy 17 tracks. During its operation, TRANSCODE also requires 4 sets of consecutive working tracks. The working tracks are designated as follows:

t_1	as the	first of	n_1	consecutive tracks
t_2	"	"	n_2	"
t_3	"	"	n_3	"
t_4	"	"	n_4	"

where (t_1) L, (t_1+1)L, --- (t_1+n_1-1)L store the modified TC instructions to be translated,

t_1R stores the CNST's,

(t_1+1)R stores the untranslated addresses,

(t_1+2)R stores the final INDEX page,

(t_1+3)R, (t_1+4)R, -- (t_1+n_1-1)R store the TC directory of instruction numbers,

t_2, (t_2+1), --- (t_2+n_2-1) store the translated inst'ns. and also the FNTN101, FNTN102, ...

(t_3)L, (t_3+1)L, --- $(t_3+\frac{n_3}{2}-1)$L store DRUM consignments 1, 2, 3 --- $\frac{n_3}{2}$,

(t_3)R, (t_3+1)R, --- $(t_3+\frac{n_3}{2}-1)$R store DRUM consignments $(\frac{n_3}{2}+1)$, --- n_3.

t_4 stores FNTN 000, which is a TC routine required if KOPY is to be used.

t_4+1, (t_4+2), ---(t_4+n_4-1) store FNTN consignments 1, 2, 3 ---(n_4-1).

Normal Track Assignments

Normally, all the TC routines except FNTN 000 will be assumed already stored on tracks TA to ØA inclusive. A Toronto directory tape is preserved by the librarian for this purpose. If KOPY is required, the FNTN 000 tape should also be read in, and it will automatically go to track t_4.

Also, it will normally be assumed that

t_1 = T/ , $n_1 = 16$,

t_2 = /E , $n_2 = 16$,

t_3 = TE , $n_3 = 16$,

t_4 = T@ , $n_4 = 16$.

These working tracks are all specified in a page of INDICES, which is treated like one of the TC routines. A copy of this INDEX page is given, so that any changes in t_1, t_2, t_3, t_4 or n_3 may be made as required.

INDICES for TRANSCODE

Available during TC: INPUT on S5,
" " TC: Translation on S2.

R :		J S
////	/	↓
////	E	
////	@	$n_3 \div 2$
K///	A	$R //
$\frac{1}{2}$A//	:	////
/@//	S	@J //
(t_1)/N	I	TJ //
(t_2)R/	U	(t_4)@/
(t_1+3)DI	$\frac{1}{2}$	AJ //
(t_1+3)EI	D	EJ //
(t_3-1)//	R	AR //
(t_1+1)DC	J	LJ //
///$	N	YJ //
(t_1+1)EC	F	ER //
////	C	ZJ //
////	K	MR //
↑	T	GJ //
	Z	KJ //
	L	OJ //
Working	W	$D //
	H	CR //
	Y	(t_1+2)D:
Space	P	@D //
	Q	UJ //
	O	CJ //
	B	WJ //
	G	$$$$
	"	E///
	M	/J //
	X	(t_1)EC
	V	QR //
	$	(t_2)@/

UQN

Tape Contents of FNTN Tape ≡ KOPY output

```
Spaces  FNTN 00n  Spaces
K/½/
        contents of first column as in Toronto system
K/D/
          "        second    "      "      "
K/R/
          "        third     "      "      "
K/J/
          "        fourth    "      "      "
```

KLU@ (check sum for cols. 1 and 2)C/½ KLU@(Check sum for cols. 3 and 4) C/R EQS

(In write-taping,
RRRR is also punched at the end of the last FNTN, and the FNTN's are automatically numbered as 101, 102,103 ---.)

Real Code Equivalents of TC addresses

XOn ≡ XF-3(n-1)
YOn ≡ XK-3(n-1)
ZOn ≡ XU-3(n-1)
COn ≡ XK-3(n-1) and this is marked for special consideration during translation.

NUMB and CNST read in numbers, consecutively, to S4. Storage is backwards, beginning with $[XD]_0^{39}$ for amplitude, and $[\$D]_0^{19}$ for 2×(exponent). Translation from floating decimal to floating binary takes place.

INST, with certain modifications, are read in to consecutive tracks via S_4, 4 short lines per instruction.

A special transcode directory is built up in S_3 during translation, and stored on consecutive $\frac{1}{2}$ -tracks. An entry in the form

$$[(\text{track})(\text{machine address of first real instruction})]_0^{19}$$

is made in this directory for each TC instruction. QUIT avails itself of this directory for supplying the untranslated addresses occurring in forward TRNS instructions.

The last track to be used for the machine instructions can quickly be determined by inspecting the last entry made on S3 when the $$/L stop is encountered. This should not exceed the last track assigned for the purpose.

OVER $\alpha.B_\alpha$ 0.0 $\gamma.B_\alpha$		(α)	$(T+B_\alpha)$	$\frac{1}{2}$	
		(γ)	$(T+B_\gamma)$	A	
		$(\alpha+1)$	$(T+B_\alpha)$	$\frac{1}{2}$	
		$(\gamma+1)$	$(T+B_\gamma)$	A	
KOMP $\alpha.B_\alpha$ $\beta.B_\beta$ $\gamma.B_\gamma$		$(\alpha+1)$	$(T+B_\alpha)$	$\frac{1}{2}$	
		E $\frac{1}{2}$	T	A	
		(α)	$(T+B_\alpha)$	$\frac{1}{2}$	
		E:	/	M	
		(α)	$(T+B_\alpha)$	F	
		$/\frac{1}{2}$	TA		
		$(\beta+1)$	$(T+B_\beta)$	$\frac{1}{2}$	
		: $\frac{1}{2}$	T	A	
		(β)	$(T+B_\beta)$	F	
	r-1	A $\frac{1}{2}$	T	A	
	r	(r)	Q	O	
	r+1	HJ	/	P	
		U$\frac{1}{2}$	T $\frac{1}{2}$		
		$(\gamma+1)$	$(T+B_\gamma)$	A	
		I $\frac{1}{2}$	T $\frac{1}{2}$		
		(γ)	$(T+B_\gamma)$	A	
$\left\{\begin{matrix}\text{READ}\\ \text{WRTE}\end{matrix}\right\}$ $\alpha.0$ 0.B $\gamma.0$		(ab)	(T+B)	$\frac{1}{2}$	where $[ab]_0^{39}$ is set = (transf. and check) during translation and $[JJ]_0^{19}$ = $\mathrm{M}\$\$\$$ in FLOATER/B.
		P I	(T+B)	B	
		Q I	(T+B)	B	
		P I	T	C	
		P I	T	A	
		P I	/	:	
		Q I	/	:	
		/ /	/	V	
		J J	/	Q	
HALT	///G				
VOID	$//\mathrm{T}\$$				

<u>Real Code Equivalents of TC Instructions</u>

ADDN $\alpha.B_\alpha$ $\beta.B_\beta$ $\gamma.B_\gamma$	$(\alpha+1)$	$(T+B_\alpha)$	$\frac{1}{2}$	
	$E\frac{1}{2}$	T	A	
	(α)	$(T+B_\alpha)$	$\frac{1}{2}$	
	$/\frac{1}{2}$	T	A	
	$(\beta+1)$	$(T+B_\beta)$	$\frac{1}{2}$	
	$:\frac{1}{2}$	T	A	
Real address = r-2	(β)	$(T+B_\beta)$	$\frac{1}{2}$	
r-1	$A\frac{1}{2}$	T	A	
r	(r)	Q	O	
	HJ	/	P	
	$U\frac{1}{2}$	T	$\frac{1}{2}$	
	$(\gamma+1)$	$(T+B_\gamma)$	A	
	$I\frac{1}{2}$	T	$\frac{1}{2}$	
	(γ)	$(T+B_\gamma)$	A	
SUBT Same as ADDN but with $[r-2] = (\beta)(T+B_\beta)N$.				
MULT " " " $[r+1] = Y\ J\ /\ P$.				
DIVN " " " $[r+1] = P\ J\ /\ P$.				
$\frac{1}{2}$QRT $\alpha.B_\alpha$ 0.0 $\gamma.B_\gamma$	$(\alpha+1)$	$(T+B_\alpha)$	$\frac{1}{2}$	
	$E\frac{1}{2}$	T	A	
	(α)	$(T+B_\alpha)$	$\frac{1}{2}$	
	$/\frac{1}{2}$	T	A	
r	(r)	Q	O	
	WJ	/	P	
	$U\frac{1}{2}$	T	$\frac{1}{2}$	
	$(\gamma+1)$	$(T+B_\gamma)$	A	
	$I\frac{1}{2}$	T	$\frac{1}{2}$	
	(γ)	$(T+B_\gamma)$	A	
ZERO $\alpha.B$ 0.0 0.0	N D	T	/	where $[ND]_o^{39} = //// ///\frac{1}{2}$
	(α)	(T+B)	A	$[CD]_o^{19} = ///0$ ln
	FD	T	/	
	$(\alpha+1)$	(T+B)	A	FLOATER/A .

COKE	r	(r+3)/:	
	r+1	(r+4)/:	
	r+2	A: /Q	
	r+3	ZN //	false line for FLOATER/A
	r+4	LN //	false line for FLOATER/B
	r+5	(r+7)/:	
	r+6	E: /Q	
	r+7	S//:	

	Backwards, but to same two pages		Backwards to another two pages, or forwards	
TRNS α.0 0.0 0.0	r	//T$	r	//T$
	r+1	E:/Q	r+1	E:/Q
	r+2	S:/Q	r+2	S:/Q
	r+3	//T$	r+3	(r+3)$0
	r+4	(r+5)/P	r+4	GS/P
	r+5	($\bar{\alpha}$) //	r+5	(Link corresponding to α)
	r+6	(same track) @/	r+6	
TRNS α.0 0.B 0.0	r	DJ(T+B)G	r	DJ(T+B)G
	r+1	E:/0	r+1	E:/0
	r+2	S:/Q	r+2	S:/Q
	r+3	//T$	r+3	(r+3)$0
	r+4	(r+5) /P	r+4	GS/P
	r+5	($\bar{\alpha}$)//	r+5	(Link corresponding to α)
	r+6	(same track) @/	r+6	
TRNS α.0 0.0 γ.0	r	(γ)T$\frac{1}{2}$	r	(γ)T$\frac{1}{2}$
	r+1	E:/M	r+1	E:/M
	r+2	S:/Q	r+2	S:/Q
	r+3	//T$	r+3	(r+3)$0
	r+4	(r+5)/P	r+4	GS/P
	r+5	($\bar{\alpha}$)//	r+5	(Link corresponding to α)
	r+6	(same track)@/	r+6	

where [DJ] = A/// in FLOATER/B and $\bar{\alpha}$ = control entry corresponding to α. A test is made that r ≠ GE. If r = GE, //T$ is made to precede the instructions, so as not to split the link.

LOOP α.0 0.B 0.0 (ab)(T+B)0 where $[ab]_0^{19} = 3\ (\alpha-1)$

Note 1. LOOP essentially sets B-line B to count α times, as well as to modify X,Y or Z addresses. The B-conditional TRNS instruction mentions B-line B by subtracting 3 from it, thus enabling it, in the next cycle, to pick up the next floating number 3 machine lines back in an electronic page.

Note 2. The two instructions

```
INCB  000.B  003.0  000.0
TRNS  00α.0  000.B  000.0
```

have the effect of a normal B-conditional control transfer.

Note 3. Arithmetical operations may be carried out on integers by using the various B-line instructions, but care must be taken not to confuse these with the arithmetic instructions, proper (ADDN, KOMP etc.) which operate on 60-bit floating binary numbers.

BSET 0.B β.0 0.0 0.B 0.0 γ.0		(ab)(T+B)0 (γ+2)(T+B)0	where $[ab]_0^{19} = \beta$
INCB 0.B β.0 0.0		(ab) (T+B)G	where $[ab]_0^{19} = -\beta$
JOTB 0.B 0.0 γ.0		(γ+2)(T+B)B	
NEGB 0.B β.0 0.0 0.B 0.0 γ.0		(ab) (T+B)G (γ+2)(T+B)G	where $[ab]_0^{19} = \beta$
PRNT α.B_α β.0 γ.0	r	(r+11)T/	
	r+1	PI TA	
	r+2	(r+10)LO	
	r+3	(r+9)WO	
	r+4	(r+4)@O	
	r+5	GS/P	
	r+6	XA//	
	r+7	TN//	false line for TC:DECOUT
	r+8	S:/Q	
	r+9	(α-1) //	
	r+10	(γ)//	
	r+11	- {E//(β-1}	
	r+12	(B_α-1)//	
	r+13	UJ /J	where $[UJ]_0^{39}$ = ...W
	r+14	ØD/:	and $[ØD]_0^{19}$ = ..T.
FNTN α.0 β.B_β γ.B_γ		(β+1)(T+B_β)½	Note: The FNTN routine must operate in S0 and S1, with control entry Ø Ø, and leave the answer in $[/\frac{1}{2}]_0^{39}$, $[@\frac{1}{2}]_0^{19}$.
		E ½ T A	
		(β) (T+B_β)½	
	r-1	/ ½ T A	
	r	(r) Q O	
	r+1	G S / P	
		Ø Ø / /	
		(track)@/	
		E ½ T ½	
		(γ+1) (T+B_γ) A	
		/ ½ T ½	
		(γ) (T+B_γ) A	

Note In the TRNS, PRNT and FNTN instructions a linking sequence to the RCS/T occurs, and it is important that the first short line of the link should not be placed in ØE. During translation, a test is made for this contingency, and an extra dummy instruction inserted prior to the machine orders quoted above, should it arise.