ICT 1301
TAS
Thirteen Hundred
Assembly System

Scanned 2010
at the ict1301
Resurrection Project
ict1301.co.uk

Chapter 6

HOW TO USE TAS

THE SOURCE PROGRAM 6.1

Writing the Source Program 6.1.1

Reduction of the defined problem into TAS coding will generally be introduced at the same point as a machine-coded program would be produced; in other words, after flow-charting. However, it may be possible to translate the systems flow-charts directly into TAS language without the need to produce simpler steps.

It is always necessary to define all Tables and Input/Output Formats before writing the procedure section. This is because all Field names involved in input and output, which are to be mentioned in the calculation etc., will have a known position relative to each other. Although the positions are not known absolutely until after compilation, a form is provided which may be filled in by the programmer as he creates each new name (see Appendix H). As each new Field name is mentioned by the programmer he enters it on a form, together with the number of I.A.S. words occupied by the Field named. In this way he knows the order in which the Fields are stored within I.A.S., for modification purposes etc., and can also control the number of I.A.S. locations used by Fields.

Punching and Assembling the Source Program 6.1.2

The program is punched on the cards provided. These all have exactly the same format (see Appendix H) but each different type of card, i.e. Table card, Program card etc., carries a different designation:

Table cards	- 01
Block Heading cards	- 02
Input/Output Format cards	- 03
Subroutine Heading cards	- 04
Program Instruction cards	- 05
File Description cards	- 06
Record Format cards	- 07
End of Program card	- 09

The TAS Program Sheet has provision made for specifying the designation, which always occurs in columns 1 and 2 of a card. In the case of Table and File Description sheets, this is fixed and is preprinted.

A sequence check may be carried out on a six digit sequence number and a six digit program number. If this check is not inhibited, the program number must be identical in all cards relating to the same program, and the card sequence numbers must ascend in increments of 1 throughout the entire pack, the first card of the pack being numbered 1.

The source program is then assembled in the following order:

1. Table cards
2. File Description cards
3. Record Format cards
4. Input Card Format cards
5. Output Card Format cards
6. Procedure cards (Block Heading, Subroutine Heading, and Program Instruction cards). Global Subroutines must come at the end of this group.
7. End of Program card (blank except for designation).

COMPILATION 6.2

Procedure 6.2.1

The TAS compiler is held on cards punched in 'fast read' form. It uses 12,000 words of drum storage. It is fed into the machine, and, once the E card stop has been reached, the source programs are read in as data to the compiler.

As each source program is read in, it is listed, and the appropriate instructions compiled. At the end of the Input Formats, Output Formats, and each program block, object program cards are punched and, after the last block, a final special group of cards is produced.

Final Stop 110999 is displayed in CR3 on the successful completion of a compilation. The cards in the punch stacker can then be removed and assembled into an object program. Before proceeding with the next compilation, the punch must be re-primed; compilation may then continue on the next source program by merely pressing Start. Once the compiler has been loaded into the machine, as many compilations as needed may be made in consecutive runs, without having to reload the program, provided the correct end stop has been reached. In the case of the 'error stops' (see below) it is still possible to re-start from the Compiler E Card.

The compilation proceeds at the rate of approximately 1 minute per block of TAS program.

Aids and Directories 6.2.2

Before stopping, the compiler lists various directories which may be required by the user. These include an Available Space List, indicating the size and location allocated for each block of program, a Table Directory showing the drum location allocated for each Table, a Field Names Directory listing each Field used together with its I.A.S. location, and, finally a Label Directory giving the absolute address of each label used, for cross reference purposes. In addition, TAS 2 will provide Directories of Files and Records, similar to that of Field Names.

Precompilation 6.2.3

It is recommended that a run with the punching inhibited should always be carried out first for any source program. This means that normal compilation and card listing is performed, but an object program is not punched out. This reduces the compiling time by about 40% while still enabling all punching errors and misuse of the code to detected.

If an instruction has been incorrectly written the compiler indicates this by printing the words:

** THE LAST MACRO IS INCORRECT **

immediately after the invalid instruction.

Generally only one or two 'precompilations' of this sort are necessary to obtain a valid source program. Compilation with the production of an object program may then be carried out.

Error Detection 6.2.4

During compilation, the run may come to one or more error stops. From some of these it is possible to continue without punching; others force abandonment of the run.

A list of these stops will be found in Appendix C, together with a brief description of their main causes.

It should be remarked in this connection that the cause listed against a particular stop is not inevitably the error which caused the stop. Obviously, on a compiler designed to work at such a high speed and to use such a small amount of storage, it is not feasible to check every possible source of error. For this reason, an incorrect source program card may create a condition in the compiler which causes exit, either then or later, to a stop listed as being for some other cause.

Therefore, on coming to a stop for which the official reason is clearly impossible, the programmer should check carefully the program listing to that point, in order to pinpoint the source of error. Errors of this type are often to be found in Input and Output Formats.

By no means every error which can be made in a source program causes an error stop: it is perfectly possible for an incorrect program to reach the final stop. Printouts must, therefore, be checked most carefully for mis-spellings, wrong punchings, incorrect cross-references, etc.; and for this purpose full use should be made not only of the program listing but also of all aids and directories.

It is worth remarking that the commonest single source of error in TAS programs is the mis-punching of zeros as letter O.

Manual Indicators 6.2.5

Setting M.I.20 'On' will inhibit the sequence check on the source program described above.

Setting M.I. 21 'On' will inhibit punching during 'precompilation' runs.

M.I. 22 (TAS 2 only) is used to distinguish between Tape and Non-Tape programs. Setting it 'On' will cause compilation of a program *without* Tape Control Routines etc. ('Record Formats' may still be described in order to lay out the I.A.S. conveniently).

Saving Precompilation Time 6.2.6

The order of presentation of blocks (other than Global Subroutines) to the compiler is not significant. Therefore, when a program has come to an impassable error stop, been corrected, and is to be re-run, those blocks which have compiled successfully on the previous run should be moved to the back of the source pack. It is only a waste of time to recompile proved blocks while others remain unknown quantities.

THE OBJECT PROGRAM 6.3

The Punch Out 6.3.1

The object program is punched by the machine in fast read form in the following order:

(a) INPUT/OUTPUT KEYS

Input and Output Format keys for use with the control program are represented on cards carrying block numbers 101 and 102 in columns 8 to 10. (In addition, TAS 2 outputs, immediately before block 1, an un-numbered block of 7 Control Parameter Cards.)

(b) PROGRAM

The object program in blocks of 41 cards bearing block numbers: 1, 2, 3 etc.

(c) THE SPECIAL 'B' CARD

This card holds the entry point to the object program.

(d) FORWARD REFERENCES

These cards hold forward jumps to labels whose positions were unknown when the references were compiled.

(e) SUBROUTINE 'B' CARDS

There will be one of these for each Library subroutine mentioned, and it will have the number which was allocated by the source program in columns 7 to 9 .

(f) SPECIAL 'E' CARD

This is the last card, and may also hold information for the TAS Control.

Construction of the Complete Pack 6.3.2

When the Punch Out has been obtained:

(a) Library subroutines are inserted as explained above (4.7.20).

(b) Prepunched reference Tables, with 'B' Cards punched by the user from the information on the Table Directory, are inserted immediately before the TAS 'E' Card.

(c) The TAS Control program pack is placed in front of the compiled object program.

(d) Any necessary data is placed behind the 'E' Card.

(e) A control signal card is placed as the final card of the pack. It contains the 11, 0, 7 and 8 positions punched in every column from 75 to 80, and signals to the control that no more data cards are to be read. This punching must also appear in the 'DES' column.

This card is not available to the object program. It must always be present, even if the object program does not use the reader.

The object program is now ready to run, unless the TAS 1 "B" Control Pack is being used, in which case the procedure explained in Appendix S must be followed.

Testing the Object Program 6.3.3

It is expected that, by using the code, many of the usual programming errors will be avoided. However, errors in logic can only be found on test runs, and these will still be necessary.

Corrections to the program are made most easily at the source level and, since compilation is so fast, there is no hardship in recompiling an existing program.

If desired the object program may be printed out, traced, etc., using normal service routines, and may be modified directly like any other program. However, care must be exercised to avoid altering basic structures set up by the compiler and, in general, this form of alteration is not recommended.

Layout of the Object Program in the Machine 6.3.4

The I.A.S. and Drum Storage, at object program running time, is set out as follows:

TAS 1

I.A.S.

0 - 189 User's current object program block

190 - 199 Working Stores

200 - 231 TAS Control program

232 - 389 I.A.S. Fields (see Appendix P)

390 - 399 Transfer and Input/Output parameter decade

400 - 599 ("B" Control Pack only) TAS Input/Output Control program.

DRUM

The first 3000 words (0 - 2999) hold the TAS Control program, with all input/output, PPF Control, and Standard subroutines.

From word 3000 on is the area allocated by the compiler to Table Storage, as requested by the source program. Each Table begins at the next clear decade.

Following this area, commencing at the first completely free *channel*, is the user's object program, stored one block per channel.

Note: Because of the way the drum storage is allocated it is not possible to have a table of indefinite length. To allow the maximum area for a table being built up on the drum, the programmer will have to calculate the amount of free space on the drum and define his tables to cover this area.

TAS 2 (with Tape Control Routines)

I.A.S.

0 - 189 Current object program block

190 - 199 Working Stores

200 - 799 TAS Control program

800 on File buffers, Record areas, and other I.A.S. Fields (see Appendix P).

DRUM

As for TAS 1

TAS 2 (without Tape Control Routines)

I.A.S.

0 - 189 Current object program block

190 - 199 Working Stores

200 - 599 TAS Control program

600 on I.A.S. Fields (see Appendix P).

DRUM

0 - 1999 TAS Control program

2000 on Tables and object program, as for TAS 1.

Appendix A

NAMES

Rules and Restrictions

All names created by the TAS programmer must obey certain rules:

1 Maximum length is six characters.

2 The first character must be alphabetic.

3 Succeeding characters must be chosen from the range A-Z 0-$\overline{11}$ & % / . - @ $\frac{1}{4}$ $\frac{1}{2}$ $\frac{3}{4}$ and space.

4 The following names are reserved and may not be used as Field, Table or File names:

P R E S . .

I T E M . .

W S x x x x

L . . n n n

I N P U T .

O U T P U T

S P . V A R

Note: A Dot represents a blank space or zero
n represents any number
x represents any character

In addition, the following names must *not* be used for Files:

A W A Y . . D U M P . . E N D . . . I N . . . . R E C x . .

5 Every name used must be unique. No data area may have the same name as any other. The sole exception to this rule is that of common Fields in Redefining Records (see 5.2.2).

Appendix B

STATE OF ACCUMULATOR AND MILL INDICATORS, AFTER OBEYING EACH VERB

VERB	NOTES	STATE OF PRES	MILL INDICATORS AFFECTED	MILL INDICATORS SET ACCORDING TO
CLEAR	PRES	Zero	No	-
	Data area	Unchanged	No	-
COMPare		Qty. named in field A	Yes	Result of 'B' - 'A'
DIVide	with REM (NOT 'RESULT PRES')	Remainder	Yes	Not known
	All other cases	Result	Yes	Not known
GO TO		Unchanged	No	-
LOAD		See 'LOAD' (4.7.7)	Yes	See 'LOAD'
MASK		Result	Yes (except OVERFL)	Result
MODIFY		Not known	Yes	Not known
MOVE	Involving Tables	Unchanged	Yes	Not known
	All other cases	Unchanged	No	-
MULTiply		Result	Yes	Result
OBEY	Block S/R	According to verbs in S/R	According to verbs in S/R	
	Otherwise	Not known	Yes	Not known
PACK		Result	No	-
PRINT		Not known	Yes	Not known
PUNCH		Not known	Yes	Not known
READ		Not known	Yes	Not known
RENAME		Unchanged	No	-
RENEW		Not known	Yes	Not known
SHIFT		Result	No	-
STOP		Unchanged	No	-
SUM		Result	Yes	Result
TEST		Unchanged	No (except OVERFL unset if tested)	
TURN	Conditionally	Qty. named in field C	Yes	Result of 'D' - 'C'
	Unconditionally	Unchanged	No	-
WRITE		Not known	Yes	Not known
1301		According to 1301 instructions	According to 1301 instructions	

Appendix C

ERRORS DURING COMPILATION

TROUBLE	USUAL CAUSE	RESTART PROCEDURE (IF ANY)
LOOPING	Misfeed (always)	Set M.I.29
STOPS		
11 0101	I.A.S. Parity (always)	Restart compilation
11 0102	Drum Parity (always)	Press Start until parity clears
11 0103	Misfeed (always)	Replace card and press Start
11 0104	Sequence error (always)	Replace rejected card, set M.I's 20 and 21, and press Start
11 0105	Misread *or* Misfeed (always)	Replace last card read (may be rejected), and press Start
11 0106	Incorrect designation on TAS card	Set M.I's 20 and 21, and press Start
11 0107	Block too large	Abandon
11 0108	Too many Fields	Abandon
11 0109	Too many constants in block	Abandon
11 0110	Label referenced not present in program, *or* Label 0 not present in program	Set M.I's 20 and 21, press Start. (Number of times stop is reached equals number of references to missing labels. Label 0 missing = one stop.)
11 0111	Two instructions with same label	Abandon
11 0112	Last label of block missing	Set M.I.21, press Start
11 0113	Incorrect use of Working Stores	Abandon
11 0114	Too many output formats	Abandon
11 0115	S/R number outside range	Abandon
11 0116	Block S/R missing *or* too many Subroutines	Abandon
11 0117	Object of MODIFY not in block *or* MODIFYs and objects not in same order	Abandon
11 0118	Incorrect modification	Abandon
11 0119	Sequence error on 1301 relativised program cards	Restart as for 11 0104
11 0120	Too many words of machine code in 1301 section	Abandon
11 0121	Relativiser (not B) in 1301 section	Set M.I.21 and press Start
11 0122	Exit from block in 1301 section	Abandon
11 0123	Compiler parity, *or* machine error	Restart compilation. If arrived at again, print out I.A.S. and first drum, then abandon.

Incorrect modification

TROUBLE	USUAL CAUSE	RESTART PROCEDURE (IF ANY)
11 0124	Last Macro ('Move' verb) incorrect	Abandon source program
11 0125	Last Macro (any other verb) incorrect	Set M.I. 21, press Start
11 0127	Too many Forward jumps to other blocks	Set M.I. 21, press Start
11 0128	Too many jumps outside block	Set M.I. 21, press Start
11 0129	Block card missing	Set M.I. 21, press Start until stop clears
11 0130	Double use of Record Format character	Set M.I. 21, press Start
11 0131	Record Redefinition invalid	Set M.I. 21, press Start
11 0132	Redefining Record larger than original Record	Set M.I. 21, press Start
11 0133	Field name given double location	Set M.I. 21, press Start
11 0134	Tape File defined, or Tape Macro used, with M.I. 22 set	Set M.I. 21, press Start
11 0135	File buffer too large	Set M.I. 21, press Start
11 0136	Undefined File referenced	Set M.I. 21, press Start
11 0137	Undefined Record referenced	Set M.I. 21, press Start
11 0138	Incorrect AT END jump in Tape Macro (Label not in block, *or* too many jumps in block).	Set M.I. 21, press Start

Appendix D

TAS CONVENTIONS

1 *For writing of Source Program*

Each installation will have its own conventions. In TAS, as in any assembly system or autocode it is particularly important to avoid confusion between similarly shaped letters and numbers.

e.g.

(1) seven = 7
one = 1
letter i = I

(2) zero = 0
letter o = Ø

(3) two = 2
letter z = Z

2 *For punching of Source Program*

CARD DIGIT	NUMERIC NO ZONE	NUMERIC + ZONE 10	NUMERIC + ZONE 11	NUMERIC + ZONE 0	NUMERIC + ZONE 1
10	10				
11	11				
0	0				
1	1	A	J	&	
2	2	B	K	S	£ *or* %
3	3	C	L	T	¼
4	4	D	M	U	-
5	5	E	N	V	/
6	6	F	Ø	W	½
7	7	G	P	X	.
8	8	H	Q	Y	+ *or* @
9	9	I	R	Z	¾
CODED ZONE COMPONENT	1	2	3	4	5

Notice that the only differences between this and standard 1300 Series convention are those relating to the symbols in the last column.

3 *Source Program Listing*

The compiler prints out + as @. It also shows $\overline{12}$ as C, $\overline{13}$ as D, $\overline{14}$ as E and $\overline{15}$ as F, when these are punched in Source Card fields, A, B or C. Elsewhere these are not printed out at all.

Appendix E

VARYING DESIGNATION COLUMN

The TAS compilers are arranged to accept 'designations' from Data Cards in column 80 only. As this is not convenient for some customers, the details shown below will enable the designation to be accepted from any specified column or columns.

I To alter the control to accept designation in any other single column

The modification required is the placing of a single card, as set out below, immediately before the OBJECT program pack.

Note: The position of the Data card designation must remain constant for any one job.

TAS 1

"A" Control

C	D	F	A	R
1 (X)	B		0	
			1573	
		42	75	
		37	Y	
		54	Z	
		57	11	

TAS 1

"B" Control

C	D	F	A	R
1 (X)	B		0	
			1573	
		42	475	
		37	Y	
		54	Z	
		57	11	

TAS 2

C	D	F	A	R
1 (X)	B		0	
			3	
		38	2	
		37	Y	
		54	Z	
		57	11	

'Card Column' Table

Y =	TAS 1 "A"	TAS 1 "B"	TAS 2	Z = 6	7	8	9	10	11
	180	580	399	*	*	*	*	1	2
	181	581	398	3	4	5	6	7	8
	182	582	397	9	10	11	12	13	14
	183	583	396	15	16	17	18	19	20
	184	584	395	21	22	23	24	25	26
	185	585	394	27	28	29	30	31	32
	186	586	393	33	34	35	36	37	38
	187	587	392	39	40	41	42	43	44
	188	588	391	45	46	47	48	49	50
	189	589	390	51	52	53	54	55	56
	190	590	389	57	58	59	60	61	62
	191	591	388	63	64	65	66	67	68
	192	592	387	69	70	71	72	73	74
	193	593	386	75	76	77	78	79	80

2 ***Handling multi-column designations.***

Only in TAS 2 can multi-column designations be dealt with. In TAS 1 the user must distribute his cards according to a standard format, and inspect and redistribute his cards as appropriate by object program.

In TAS 2 there is a period between the physical read of a card by the TAS control, and the logical 'read' in the object program. During this time, the card image lies in a buffer at I.A.S. 280-293. The technique for handling multi-column designations consists of extracting the real designation and replacing it by an appropriate one-column 'pseudo-DES' acceptable to TAS. This should be done immediately before each 'read' instruction, by means of a '1301' section. Two sample programs are shown below, each replacing a real designation in columns 78 to 80 with a 'pseudo - DES' in column 80.

It should be noted that a real designation will be much simpler to handle if all the columns to be inspected lie in the same word of the buffer.

Layout of Buffer

I.A.S. Word	7	8	9	10	11	12	Digits
293	*	*	*	*	1	2	
292	3	4	5	6	7	8	
291	9	10	11	12	13	14	
290	15	16	17	18	19	20	
289	21	22	23	24	25	26	
288	27	28	29	30	31	32	
287	33	34	35	36	37	38	
286	39	40	41	42	43	44	
285	45	46	47	48	49	50	
284	51	52	53	54	55	56	
283	57	58	59	60	61	62	
282	63	64	65	66	67	68	
281	69	70	71	72	73	74	
280	75	76	77	78	79	80	

(a) 16 designations, defined consecutively.

TITLE						PROGRAMMER		PROGRAM No.	75 80	SHEET No. 1 / 1
Des 1 2	Label 3	Field 'A' 9 15		Field 'B' 21 27		Field 'C' 33 39		Field 'D' 45 51	Field 'E' 57 63	Card No. 69 74	Remarks
05	L 901	GO TO	L 903								Replace designation
05		TEST	10	SET	L 902						Error?
05		READ	1	DESn	L n	DESn	L n	etc			
05	L 902	TURN	10	OFF							
05		MOVE	nn	FROM	X	TO	Y				Error: preserve area
05		READ	1								to be overwritten Clear wrong card
05		MOVE	nn	FROM	Y	TO	X				Restore overwritten
05		GO TO	L 901								area
05	L 903	1301	14								'Replacing' routine (No.1.)

(b) 12 designations, defined randomly.

TITLE						PROGRAMMER		PROGRAM No.	75 80	SHEET No. /
Des 1 2	Label 3	Field 'A' 9 15		Field 'B' 21 27		Field 'C' 33 39		Field 'D' 45 51	Field 'E' 57 63	Card No. 69 74	Remarks
05	L 801	GO TO	L 803								Replace designation
05		READ	1	DESn	L n	DESn	L n	DESx	L 801		x = 'Harmless' desn.
05	L 803	1301	19								'Replacing' routine (No.2.)

I.C.T. COMPUTERS

1300 SERIES PROGRAM SHEET	JOB (No. 1) Replacing by one of up to sixteen designations, consecutively defined in the formats from zero.	BLOCK No. 903
	PROGRAMMER:-	SHEET No. 1 / 1
		/ /

C	I	D	F	A	R	NARRATIVE
1	0		41	10	B	
			37	280		Extract original designation
	1		35	11	B	Delete from card image
			65	280		
	2					
				n		Count
2	3		66	3	B	Modify comparison
			68	A		Compare first original designation
	4	4	01	7	B	
			67	2	B	
	5	4	02	3	B	More originals to compare?
			11	3500		No. Error.
3	6	8	10			Indicate error to TAS error routine
		4	00	7	B	
	7		37	12	B	Install 'distribution DES'
			63	2	B	(which equals the number of times
	8		36	280		looping has occurred)
		4	00	9	B	
4	9		45	12	B	Restore routine and count
			2	2	B	
	10	P				LINK
	11					
				151515		
5	12					
				n		K = count
	13		66	3	B	
			68	A		
	14					

FORM 1/1666 (10.63)
Printed in Great Britain by International Computers and Tabulators Limited. London.

I.C.T. COMPUTERS

1300 SERIES PROGRAM SHEET	JOB:- (No. 2) Replacing by one of up to twelve designations defined randomly	BLOCK No. 803
		SHEET No. 1 / 2
	PROGRAMMER:-	/ /

C	I	D	F	A	R		NARRATIVE
1	0		41	12	B		
			37	280			
	1		35	13	B		Extract original designation
			65	280			Delete from card image
	2		66	2	B		Modify Comparison
			68	A			compare first original designation
2	3	4	01	7	B		
			66	7	B		Modify extraction
	4		67	14	B		Count down
		4	02	2	B		More originals to compare?
	5		11	3500			No. Error.
			37	15	B		Install harmless designation
3	6		36	280			
		4	00	9	B		
	7		37	16	B		Extract appropriate 'distribution DES'
			57				from stack
	8		35	17	B		
			36	280			and install in card image
4	9		37	18	B		Restore routine and count
			63	14	B		(alternative methods set out below)
	10		65	7	B		
			64	14	B		
	11		65	2	B		
			67	2	B		
5	12	P					LINK
	13						
				151515			
	14						
				n			Count

FORM No.1/1666 Printed in Great Britain by International Computers and Tabulators Limited.

I.C.T. COMPUTERS

1300 SERIES PROGRAM SHEET

JOB:-

BLOCK No. 803

SHEET No. 2 / 2

PROGRAMMER:-

/ /

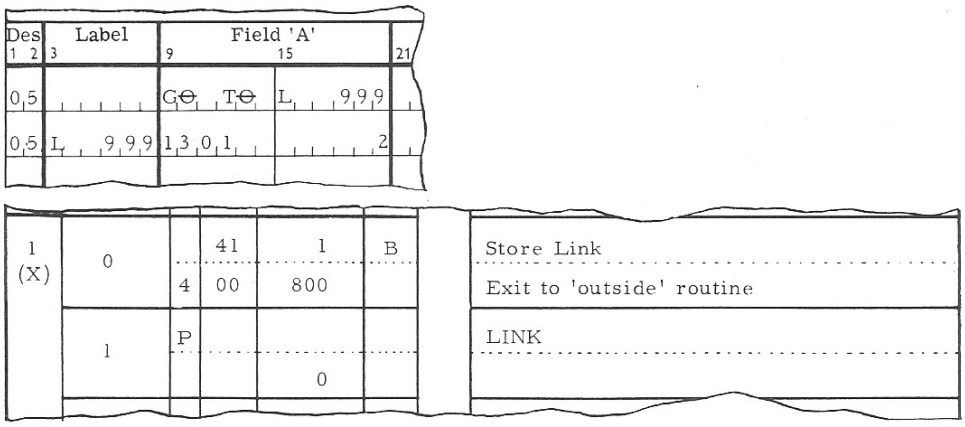
C	I	D	F	A	R	NARRATIVE
6	15			X		Harmless designation
	16		LK FE	JIHG DCBA		Original designations
	17			$\overline{15}$		
7	18			n		K = Count
	19		66 68	2 A	B	Constants used in alternative
	20		37 57	16	B	'restoration' routine shown below.
4	9		45 1	18 14	B B	Faster method of restoration, involving two extra words of storage (word 11 only
	10		45 1	20 7	B B	may be inserted in the main routine, keeping it within two decades, while
	11		45 1	19 2	B B	increasing the speed).

FORM No.1/1666 Printed in Great Britain by International Computers and Tabulators Limited.

Appendix F

UTILIZING SPARE I.A.S.

Below is shown an example of the technique of communicating between TAS and Tape Housekeeping (or any other) routines held outside the TAS-Controlled I.A.S. area.

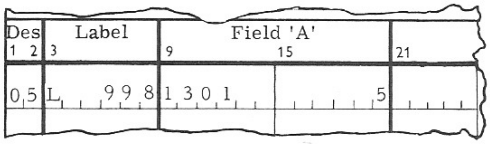


Des 1 2	Label 3	Field 'A' 9 15	21
0,5		GO TO L 999	
0,5	L 999	1301 2	

1 (X)	0		41	1	B	Store Link
		4	00	800		Exit to 'outside' routine
	1	P				LINK
				0		

At word 800 is the routine it is desired to enter. This routine will also, as its first instruction, store a link.

Note that it will also be necessary to have a short 1301 routine at the beginning of the program to call down this 'outside' routine. e.g.



Des 1 2	Label 3	Field 'A' 9 15	21
0,5	L 998	1301 5	

1	0		41	3	B	Store Link
		4	00	1	B	
	1		81	800		Move routine
			6	320		to I.A.S.
	2	4	07	4	B	Drum Parity?
		4	00	3	B	
2 (X)	3	P				LINK
				0		
	4		11	1006		Drum Parity Stop
		4	00	1	B	Repeat Transfer

Appendix G

INCORPORATING MAGNETIC TAPE SUBROUTINES

Magnetic tape subroutines can be incorporated into TAS 1 and TAS 2 (non-tape) programs to permit the processing of magnetic tape files. At the outset the following limitations should be noted:

1 No time-sharing with magnetic tape transfers is possible. Magnetic tape read and write operations can be performed simultaneously, but neither computing nor any other input or output operations can be time-shared with magnetic tape. The user must ensure that time-sharing does not occur accidentally.

2 A minimum of 1200 (TAS 2) or 800 (TAS 1) words of I.A.S. is required. The 'spare' 400 or more words hold the magnetic tape subroutines and tape read and write areas.

TECHNIQUE

The basic programming technique when processing magnetic tape is to employ subroutines to control the input and output of magnetic tape files; control not only in the sense of ensuring that the information is correctly read from, or written to, magnetic tape, but also in the sense that the records are presented for processing, and stored away after processing, in the correct sequence. To both these ends standard subroutines are available:

1 The magnetic tape housekeeping routines to ensure that the physical reading and writing is correct.

2 The record present routines to ensure that main file records are presented and stored away in correct sequence.

When processing records, whether they be from magnetic tape or any other media, it is common practice to allocate a 'record area' in which each record from a particular file is processed in turn. This avoids the program modification which would be necessary if the 'record area' were floating within the input or output area. For TAS written programs, this 'record area' will be part of the I.A.S. data storage area.

The need for subroutines to present each record from the input area to the 'record area' and to store the processed records in the output area applies only to files with multi-record blocks. For files with single-record blocks, the input or output area can obviously be the 'record area'.

To summarize the technique:

1 The records from magnetic tape will be processed in a 'record area' which will be part of the TAS data storage area. It may be possible to use the same 'record area' for records belonging to different files.

2 When the file consists of single-record blocks the input area and/or output area can also be the 'record area'; unless, of course, tape read/write overlap, which may prevent this, is required on that file.

3 When the file consists of multi-record blocks these blocks are read into, or stored for output in, the 'spare' words of I.A.S.; then:

(a) the record present routines can be used for main file control,

(b) for subsidiary input file control, a record present subroutine (see Appendix G (a)) should be used to communicate with the tape housekeeping routines in order to obtain the next block when the present one becomes exhausted. This routine should indicate the end of file to the main program,

(c) for subsidiary output file control, a record stacking subroutine (see Appendix G (b)) should be used to store away the records after processing in an output area. This routine should communicate with the tape housekeeping routines when the output area is full.

The 'spare' words should be used to contain these routines as well as the tape housekeeping routines and input/output areas.

PROBLEMS AND SOLUTIONS

The problems associated with adopting the technique just described, and possible solutions are:

1 *In TAS 1, how to define and reserve magnetic tape 'record areas' in the TAS data storage area*

A magnetic tape record can be defined, and storage reserved as though it were an output card or line - see Appendix P. The record may also be described as an input card, but magnetic tape is easier to employ, since it is not required to ensure that the number of columns specified totals 80.

The 'Rename' facility may also be used to define a record if this is found to be necessary.

Should a record be so large that it cannot be accommodated in the data storage area, it must be split and processed in small sections. The 'Rename' facility will be required to rename the data storage area when the programmer is processing each part of the record.

2 *How to incorporate and communicate with the Magnetic Tape Housekeeping and Record Present routines*

The reader is advised to familiarize himself with the Manual of Tape Housekeeping Routines and Conventions, and the Subroutine Specification Sheets for the Record Present routines.

At the outset of programming the processing of tape files in TAS, it is advisable to determine which subroutines are required and how they are going to be employed. Many tape routines require input parameters, and consideration should be given to how these parameters are to be set. It is preferable to keep the tape housekeeping and associated routines self-supporting, so that when writing the TAS program one need only perform a jump instruction to the appropriate routine which presents (or stacks) the next record into (or from) the record area.

Once the programmer has determined which routines are required and where the tape input and output areas are going to be, the I.A.S. chart for the 'reserved' words can be drawn up to determine the absolute addresses to which the TAS program must jump to obtain or store the tape records.

The setting of the relativisers for the tape housekeeping routines etc. is explained in the tape manual. In some cases, relativiser settings for the 'record area(s)' will not be known until the object program has been compiled.

Consideration should also be given to Job Set-up and to how the tape housekeeping, associated routines and tape deck statistics are to be brought down into I.A.S. at the beginning of the program. The necessary drum transfers to do this could be the initial part of the TAS program written in '1301'.

3 *Object Program Running*

Having compiled the object program, the remaining (if any) tape relativisers can be set, and the completed set of routines introduced either at the beginning of the compiled object program or before the compiled 'E' card.

Appendix G(a)

RECORD PRESENT ROUTINE
Fixed Length Records

ENTRY POINT	Word 0
TAPE INPUT AREA	RRN 4
RECORD AREA	RRN 9
END OF FILE	The user can utilize Word 1/92 - see Tape Housekeeping Routines - to communicate the end of file condition. e.g. Word 1/92 can read: 8 10 4 00 6 20
GENERAL COMMENT	If the 90 kc/s or $22\frac{1}{2}$ kc/s Single Read routine is being used, the first half of word 8, and the parameter in word 15, are both unnecessary, and the user may shorten his routine accordingly.

I.C.T. COMPUTERS

1300 SERIES PROGRAM SHEET	JOB:- RECORD PRESENT ROUTINE	BLOCK No. 20
		SHEET No. 1 / 2
	PROGRAMMER:-	/ /

C	I	D	F	A	R	NARRATIVE
1		B			20	
2			41	6	B	
	0		37	4	B	
			35	14	B	
	1		62	10	B	Is a block to be read
			42	3	B	from tape?
	2		37	13	B	
3		P				
	3			0		
	4		45	13	B	Present next record (this setting will
			X	0	9	cause initial tape read)
			37	12	B	
	5		64	4	B	
4		P				LINK
	6			0		
			45	11	B	Reset 'present next record'
	7		1	4	B	
			37	15	B	Read from tape
	8	4	00	0	98	
5		4	00	4	B	
	9					
			68			
	10	4	01	7	B	
			45	1	4	'Present next record' demodifier
	11		X	0	9	
6		P		X		Number of words per record
	12					
			1515	1515	15	
	13		1515	1515	15	
				1515	15	
	14					

FORM No.1/1666

Printed in Great Britain by International Computers and Tabulators Limited.

I.C.T COMPUTERS

1300 SERIES PROGRAM SHEET	JOB:-	BLOCK No 20
		SHEET No. 2/2
	PROGRAMMER:-	/ /

C	I	D	F	A	R		NARRATIVE
~~7~~ (X)	15	P					Tape Read Parameter

FORM No. 1/1666

Appendix G(b)

RECORD STACKER ROUTINE

Fixed Length Records

ENTRY POINT	(a) to stack a record	- Word 0
	(b) to write away a partly filled output area (not self-resetting from this entry).	- Word 10
TAPE OUTPUT AREA	RRN 3 - last word of which is word m.	
RECORD AREA	RRN 9 - key of record in word n.	
END OF FILE	It is assumed that the user will communicate with the tape write routine to have an end of file label written.	
GENERAL COMMENT	If the 90 kc/s or $22\frac{1}{2}$ kc/s Single Write routine is being used, the '37' instructions in words 5 and 13, and the parameter in word 18, are all unnecessary, and may be deleted.	

I.C.T. COMPUTERS

1300 SERIES PROGRAM SHEET	JOB:- RECORD STACKER ROUTINE	BLOCK No. 21
		SHEET No. 1 / 2
	PROGRAMMER:-	/ /

C	I	D	F	A	R	NARRATIVE
1		B			21	
2	0		41	9	B	
			37	17	B	
	1	P				Number of records per block
				Y		
	2		45	0	9	Stack record
			X	1	3	
3	3		45	n	9	Set key of record
			1	0	3	
	4		64	2	B	Modify for next record
			67	1	B	Block full?
	5	4	02	9	B	No, exit
			37	18	B	
4	6		45	15	B	Reset count and 'stack Record' instruction
			2	1	B	
	7		45	14	B	Set end of block marker
			1	m	3	
	8	4	00	0	99	Write block to tape
		4	00	9	B	
5	9	P				LINK
	10		41	9	B	
			37	1	B	Block partly full?
	11		68	15	B	
		4	01	9	B	No, exit - No Records in area
6	12		69	17	B	Form address of position
			65	7	B	of end of block marker
	13		37	18	B	
		4	00	7	B	
	14		15 15	15 15 15 15		
			15 15	15 15 15 15		

FORM No.1/1666 Printed in Great Britain by International Computers and Tabulators Limited.

I.C.T COMPUTERS

1300 SERIES PROGRAM SHEET	JOB:-	BLOCK No. 21
		SHEET No. 2 / 2
	PROGRAMMER:-	/ /

C	I	D	F	A	R	NARRATIVE
~~7~~	15	P		Y		
	16		45 X	0 1	9 3	
	~~17~~	P		X		No. of words per record
8 (X)	18	P				Tape Write Parameter

FORM No. 1/1666

Appendix H

TAS COMPILERS AND STATIONERY — OFFICIAL NUMBERS AND FORMATS

Official Subroutine numbers:

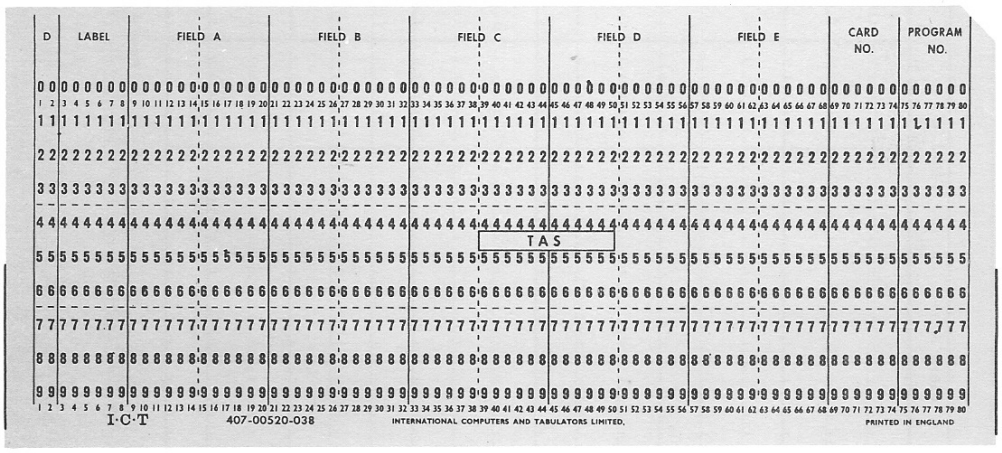
TAS 1 Compiler: I/01/00

TAS 2 Compiler: I/02/00

Official Stationery numbers:

TAS Source Program Cards	:	407/00520/038
TAS Program sheets	:	1/1729
TAS Format sheets	:	1/1738
TAS Table and Tape File Description sheets	:	1/1739
TAS Record of Field Names	:	1/1737

Examples of these are shown below. The forms are scaled down; they are in fact foolscap sheets.



I·C·T **TAS PROGRAM SHEET**

DATE / /

TITLE | PROGRAMMER | PROGRAM No. 75 80 | SHEET No. /

Des 1 2	Label 3	Field 'A' 9 15		Field 'B' 21 27		Field 'C' 33 39		Field 'D' 45 51		Field 'E' 57 63		Card No. 69 74	Remarks

FORM 1/1729 (10 63)

Printed in Great Britain by International Computers and Tabulators Limited, London.

I·C·T **TAS FORMAT SHEET**

TITLE	DATE / /
PROGRAMMER	PROGRAM No. 75 80 · SHEET No. /

DES 1 2	TYPE 3 8	KEY	FIELD NAME	CARD No. COLS 69-74	

FORM 1/1738 (10.63)

Printed in Great Britain by International Computers and Tabulators Limited, London.

I·C·T **TAS RECORD OF FIELD NAMES**

	DATE / /
TITLE	PROGRAM No.
PROGRAMMER	SHEET No. /

FIELD NAME	SIZE	REMARKS	FIELD NAME	SIZE	REMARKS

FORM 1/1737 (10.63)

Printed in Great Britain by International Computers and Tabulators Limited, London.

I·C·T TAS TABLE AND TAPE FILE DESCRIPTION SHEET

DATE / /

TITLE	PROGRAMMER	PROGRAM No. 75 80	SHEET No. /

Des 1 2	3	Name of Table 15	21	Number of Items 27	33	Length of Item 39 44		Card No. 69 74	Remarks
0 1	TABLE NAME		ITEMS		LENGTH				
0 1									
0 1									
0 1									
0 1									
0 1									
0 1									
0 1									

Des 1 2	3	Input or Output 9	File Name 15	21	Block Length 27	Deck Address 33	Position of Keyword 39	Sim. 45	Associated File 51	Single or Double 57 62		Card No. 69 74	Remarks
0 6	FILE			LENGTH		DECK							
0 6	FILE			LENGTH		DECK							
0 6	FILE			LENGTH		DECK							
0 6	FILE			LENGTH		DECK							
0 6	FILE			LENGTH		DECK							
0 6	FILE			LENGTH		DECK							
0 6	FILE			LENGTH		DECK							
0 6	FILE			LENGTH		DECK							

FORM 1/1739 (10.63)

Printed in Great Britain by International Computers and Tabulators Limited, London.

Appendix I

SAMPLE PROGRAM LISTING

The program represents an invoicing run for a Wines and Spirits Distributor.

Each order for goods is presented on two successive input cards, the first holding the customer's reference number, name and address, and the second the customer's reference number (for checking purposes), and details of up to eight items ordered. For each item are stated the quantity ordered, a stock number and a type code. The second and third digits of the stock number form a 'description key', used to obtain, from a self-indexing drum Table, the name and price of the particular item; the 'type code' references another self-indexing Table of quantity types.

From this information an invoice is produced giving details of each item ordered, as well as totals, discount etc.; and also a Stores Control voucher.

Checks are made, firstly on the correct order of cards presented, secondly to ensure that each group consists of only two cards, and lastly to ascertain that the reference numbers of the two cards in any one group correspond. Should reference numbers not correspond, an error routine is entered which prints on the spoiled invoice a standard 'error' statement (also held on a drum Table) incorporating the wrong numbers. In other cases an error stop is reached so that the operator may reverse the cards.

The first card of each run is a special card containing the date and the number to be allocated to the first invoice produced. At the end of the run a card is punched with the next unused invoice number. This will be re-fed as the first card of the next day's run.

INPUT CARDS

1 DES 1 Invoice Number Card (First Card)

Columns:

1 - 58 Blank
59 - 70 Invoice Number (D)
71 - 79 Date - Punched by hand (e.g. 15 JAN 65) (A)
80 Designation ('1')

2 DES 2 Name and Address Card

Columns:

1 - 7 Customer's Reference Number (D)
8 - 31 Customer's Name (A)
32 - 55 Street (A)
56 - 67 Town (A)
68 - 79 County (A)
80 Designation ('2')

3 DES 3 Items Card

Columns:

1 - 7	Customer's Reference Number (D)
8 - 12	Item 1: Stock No. (D)
13 - 15	Quantity ordered (D)
16	Type Code (D)
17 - 21	Item 2: Stock No. (D)
22 - 24	Quantity ordered (D)
25	Type Code (D)

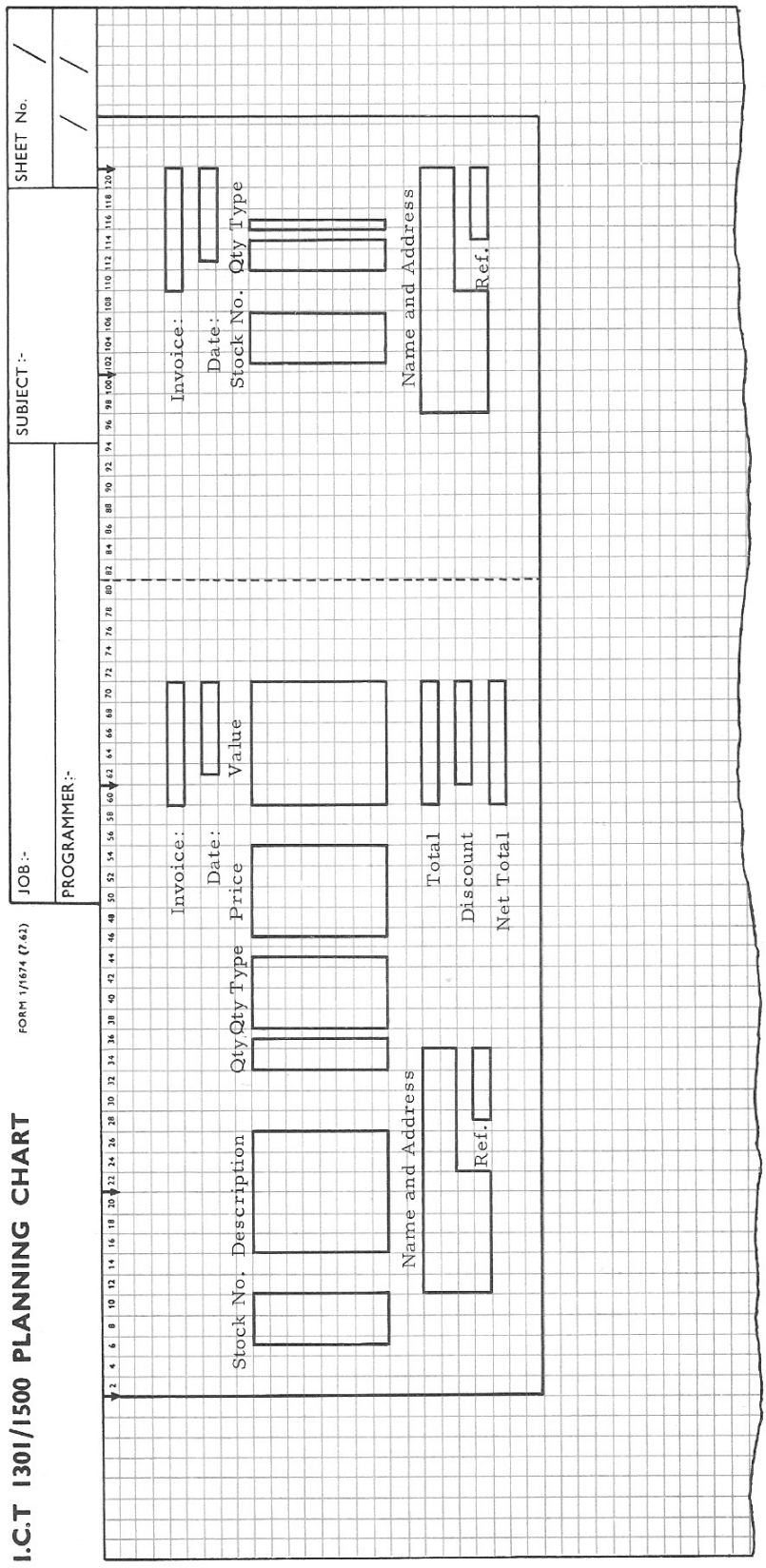
and so on down to.....

71 - 75	Item 8: Stock No. (D)
76 - 78	Quantity ordered (D)
79	Type Code (D)
80	Designation ('3')

OUTPUT CARD (Invoice Number Card)

Columns:

1 - 58	Blank
59 - 70	Invoice Number (D)
71 - 79	Blank
80	Designation ('1')


I.C.T 1301/1500 PLANNING CHART
FORM 1/1674 (7.62)
JOB:-
PROGRAMMER:-
SUBJECT :-
SHEET No.
Stock No. Description Qty Qty Type Price
Invoice:
Date:
Value
Name and Address
Ref.
Total
Discount
Net Total
Invoice:
Date:
Stock No. Qty Type
Name and Address
Ref.

TABLES

1 Descriptions and Prices

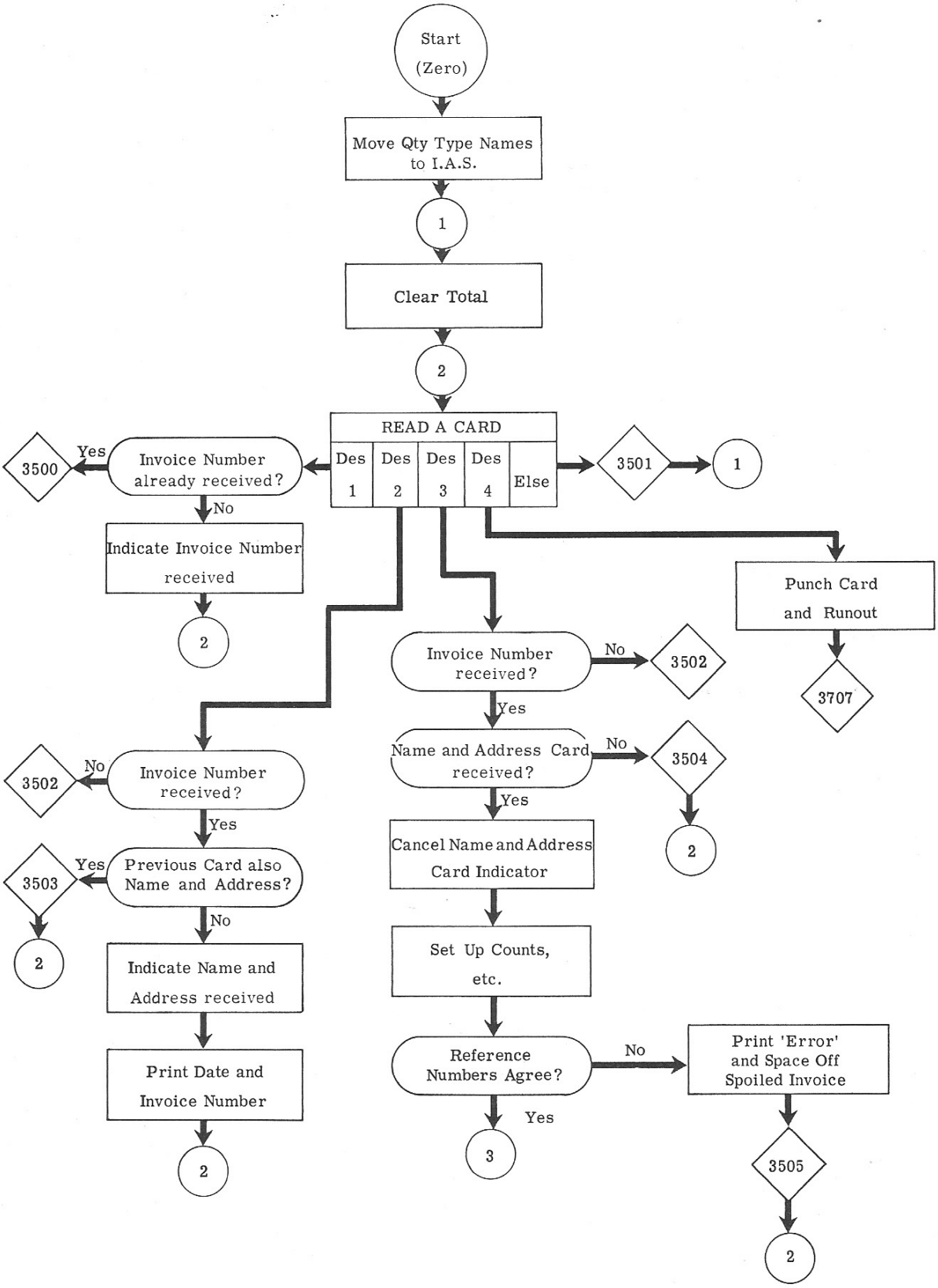
Each Item consists of 3 words. Words 1 and 2 are the alphanumeric description of the item and word 3 is the price of a quantity of that item.

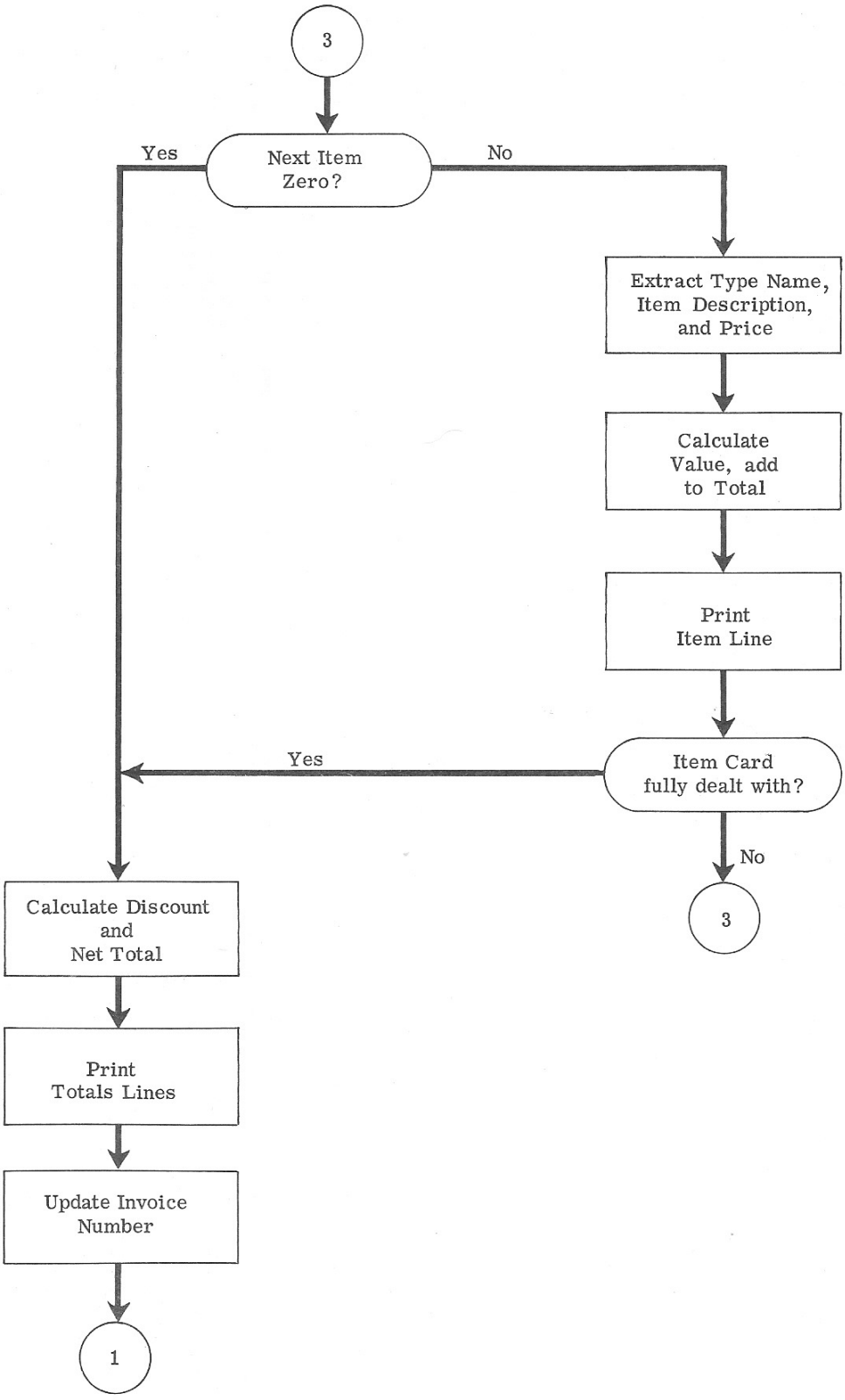
Item No.	Item	Price
0	LIGHT ALE	9 8
1	PALE ALE	9 9
2	BROWN ALE	9 6
3	BITTER CAN	14 3
4	MILD CAN	9 0
5	LAGER	15 8
6	SODA SYPHONS	13 3
7	CIDER	8 4
8	WHISKY	1 9 0
9	GIN	1 7 0
10	RUM	1 11 0
11	BRANDY	1 10 6
12	SHERRY DRY	4 12 6
13	SHERRY SWEET	4 15 6
14	PORT	4 18 0
15	CHAMPAGNE	5 6 0
16	MARTINI	8 16 9
17	VODKA	1 11 0
18	PIMMS NO 1	1 11 0
19	PIMMS NO 2	1 11 0
20	WINE	2 4 0
21	BABYCHAM	4 18 0
22	PINK LADY	4 18 0
23	LEMONADE	4 2 0
24	MINERALS	4 8 0
25	LIME	1 2 6
26	ORANGE	1 2 6
27	BITTER LEMON	4 18 0

2 Quantity Types

Each item consists of two words, the alphanumeric description of the quantity type.

Code	Quantity
0	CRATES
1	GROSS
2	DOZEN
3	BOTTLES
4	CASES


Start
(Zero)
Move Qty Type Names to I.A.S.
1
Clear Total
2
READ A CARD
Des 1
Des 2
Des 3
Des 4
Else
3501
1
Yes
3500
Invoice Number already received?
No
Indicate Invoice Number received
2
Punch Card and Runout
3707
Invoice Number received?
No
3502
Yes
Name and Address Card received?
No
3504
Yes
2
Cancel Name and Address Card Indicator
Set Up Counts, etc.
Reference Numbers Agree?
No
Print 'Error' and Space Off Spoiled Invoice
3505
2
Yes
3
No
3502
Invoice Number received?
Yes
Yes
3503
Previous Card also Name and Address?
No
2
Indicate Name and Address received
Print Date and Invoice Number
2



```
TAS 2  COMPILATION      PROGRAM



TABLES

0001  01    TABLE    NAME     BOOZE     ITEMS        28    LENGTH        3
0003  01                      ERROR                  20                  1
0002  01                      PACKS                   5                  2



INPUT FORMATS

0004  03    DES1     I58                D12       INV NO    A09/2     DATE      D01       ONE
0005  03    DES2     D07      CR N&A    A24/4     NAME      A24/4     STREET    A12/2     TOWN      A12/2     COUNTY
0006  03    DES2     I01
0007  03    DES3     D07      CRITEM    D05       STK NO    D03       QTY       D01       TYPE      D05       STK2
0008  03    DES3     D03      Q2        D01       T2        D05       S3        D03       Q3        D01       T3
0009  03    DES3     D05      S4        D03       Q4        D01       T4        D05       S5        D03       Q5
0010  03    DES3     D01      T5        D05       S6        D03       Q6        D01       T6        D05       S7
0011  03    DES3     D03      Q7        D01       T7        D05       S8        D03       Q8        D01       T8
0012  03    DES3     I01



OUTPUT FORMATS

0013  03    LINE11   D70      INV NO    DC0       INV NO
0014  03    LINE12   A70/2    DATE      AC0/2     DATE
0015  03    LINE13   D10      STK NO    DW6       STK NO    D35       QTY       DH3       QTY       DH5       TYPE
0016  03    LINE13   A26/2    DESCR     S54       PRICE     A43/2     QTDESC    S70       VALUE
0017  03    LINE14   A34/4    NAME      ACQ/4     NAME      S70       TOTAL
0018  03    LINE15   A34/4    STREET    ACQ/4     STREET
0019  03    LINE16   A22/2    TOWN      AW8/2     TOWN      S70       DISCNT
0020  03    LINE17   A22/2    COUNTY    AW8/2     COUNTY    D34       CR N&A    D 0       CR N&A
0021  03    LINE18   S70      NETTOT
0022  03    LINE19   A60/10   ERRLHS    AC0/10    ERRRHS    D70       CR N&A    DH0       CRITEM


BLOCK NO.    1

0024  02    BLOCK    TO       L  999
0025  05          L    0      MOVE        50    FROM      ERROR     TO        ERRLHS
0026  05          L    1      CLEAR       TOTAL
0027  05          L    2      READ         1    DES1      L    6    DES2      L   20    DES3      L   30    DES4      L    7
0028  05                      STOP      3501    RESUME    L    1
0029  05          L    6      TEST        10    SET       L  350
0030  05                      TURN        10    ON
0031  05                      GO TO     L    2
0032  05          L  350      STOP      3500
0033  05          L    7      PUNCH        1              CARD01
0034  05                      GO TO     RUNOUT
0035  05                      STOP      3707
0036  05          L   20      TEST        10    SET       L   21
0037  05          L  352      STOP      3502
0038  05          L   21      TEST        11    SET       L  353
0039  05                      TURN        11    ON
0040  05                      PRINT        2    SP06PT    LINE11    SP02      LINE12
0041  05                      GO TO     L    2
0042  05          L  353      STOP      3503    RESUME    L    2
0043  05          L   30      TEST        10    SET       L   31    UNSET     L  352
0044  05          L   31      TEST        11    SET       L   32
0045  05                      STOP      3504    RESUME    L    2
0046  05          L   32      TURN        11    OFF
0047  05                      LOAD                *            8    RESULT    COUNT
0048  05                      CLEAR     MODFYR
0049  05                      COMP      CR N&A    WITH      CRITEM    MORE      L  355    EQUAL     L    3    LESS      L  355
0050  05          L  355      PRINT        3    SP05      LINE19    SP05      LINE19    SP06      LINE19
0051  05                      STOP      3505    RESUME    L    2
0052  05          L    3      MODIFY    STK NO    IN      L 33/8    WITH      MODFYR
0053  05                      RENAME    STK NO    AS      STOCK
0054  05          L   33      MOVE         3    FROM      STOCK     TO        STOCK
0055  05                      LOAD                ADD       QTY       RESULT    PRES
0056  05                      TEST      ZERO      SET       L    4
0057  05                      SHIFT     STOCK     LEFT C       8    RESULT    PRES
0058  05                      SHIFT     PRES      RIGHTZ      10    RESULT    PRES
0059  05                      MULT      PRES      *            3    RESULT    WS  01    OVER10          0
0060  05                      MODIFY    BOOZE     IN      L 34/8    WITH      WS  01
0061  05          L   34      MOVE         3    FROM      BOOZE     TO        DESCR
0062  05                      LOAD      TYPE      ADD       TYPE      RESULT    WS  02
0063  05                      MODIFY    IASTAB    IN      L 35/8    WITH      WS  02
0064  05          L   35      MOVE         2    FROM      IASTAB    TO        QTDESC
0065  05                      MULT£10   PRICE     BY        QTY       RESULT    VALUE     OVER10          0
0066  05                      LOAD£     PRES      ADD TO    TOTAL
0067  05                      LOAD                ADD       MODFYR    RESULT    PRES
0068  05                      TEST      ZERO      SET       L   36
0069  05                      PRINT        1      SP01      LINE13
0070  05                      GO TO     L   37
0071  05          L   36      PRINT        1      SP03      LINE13
0072  05          L   37      LOAD         3      ADD TO    MODFYR
0073  05                      LOAD         1      SUB FR    COUNT
0074  05                      TEST      POS       SET       L    3
0075  05          L    4      MULT£10   TOTAL     *            4    RESULT    DISCNT    OVER10          2
0076  05                      LOAD£     TOTAL     SUB       DISCNT    RESULT    NETTOT
0077  05                      LOAD         3      ADD TO    COUNT
0078  05                      MODIFY    SP VAR    IN      L 38/8    WITH      COUNT
0079  05          L   38      PRINT        5    SP VAR    LINE14    SP01      LINE15    SP01      LINE16    SP01      LINE17
0080  05                      PRINT     CONTD     SP01      LINE18
0081  05                      LOAD         1      ADD TO    INV NO
0082  05          L  999      GO TO     L    1



END OF PROGRAM
```

AVAILABLE SPACE LIST

BLOCK NO.	WORDS USED	WORDS AVAILABLE	START DECADE ADDRESS
1	139	61	220

TABLE NAMES DIRECTORY

TABLE NAME	START DECADE ADDRESS
BOOZE	200
PACKS	209
ERROR	210

RECORD NAMES DIRECTORY

RECORD NAME	I.A.S. ADDRESS

FIELD NAMES DIRECTORY

FIELD NAME	IAS ADDRESS	FIELD NAME	IAS ADDRESS	FIELD NAME	IAS ADDRESS	FIELD NAME	IAS ADDRESS
INV NO	600	STREET	609	QTY	619	S3	624
DATE	601	TOWN	613	TYPE	620	Q3	625
ONE	603	COUNTY	615	STK2	621	T3	626
CR N&A	604	CRITEM	617	Q2	622	S4	627
NAME	605	STOCK	618	T2	623	Q4	628
T4	629	Q6	634	S8	639	QTDESC	645
S5	630	T6	635	Q8	640	VALUE	647
Q5	631	S7	636	T8	641	TOTAL	648
T5	632	Q7	637	DESCR	642	DISCNT	649
S6	633	T7	638	PRICE	644	NETTOT	650
ERRLHS	651						
ERRRHS	661						
IASTAB	671						
COUNT	681						
MODFYR	682						

LABEL DIRECTORY

LABEL NO.	BLOCK	IAS ADDRESS
0	1	1
1	1	4
2	1	5
3	1	38
4	1	69
6		11
7		14
20	1	21
21	1	23
30	1	28
31	1	29
32	1	31
33	1	40
34	1	48
35	1	56
36	1	64
37	1	66
38	1	76
350	1	13
352	1	22
353	1	27
355	1	35
999	1	79

1378412

13 DEC 64

11502	CHAMPAGNE	36 BOTTLES	5 6 0	190 16 0	
30821	WHISKY	24 BOTTLES	1 9 0	34 16 0	
10921	GIN	12 BOTTLES	1 7 0	16 4 0	
31654	MARTINI	3 CASES	8 16 9	26 10 3	
61800	PIMMS NO 1	6 BOTTLES	1 11 0	9 6 0	
40372	BITTER CAN	5 DOZEN	14 3	3 11 3	
40122	PALE ALE	12 CRATES	9 9	5 17 0	
				287 0 6	
				11 9 7	
				275 10 11	

MRS. K.B.HOSKING
6 HOLLAND PARK ROAD
KENSINGTON
LONDON W.14 2187439

1378412

13 DEC 64

11502	36 3	
30821	24 3	
10921	12 3	
31654	3 4	
61800	6 3	
40372	5 2	
40122	12 0	

MRS. K.B.HOSKING
6 HOLLAND PARK ROAD
KENSINGTON
LONDON W.14 2187439

Sample Invoice

Appendix J

LIMITATIONS

	TAS 1	TAS 2
SIZE OF PROGRAM		
Number of blocks (excluding Library Subroutines)	99	99
Number of words in each compiled block (including Working Stores)	200	200
Number of Modify Verbs in one block	16	16
Number of constants in any one block	100	100
Number of exits from any one block	50	50
Number of Block Subroutines in any one block	20	20
Number of Block Subroutines in a program	50	50
Number of Global and/or Library Subroutines in program	30	30
Number of forward jumps (to labels or subroutines not yet presented to compiler) in program	200	200
DATA		
Number of Tables in program	80	80
Number of Items in one Table	999	999
Number of words in one Item	99	99
Number of I.A.S. words of Fields (x = size of machine I.A.S.)	158	x - 600 (non-tape) x - 800 (tape)
Number of Working Stores in each block	10	10
INPUT-OUTPUT		
Input Formats	10	16
Output Formats	20	30
Fields per Input Card	30	40
Fields per Output Card or Line	15	29
Size (in words) of Field input (Alpha)	8	10
(Other)	1	1
Size (in words) of Field output (Alpha)	10	10
(Other)	1	1

		TAS 1	TAS 2
Files		-	8
Files per Deck Address		-	1
Record Formats		-	26
Size (in words) of Record		448	448
Size (in words) of File block		449	449
IN ONE INSTRUCTION			
Cards Read		1	1
Cards punched		1	1
Lines printed		10("A")	10
		1("B")	
Consecutive words summed		50	50
Consecutive words moved:	Unmodified instruction	50	50
	Modified instruction	10	20
	Table to Table move	1	1
RANGES			
Label numbers		0 to 999	0 to 999
Stop numbers		2001 to 3999	2001 to 3999
Program Indicators		10 to 18	10 to 18
Manual Indicators		20 to 28	20 to 28
Input card DES's		0 to 9	0 to $\overline{15}$
Output Formats		1 to 20	1 to 30
Deck Addresses		-	1 to 8
Subroutine numbers:	Block	150 to 199	150 to 199
	Global	200 to 299	200 to 299
	Library	300 to 399	300 to 399
FIXED ADDRESSES			
Sprag engaged		755 (drum)	334 (I.A.S.)
RRN 80 (Job Set-up) (Drum decade)		-	200
RRN 86 (Job Set-up) (Drum decade)	- $\frac{1}{2}$" or 1" system	-	272
	- $\frac{1}{4}$" system	-	244

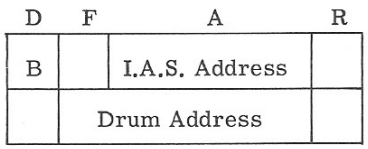
Appendix K

'B' 'C' 'E' AND 'F' CARDS

Four special types of 1300 series machine code program cards are of interest to the TAS user. These are:

1 B FOR BLOCK

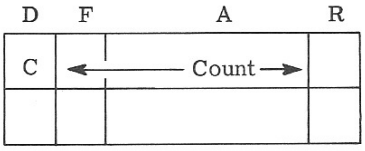
Programs written in blocks require that each block be preceded, on entry, by the first address, or block, relativiser. This takes the following form and is equivalent to a double-length instruction:



The designation B identifies it as a block relativiser to the computer. The first word that follows is stored on the drum in the word given as the 'drum address' in the block relativiser. Subsequent words are stored successively, blanks being ignored, until the block has ended. This will be signalled by an 'end of block marker' (an X punched in column 17) in the last card of the block. The I.A.S. address given in the block relativiser has nothing to do with the address taken up by the words and may therefore be omitted from B cards for TAS Tables.

2 C FOR CLEAR

As it is possible to fill an area with information by means of normal words of instructions or constants, so is it possible to zeroize an area on the drum, before a program is entered, by means of a 'C' word. This takes the form



where 'count' is the number of successive words to be zeroized.

This word may appear as a normal program instruction in a block of program or data, and takes the place of the number of words specified by 'count'. It may be preceded and/or followed by other normal words of program or data, but its usual function in TAS will be to zeroize those Tables whose contents are expected to be zero when the object program is entered and, in this case, each 'C' word will stand alone, preceded by an appropriate 'B' word.

3 E FOR ENTRY

A word with designation E is placed at the end of a program on entry to indicate that it is desired to begin operation of that program. This word takes the following form:

D	F	A	R
E		I.A.S. Address	
	N	Decade Address	

I.A.S. Address is the I.A.S. Address at which the program is to start. This address must be in the first 200 locations.

N is the Number of decades transferred from the drum and must be ⩽20.

Decade Address is the Decade Address at which the transfer from the drum is to begin.

Note: Both addresses must be in 'absolute' form.

Acting on this information, 'Initial Orders' transfer the specified number of decades, starting at the given decade, into the I.A.S., starting at location 0. Control is then transferred to the I.A.S. Address given. An example is as follows:

D	F	A	R
E		36	
	12	631	

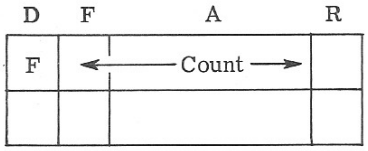
This means "After transferring 12 decades from the drum, the contents of the first drum location 6310 going to I.A.S. location 0, enter program at I.A.S. location 36".

4 F FOR FAST READ

Machine-coded program cards take two forms. The usual 'relativised' cards have programs punched as they are written (see examples in Appendix G above) with three words to a card.

'Fast Read' cards, however, are punched in 'absolute' twelve digit form, each word being punched as it appears within the computer, and five words on each card. This calls for designations included in the most significant digit of the address, negative numbers in complementary form, and all addresses absolute. The TAS object program is punched out in this way.

Each group of 'Fast Read' cards must be immediately preceded by an 'F' word, in relativised form, informing Initial Orders of the number of Fast Read words in the group. This word takes the form



where 'count' is the number of Fast Read words to follow.

The computer accepts each 'Fast Read' card as holding five words, and blank words are not ignored but entered as zeros and counted. If 'count' is not a multiple of five, it is the end of the *last* card in the group which will be ignored.

Each group of 'Fast Read' cards must have its own sequence numbering, starting from 001, punched in columns 18 to 20. For this reason, cards containing 'F' words always have an 'end of block marker' (an X punched in column 17).

Appendix L

FORMATS OF INSTRUCTIONS

Set out overleaf are the formats the various instructions may take, listed under the governing verbs.

The various possible alternatives for each card field are displayed. This list is designed to show only the technicalities of writing the actual words and numbers. It is not a list of what is obligatory and what optional, nor does it attempt to show acceptable 'combinations' of card fields. On these points the sections on the verbs themselves should be consulted.

In the list, the presence of the letters F, T, W and/or P in a card field indicates that the name of a Field, Table or Working Store, or 'PRES', respectively is acceptable in that position. An * indicates that Fields, and Tables if acceptable, may be Itemized.

I·C·T **TAS PROGRAM SHEET**

DATE / /

TITLE | PROGRAMMER | PROGRAM No. | 75 80 | SHEET No. /

Des 1 2	Label 3	Field 'A' 9 15		Field 'B' 21 27		Field 'C' 33 39		Field 'D' 45 51		Field 'E' 57 63		Card No. 69 74	Remarks
		CLEAR	FT*W P	AND	FT*W	AND	FT*W	AND	FT*W	AND	FT W		
		COMP	FT W P	WITH	FT W	MORE	L nnn	EQUAL	L nnn	LESS	L nnn		
		COMP£	FT W P	±nnnnn	n n n n n		ENDSUB		ENDSUB		ENDSUB		
		COMP£n	FT W P				LINK		LINK		LINK		
		DIV	FT*W P	BY	FT*W	RESULT	FT*W P						
		DIV £	FT*W P	BY £	FT*W			ROUND					
		DIV £10	FT*W P	±nnnnn	n n n n n			REM	FT*W P				
		GO TO	L nnn										
		GO TO	ENDSUB										
		GO TO	LINK										
		GO TO	RUNOUT										

FORM 1/1729 (10 63)

Printed in Great Britain by International Computers and Tabulators Limited, London.

I·C·T **TAS PROGRAM SHEET**

DATE / /

TITLE | PROGRAMMER | PROGRAM No. | 75 80 | SHEET No. /

Des 1 2	Label 3	Field 'A' 9 15	Field 'B' 21 27	Field 'C' 33 39	Field 'D' 45 51	Field 'E' 57 63	Card No. 69 74	Remarks
		LOAD FT*W P	ADD FT*W	As field 'B'	As field 'B'	ADD TO FT W		
		LOAD £ FT*W P	SUB FT*W			SUB FR FT W		
		LOAD £n FT*W P	±nnnnnn nnnnnn			RESULT FT W P		
			ADD TO FT*W					
			SUB FR FT*W					
			RESULT FT*W P					
		LOAD	ADD FT*W	RESULT FT*W P				
		LOAD £						
		LOAD £n						
		LOAD	SUB FT*W	As for main 'Load' verb	As for main 'Load' verb	As for main 'Load' verb		
		LOAD £						
		LOAD £n						
		LOAD	±nnnnnn nnnnnn	As for main 'Load' verb	As for main 'Load' verb	As for main 'Load' verb		
		LOAD £						
		LOAD £n						

FORM 1/1729 (10 63)

Printed in Great Britain by International Computers and Tabulators Limited, London.

I·C·T **TAS PROGRAM SHEET**

DATE / /

TITLE | PROGRAMMER | PROGRAM No. | 75 80 | SHEET No. /

Des 1 2	Label 3	Field 'A' 9 15		Field 'B' 21 27		Field 'C' 33 39		Field 'D' 45 51		Field 'E' 57 63		Card No. 69 74	Remarks
		LOAD	n	ADD TO	FT*W								n = 1, 2, 3 only
		LOAD £	n	SUB FR	FT*W								
		LOAD £10	n										
		LOAD n	F	a a a a a a	a a a a a a	a a a a a a	a a a a a a	a a a a a a	a a a a a a	a a a a a a	a a a a a a		
		MASK	FT*W P	n n n n n n	n n n n n n	RESULT	FT*W P						
		MODIFY	FT	IN	Ln n n / z	WITH	F *W						1 ≤ nnn ≤ 999 z = A, B, C, D, E
			SP VAR										
		MOVE	n n	FROM	FT*W	TO	FT*W						1 ≤ nn ≤ 50
		MULT	FT*W P	BY	FT*W	RESULT	FT*W P	OVER	1 0 n				
		MULT £	FT*W P	± n n n n n n	n n n n n n								
		MULT £n	FT*W P										

FORM 1/1729 (10 63)

Printed in Great Britain by International Computers and Tabulators Limited, London.

I·C·T **TAS PROGRAM SHEET**

DATE / /

TITLE | PROGRAMMER | PROGRAM No. 75 80 | SHEET No. /

Des 1 2	Label 3	Field 'A' 9 15		Field 'B' 21 27		Field 'C' 33 39		Field 'D' 45 51		Field 'E' 57 63		Card No. 69 74	Remarks
		OBEY	SUBnnn										150 ⩽ nnn ⩽ 299
		OBEY	SUBnnn	INPUT	F *W	F *W	F *W	F *W	F *W	F W	F W		300 ⩽ nnn ⩽ 399
					OUTPUT	OUTPUT	OUTPUT	OUTPUT	OUTPUT	OUTPUT	OUTPUT		
		OBEY	CONTD	F *W	As for above →								
				OUTPUT									
		PACK	FT*W PWITH	FT*W	RESULTFT*W P								
		PRINT	nn	SPnn	LINEnn	As field 'B'		As field 'B'		As field 'B'			Restricted to one line per macro if
		PRINT	CONTD	SPnnPT	LINEnn								TAS 1 "B" Control Pack is being used.
				SP VAR	LINEnn								
		PUNCH	1		CARDnn								
		READ	1	DESn	L nnn	DESn	L nnn	As field 'C'		As field 'C'			
		READ	CONTD		ENDSUB	ELSE	ENDSUB						
					LINK		LINK						

FORM 1/1729 (10 63)

Printed in Great Britain by International Computers and Tabulators Limited, London.

I·C·T **TAS PROGRAM SHEET**

DATE / /

TITLE | PROGRAMMER | PROGRAM No. 75 80 | SHEET No. /

Des 1 2	Label 3	Field 'A' 9 15		Field 'B' 21 27		Field 'C' 33 39		Field 'D' 45 51		Field 'E' 57 63		Card No. 69 74	Remarks
		RENAME	F *	AS	F								
		SHIFT	FT*W P	LEFT	nn	RESULT	FT*W P						
				LEFT C	nn								
				RIGHTZ	nn								
				RIGHT	nn								
		STOP	nnnn	RESUMEL	nnn								
		SUM	nn	FROM	FT*W	RESULT	FT*W P						
		SUM £	nn										
		SUM £n	nn										
		TEST	nn	SET	L nnn	UNSET	L nnn						
			POS		ENDSUB		ENDSUB						
			NEG		LINK		LINK						
			ZERO										
			OVERFL										

FORM 1/1729 (10.63)

Printed in Great Britain by International Computers and Tabulators Limited, London.

I·C·T **TAS PROGRAM SHEET**

DATE / /

TITLE								PROGRAMMER				PROGRAM No. 75 80	SHEET No. /
Des 1 2	Label 3	Field 'A' 9 15		Field 'B' 21 27		Field 'C' 33 39		Field 'D' 45 51		Field 'E' 57 63		Card No. 69 74	Remarks
		TURN	nn	ON		IF	FT*W	PMORE	FT*W				
		TURN£	nn	OF				EQUAL		+nnnnnnnnnnn			
		TURN£n	nn	OVER				LESS					
		1301	nnn							.			

FORM 1/1729 (10.63)

Printed in Great Britain by International Computers and Tabulators Limited, London.

I·C·T **TAS PROGRAM SHEET**

DATE / /

TITLE | PROGRAMMER | PROGRAM No. | 75 80 | SHEET No. /

Des 1 2	Label 3	Field 'A' 9 15		Field 'B' 21 27		Field 'C' 33 39		Field 'D' 45 51		Field 'E' 57 63		Card No. 69 74	Remarks
	TAPE	MACROS											
		READ	IN	FROM	a a a a a a	AT END	n n n						
		READ	REC a	FROM	a a a a a a	AT END	n n n						
		RENEW	a a a a a a										
		WRITE	a a a a a a	TO	b b b b b b	AT END	n n n						
		WRITE	AWAY	TO	a a a a a a								File is NOT 'SIM'
		WRITE	AWAY	TO	a a a a a a	AT END	n n n						File is 'SIM'
		WRITE	DUMP	DECK	n	BLOCK	n n n	IAS	i i i	DRUM	d d d d		
		WRITE	END	TO	a a a a a a								
		WRITE	REC a	TO	a a a a a a								File is NOT 'SIM'
		WRITE	REC a	TO	a a a a a a	AT END	n n n						File is 'SIM'

FORM 1/1729 (10.63)

Printed in Great Britain by International Computers and Tabulators Limited, London.

Appendix M

ARRANGING DATA ON MAGNETIC TAPE

A Tape File consists of various Records, which are held on tape in groups, or 'blocks'. TAS, and all standard tape sorts, demand that these records and blocks conform to a standard layout.

RECORD LAYOUT

The three most-significant digits of the first word of each record contain the number of words in that record. No distinction is made between fixed- and variable-length records.

The key may consist of any number of consecutive words, starting at any word in the record, provided that its position is constant within any one file. (The number of words of the record preceding the first word of the key will be a parameter that has to be given to tape sorting and other routines.)

The first word of any key must have value in the range 000000 000001 to 499999 999998 inclusive.

BLOCK LAYOUT

The first word of a block must have repeated in it the most-significant word of the key of the last record in the block (i.e. the highest key). If the block consists of unsorted records, the contents of the first word are immaterial.

There then immediately follows the first record in the block.

The last word of the last record in the block is followed by the end of block marker - a word of 15's.

Example

A Tape Block lying in I.A.S. from word 900 on

Word 900	1st word of key of record No.3 (Word 935)	
Word 901	007nnn nnnnnn	1st record
.		
Word 903	1st word of key	
Word 904	2nd word of key	
.		
.		
.		
Word 908	025nnn nnnnnn	2nd record
.		
Word 910	1st word of key	
Word 911	2nd word of key	
.		
.		
.		
.		
.		
Word 933	016nnn nnnnnn	3rd record
.		
Word 935	1st word of key	
Word 936	2nd word of key	
.		
.		
.		
Word 949	$\overline{15}\overline{15}\overline{15}\overline{15}\overline{15}\overline{15}$	$\overline{15}\overline{15}\overline{15}\overline{15}\overline{15}\overline{15}$

Appendix N

NOTES ON INPUT AND OUTPUT MACROS

1 READ, PRINT AND PUNCH (TAS 1)

These are all controlled by PPF-C. The control program has buffers for each unit. On being given the command Read, it presents the next card from the Read buffer, distributing it according to its format. On being given the command Print or Punch, it distributes to the next vacant location in the appropriate buffer.

When the Read buffer becomes empty, or the Print or Punch buffer becomes full, physical action is necessary. The control dumps away the object program, and calls down PPF which Reads, Prints and Punches simultaneously until all buffers are full or empty as appropriate.

This method will ensure a saving in program running time if two or more peripherals are being used at the same time in a reasonable ratio to one another. The ideal ratio is obtained when the peripherals are in operation for approximately the same length of time e.g. with a 600 c.p.m. card reader, 600 c.p.m. printer and 100 c.p.m. card punch, the ideal ratio would be 6 Read, 6 Print, 1 Punch.

2 READ, PRINT AND PUNCH (TAS 2)

The TAS Set-up routine (performed automatically before label 0 is arrived at) reads one card into a buffer. Subsequently, throughout the program, on being given the instruction 'Read', the control presents and distributes the buffered card, time-sharing this with a physical read of the next card in the hopper.

When the control receives the first instruction to Print, it distributes the line specified and holds it in a buffer. On each subsequent Print order, it prints the line in the buffer, and time-shares this with distributing the new line specified. 'GO TO RUNOUT' clears away the last line of the program and spaces 199 lines.

The first Punch order causes distribution, row binarizing, and punching. Each subsequent order causes the previous card to be checked, and the new card to be distributed, row binarized, and punched. In order to 'check' the last card punched, it will be necessary to 'Punch' one dummy blank card immediately before coming to the final stop.

3 RUNOUT (TAS 1)

GO TO RUNOUT will continue to read in cards until it finds a 'closer' card (see above, 6.3.2). Therefore, such a card must be in the reader every time this instruction is obeyed.

GO TO RUNOUT also empties the print buffer and spaces 200 lines. It does *not* affect the punch.

4 RESTARTING AFTER RUNOUT

Reading

The TAS 1 control clears the read buffer in Runout, and anything in it will be lost to the program. TAS 2 retains the card in the buffer, and it is this card which will be presented and distributed the next time a READ order is given.

Printing

The first spacing instruction encountered after Runout, like the first of the program, will be ignored.

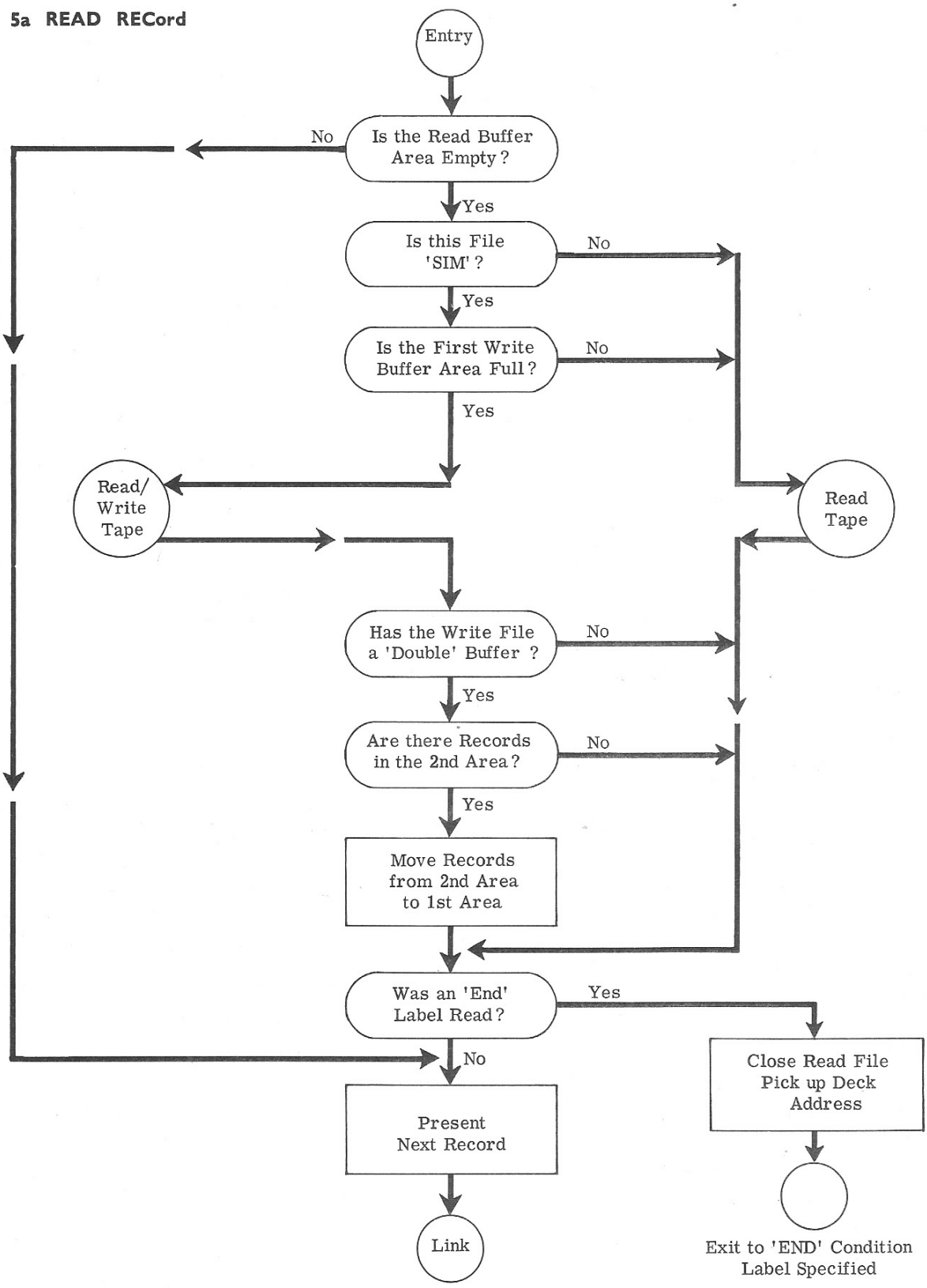
Punching

The punch must not be interfered with in any way.

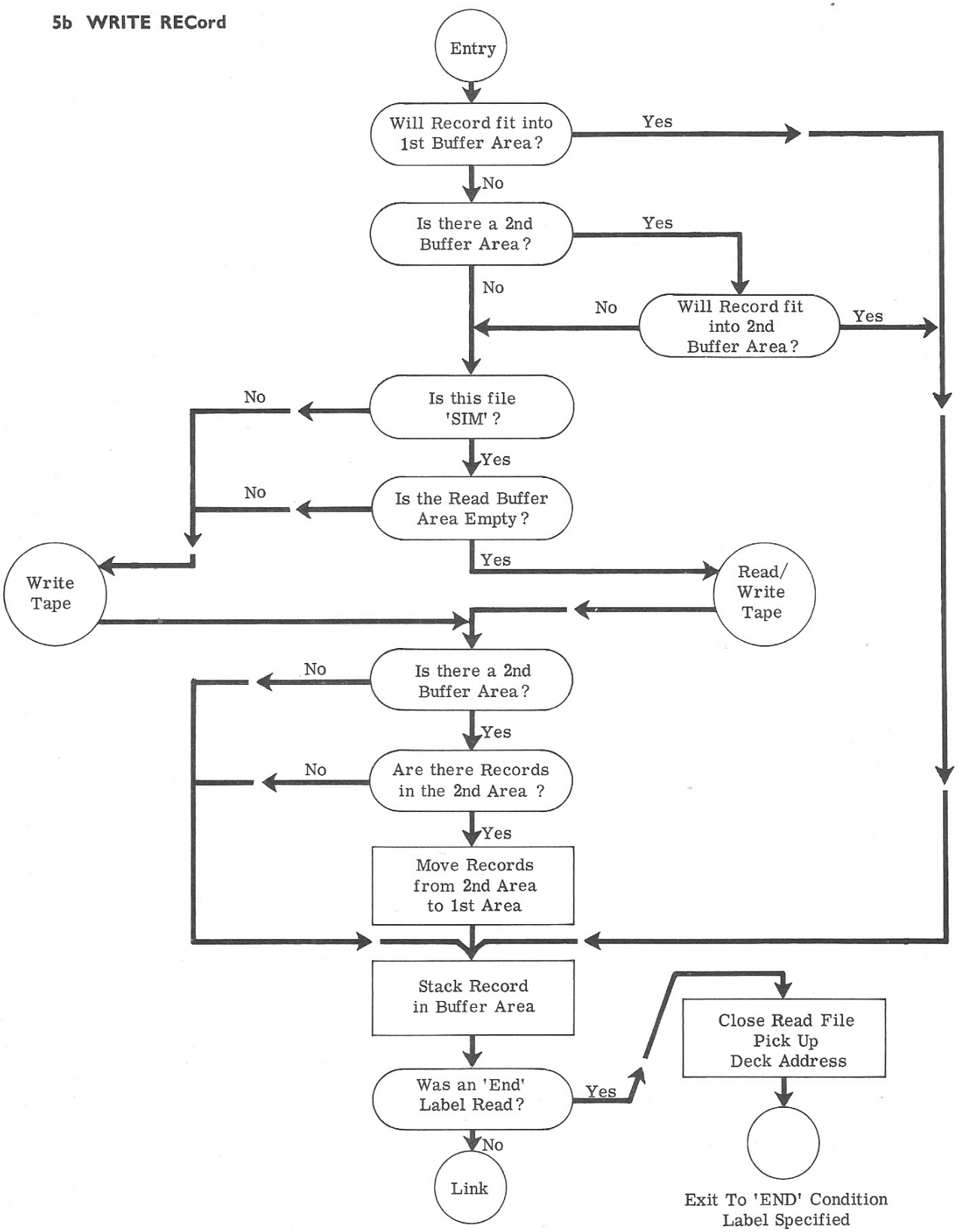
5 TAPE MACROS (TAS 2 ONLY)

The following diagrams may be of some interest and use to experienced programmers in assisting them to force certain desired conditions. An understanding of them is quite unnecessary to handle the verbs correctly.

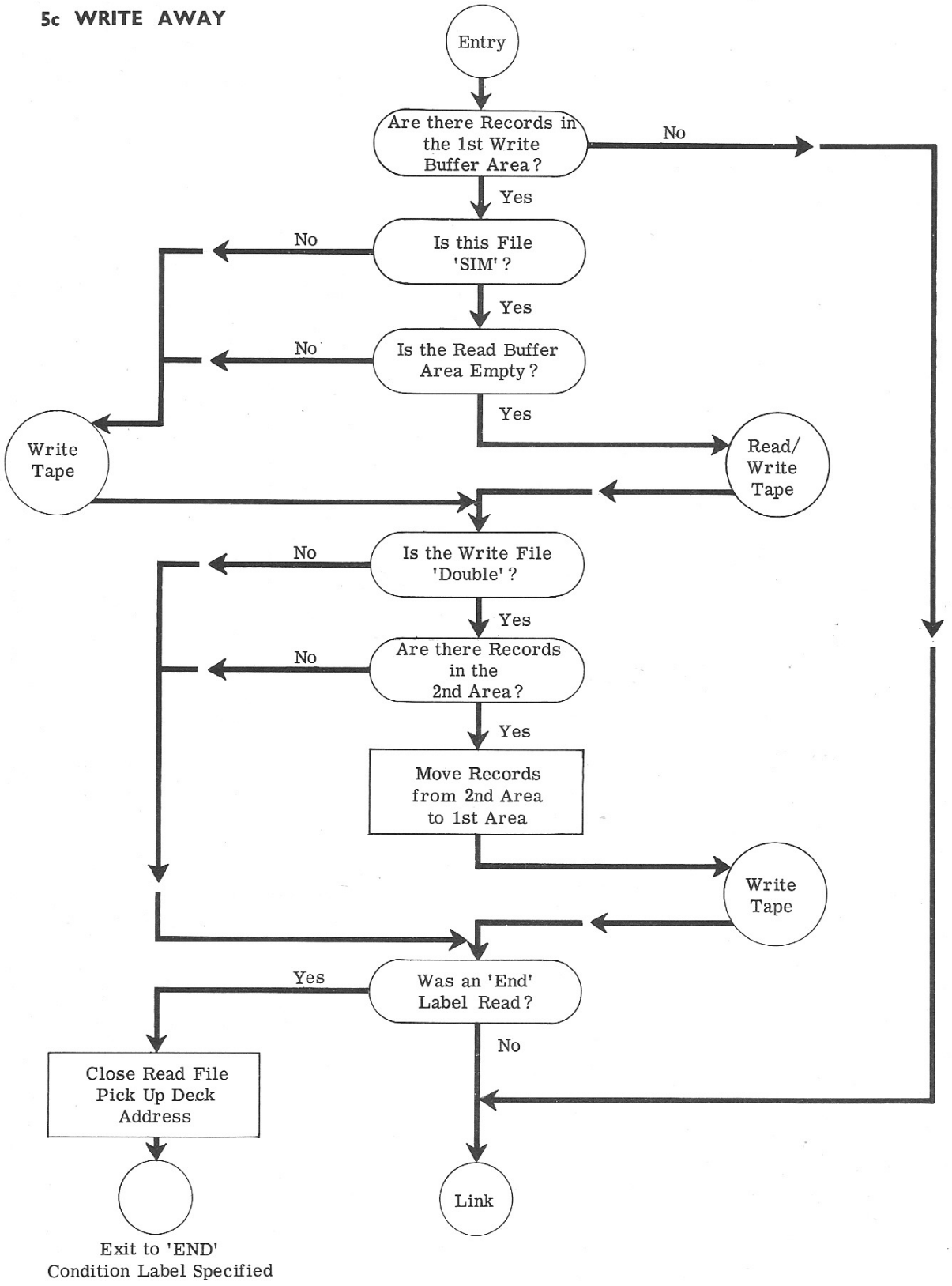
5a READ RECord



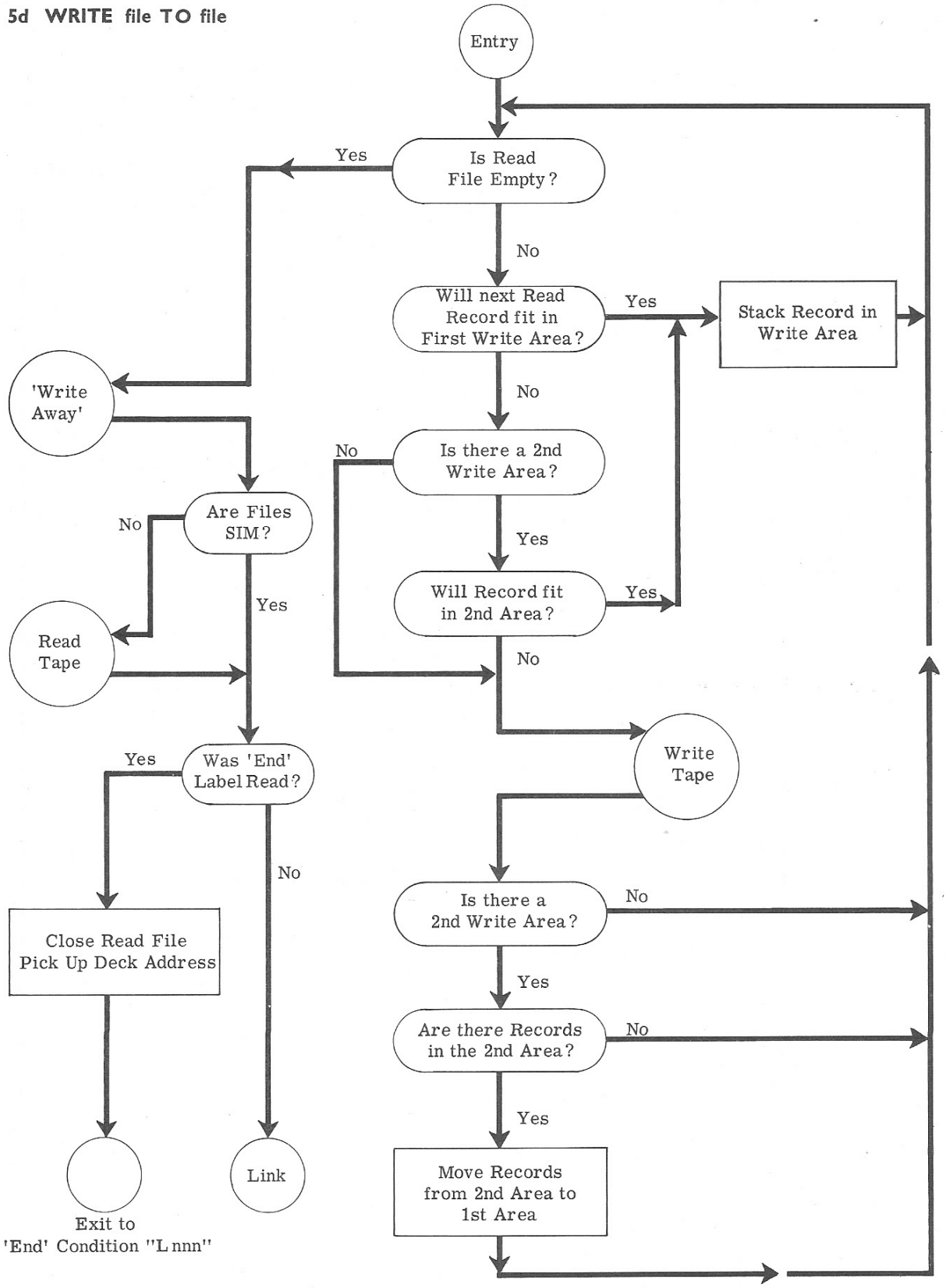
5b WRITE RECord



5c WRITE AWAY



5d WRITE file TO file



Appendix O

COMPILER LOCATIONS

When a precompilation arrives at an impassable error stop, certain information about the program so far accepted is on the drum. Machine code programmers may therefore find the following helpful.

1 Word 0 on (TAS 1 only)

Zones and numerics of Field names so far compiled, held in their correct relative positions. In the first 'free' location is an end marker of twelve $\overline{15}$'s.

2 Words 1600 to 1999

The current block, 'uncompressed', i.e. in TAS 'pseudo-code'. This area is not cleared between each block, so sections of previous longer blocks, not yet overwritten, may also be found here.

3 Words 5000 to 5199

The last block 'compressed', i.e. in machine code.

4 Word 2301 on

Each word holds the following information about one block.

(a) digits 1 to 6: number of words of TAS program compiled

(b) digits 7 to 9: number of words of '1301' compiled

(c) digits 10 to 12: I.A.S. address of last word clear before constants and Working Stores.

From this may be calculated: size of block (200-c+a+b)
available space (c-a-b)

Word 2301 represents block 1, 2302 block 2, and so on.

5 Words 2600 to 3599

Labels list (Word 2600 is label 0, 2601 label 1 etc.).

(a) digits 7 to 9: Block number

(b) digits 10 to 12: I.A.S. address

Appendix P

COMPILER ALLOCATION OF I.A.S. STORAGE SPECIFIED BY USER

The principle on which the compiler allocates I.A.S. storage is that of 'first come, first served'. As each source program card is read in, the storage referred to on it is investigated, and any 'new' area found is given an appropriate allocation.

The first cards the compiler encounters are Table cards, with which we are not concerned. Then come File Descriptions, and, as each File is met, storage is given to it.

Next are Record Format cards, and the Fields described on these are allocated areas in the order in which they appear on the cards. As each Record itself must form a continuous 'block' of words, no name on an 'original' Record may be duplicated.

So far, because of this 'block'ing requirement, all names encountered are new to the compiler, and all (except those in 'Redefining' or 'Sharing' Records) are therefore allocated separate storage.

A Field is allocated storage appropriate to the source program statements in which it appears. When the compiler first encounters each Field, it allocates an initial amount of storage as required by the statement, starting from the first I.A.S. word left free from previous allocations. For instance, should an instruction read

'MOVE (the contents of) 5 (words of I.A.S.) FROM A TO B'

the name, if new to the compiler, will be allocated five words of storage.

On each subsequent encounter the compiler checks to determine whether the Field as now defined overruns the I.A.S. storage so far allocated. If it does not, no extra allocation is made, but if it does, then additional storage is allocated such that any further Field defined will be clear of the 'lengthened' Field.

Take the following compilation sequence:

1 A encountered for the first time defined as being 1 word long.
2 B encountered for the first time defined as being 3 words long.
3 C encountered for the first time defined as being 4 words long.
4 A encountered again defined as being 7 words long.
5 B encountered again defined as being 10 words long.
6 A encountered again defined as being 20 words long.
7 D encountered for the first time defined as being 6 words long.

Suppose the first I.A.S. word free to be I.A.S. 800, the compiler will allocate:

1 Word 800 to A.
2 Word 800-803 to B.
3 Word 804-807 to C.
4 Nothing. A as defined would occupy words 800-806, all of which are by now allocated.
5 Words 808-810 to 'lengthen' B to 10 words.
6 Words 811-819 to 'lengthen' A to 20 words.
7 Words 820-825 to D.

Notice that sequences 4, 5 and 6 do not affect any actual I.A.S. addresses, which have been fixed previously. They merely reserve extra storage if necessary.

Notice also that if, in the MOVE instruction cited above, A had been met previously in an instruction involving one word only, then the order would transfer to B the contents not only of A, but also of whatever Field or Fields occupy the succeeding four words in I.A.S. For this reason, care should be taken over the way in which the compiler first encounters Fields of variable length.

FORCING STORAGE ALLOCATION

It is often possible to improve considerably the efficiency of a program by forcing the compiler to allocate certain Fields contiguous storage. This may be done in several ways.

In TAS 2 the simplest method is to define the Fields concerned as a dummy 'Record'. This may be done whether the program is tape or non-tape. M.I. 22 inhibits the use of File descriptions and Tape macros, but still permits Record definitions.

In TAS 1 similarly, a 'dummy' output format may be defined (see example below).

Example:

FORCING CONSECUTIVE STORAGE BY DEFINING AN OUTPUT CARD

Word 1	Department Number
Word 2	Man Number
Words 3-6	Name
Word 7	Tax Code
Word 8	Gross Pay
Word 9	Tax Paid
Word 10	Deductions
Word 11	National Insurance
Word 12	Net Pay
Word 13	Hourly Rate
Word 14	Holiday Designation

'Output Card'

DES 1 2	TYPE 3 8	KEY	FIELD NAME	CARD No. COLS 69-74
0 3	C A R D 0 8	D 0 1	D E P T	
		D 0 2	M A N N O	
		A 0 3 4	N A M E	
		D 0 4	C O D E	
		S 0 5	G R O S S	
0 3	C A R D 0 8	S 0 6	T A X P D	
		S 0 7	D E D S	
		S 0 8	N A T I N S	
		S 0 9	N E T	
		S 1 0	R A T E	
0 3	C A R D 0 8	D 1 1	H O L D E S	

Note:

1 It is equally possible to define a LINE.

2 It is necessary only to allocate 1 card column for each field, as the compiler always allocates 1 word per field. For alphabetic fields it is necessary to specify the number of words required.

3 The sterling fields could be defined as decimal fields but the sterling designation should be of assistance when writing the program.

4 The Fields specified should not have been defined on previous source program cards, or they will not be allocated storage in the desired form.

5 It is possible to overcome the problem of 'previous definition' by defining a record as an input card; but the risk will then be run that, should a data card having the designation specified be read in (even accidentally), the record will be immediately overwritten.

Appendix Q

DUMPING OR REPOSITIONING

If the "WRITE DUMP" macro is used or, in the event of a tape deck failure, Repositioning is required, a special table must be defined by the user for use by the compiler.

The table which may be given any name the user chooses, must be the first one specified in the program. This is necessary to fix its address relative to the area used by the TAS Control Program.

The size of the table is dependent on the size of tape blocks being used. If it is used for the "WRITE DUMP" macro, it varies with the size of the longest block specified in a Write Dump instruction.

BLOCK SIZE	TABLE LENGTH
6 to 199	400
200 to 399	600
400 to 449	800

If the table is to be used by Repositioning, the size is dependent on the longest block defined in the file descriptions.

BLOCK SIZE	TABLE LENGTH
6 to 199 words	200 words
200 to 399 words	400 words
400 to 449 words	600 words

If the table is used by both Write Dump and Repositioning, it must be equal to the larger of the sizes required according to the above formulae.

The table may be used by the main program but it must be remembered that its contents will be destroyed if a Write Dump instruction is used or a deck break-down necessitates the use of Repositioning.

It is not possible to use Repositioning should there be a deck break-down during the performance of a Write Dump instruction.

Appendix R

STANDARD STOPS DURING OBJECT PROGRAM RUNNING

ALL PROGRAMS

Stop	*Reason*	*Relevant Information*	*Restart Procedure*
11 0303	Card misfeed.		Replace card, and press Start.
11 0311	Card reader error.		Replace rejected card, and press Start.
11 0333	Card mis-read, *or* misfeed.		Replace last card read (may be rejected), and press Start.
11 0411	Card punch error.		Press Start. (Incorrect cards will be offset in the stacker.)
11 0601	Paper trolley empty.		Load more stationery, and press Start.
11 0603	Printer error.		Mark printout, and press Start.
11 0611	Printer error.		Load more stationery, and press Start.
11 0613	Printer error.		Go to restart point.
11 1006	I.A.S. parity.		Restart according to programmer's instructions.
11 1007	Drum parity.		Press Start.
11 1008	Divisor of zero specified.	Rounded quotient is zero, remainder is original dividend.	Press Start.

TAPE PROGRAMS (1 inch and $\frac{1}{2}$ inch Magnetic Tape Systems)

Stop	*Reason*	*Relevant Information*	*Restart Procedure*
11 0801 and 11 0802	Tape deck addressed not under computer control. (Occurs if a deck is not queued on a multi-reel File, or deck has not been seized.)	Deck address in digit 6 of Register B.	If the rewind time is to be occupied, set M.I.29, and press Start. If the rewind time is not to be occupied, load the next reel, unset M.I.29, and press Start. (The stop will be repeated if the deck has not been seized.)
11 0803	Tape deck addressed not under computer control.	Deck address in digit 6 of Register B.	Correct the situation, and press Start.
11 0804	Tape deck breakdown during the reading or writing of a Beginning of Tape Label.	Deck address in digit 6 of Register B.	Load the tape onto a serviceable deck, and press Start.
11 0805	Tape deck breakdown during repositioning.	Deck address in digit 6 of Register B.	Load the tape onto a serviceable deck, and press Start.
11 0806	Tape deck breakdown while reading or writing a block other than a Beginning of Tape Label.	Deck address in digit 6 of Register B.	If a repositioning is to be attempted, load the tape onto a serviceable deck, unset M.I.27, and press Start. If a restart from a dump point is to be made, set M.I.27 and press Start. (Stop 11 0819 will then occur.)
11 0807	Writing a dump block unsuccessful after nine attempts.	Dump deck address in digit 6 of Register B.	If a further 9 attempts are to be made, set M.I.29 and press Start. (The stop will be repeated should these attempts also fail.) If the deck is to be changed, load the tape onto another deck, unset M.I.29, and press Start. Should these actions prove unsuccessful, restart according to programmer's instructions.
11 0808	Tape deck breakdown during dumping.	Deck address in digit 6 of Register B.	Load the tape onto a serviceable deck, and press Start.
11 0811	Writing ring not fitted to a write tape, *or* writing ring fitted to a read tape.	Deck address in digit 6 of Register B.	Correct situation and press Start.
11 0812	Tape not positioned at the beginning of the reel when the repositioning routine entered.	Deck address in digit 6 of Register B.	Reload, positioning tape correctly, and press Start.
11 0813	Incorrect tape fitted for repositioning.	Deck address in digit 6 of Register B.	Load the correct tape, and press Start.

Stop	Reason	Relevant Information	Restart Procedure
11 0814	Incorrect tape fitted.	Deck address in digit 6 of Register A. Digit 12 of Register B is: 0 if the reel number is incorrect. 1 if the file identity is incorrect. 9 if the period number is incorrect.	Load the correct tape, and press Start.
11 0815	Purge date not exceeded.	Deck address in digit 6 of Register A. Purge date in digits 7 to 12 of Register B.	Load an acceptable tape, and press Start.
11 0818	Restart completed.		Restart according to programmer's instructions.
11 0819	Serious error condition.		Restart according to programmer's instructions.
11 0820	Dump completed.	Serial number of the dump reel in Register B. Deck address in digit 6 of Register C. Dump number in digits 10 to 12 of Register C.	Record the state of the console and peripherals, and press Start.
11 0821	Writing a Beginning of File or Beginning of Reel Label unsuccessful after 9 attempts.	Deck address in digit 6 of Register B.	If a further 9 attempts are to be made, set M.I.29 and press Start. (The stop will be repeated should these attempts also fail.) If the tape and/or deck are to be changed, reload, unset M.I.29, and press Start.
11 0822	Writing a tape data block unsuccessful after 9 attempts.	Deck address in digit 6 of Register B.	If a further 9 attempts are to be made, set M.I.29 and press Start. (The stop will be repeated should these attempts also fail.) If the reel is to be closed, unset M.I.29, set M.I.28, and press Start. (The block being written will be written to the next reel.) If the deck is to be changed, unset M.I's 27, 28, and 29, load the tape to the new deck, and press Start. If a restart from a dump point is to be made, set M.I.27, unset M.I's 28 and 29, and press Start. (Stop 11 0819 will occur.)

Stop	Reason	Relevant Information	Restart Procedure
11 0823	Writing an End of Reel or End of File Label unsuccessful after 9 attempts.	Deck address in digit 6 of Register B.	If a further 9 attempts are to be made, set M.I.29 and press Start. (The stop will be repeated should these attempts also fail.) If the deck is to be changed, unset M.I's 27 and 29, load the tape onto the new deck, and press Start. If a restart from a dump point is to be made, set M.I.27, unset M.I.29, and press Start. (Stop 11 0819 will then occur.)
11 0824	Write Master Indicator set, but no associated indicator set.	Deck address in digit 6 of Register B.	Call an engineer. When the fault has been corrected: If the tape has *not* been disturbed, press Start. If the tape has been disturbed, go to restart point.
11 0831	Reading a Beginning of Reel Label unsuccessful after 3 attempts.	Deck address in digit 6 of Register B.	If a further 3 attempts are to be made, set M.I.29 and press Start. (The stop will be repeated should these attempts also fail.) If the deck is to be changed, load the tape onto the new deck, unset M.I.29, and press Start.
11 0832	Reading a tape block unsuccessful after 3 attempts.	Deck address in digit 6 of Register B.	If a further 3 attempts are to be made, set M.I.29 and press Start. (The stop will be repeated should these attempts also fail.) If the deck is to be changed, load the tape onto the new deck, unset M.I.29, and press Start.
11 0833	Reading a tape block, other than a Beginning of Reel Label, unsuccessful after 3 attempts.	Deck address in digit 6 of Register B.	If a further 3 attempts are to be made, set M.I.29 and press Start. (The stop will be repeated should these attempts also fail.) If the deck is to be changed, unset M.I's 27 and 29, load the tape onto the new deck, and press Start. If a restart from a dump point is to be made, set M.I.27, unset M.I.29, and press Start. (Stop 11 0819 will then occur.)
11 0834	Read Master Indicator set, but no associated indicator set.	Deck address in digit 6 of Register B.	As for Stop 11 0824.
11 0835	End of Tape Marker detected on reading.	Deck address in digit 6 of Register B.	Press Start. If repeated continuously a faulty Short Block Indicator or End of Tape Indicator is indicated. Call an engineer.

Stop	Reason	Relevant Information	Restart Procedure
11 0841	End of error condition operation.		Restore Manual Indicators to normal state, and press Start.
11 0842	Tape multiple errors count equal to or greater than maximum permitted.	Deck address in digit 6 of Register A. Multiple Errors Count in digits 1 to 6 of Register B. Single Errors Count in digits 7 to 12 of Register B.	Note contents of Registers A and B, and stop number, and press Start.
11 0843	Tape data block count or hash total discrepancy.	Deck address in digit 6 of Register B.	This stop suggests either tape or computer failure. Restarting from a dump point may be effective; but a thorough investigation is suggested.
11 0858	Tape transport addressed not mechanically ready.		Rectify, and press Start.
11 0918	Rewind failure during 'Renewal'.		Restart from dump point.
11 0919	Wrong reel of file fitted.	Deck address *plus one* in digit 12 of Register A. Reel number in Register B.	Load first reel of file, and press Start.
11 0920	File addressed not under program control.		No restart possible. This stop can be arrived at only through a logical error in the program.

TAPE PROGRAMS ($\frac{1}{4}$ inch Magnet Tape Systems)

Stop	Reason	Relevant Information	Restart Procedure
11 1801	Deck found to be not ready for use when opening a new reel for writing.	Deck address in digit 6 of Register B.	If deck is broken down, a new deck address should be encoded in digit position 12 of Register C, and M.I.26 set. Press Start. If queued deck is not available and unloading time is to be occupied, set M.I.29. Otherwise wait until the deck has been fitted with new tape and re-allocated. Press Start.
11 1802	As 11 1801 stop but during reading tape.	Deck address in digit 6 of Register B.	As above.
11 1803	Deck break-down during writing of initial label.	Deck address in digit 6 of Register B.	If deck easily repaired or re-allocated, tape should be rewound manually and M.I.29 set. Otherwise restart from previous dump point.
11 1804	Deck break-down during reading of initial label.	Deck address in digit 6 of Register B.	New deck address (or present one if desired) should be encoded in digit 12 of Register C and M.I.27 unset, unless restart from a dump point is required.
11 1805	Tape deck break-down during repositioning.		If repositioning to be restarted on same deck, unset M.I.26 and press Start. If a different deck is to be used, encode new deck address in digit 12 of Register C, set M.I.26 and press Start.
11 1806	Tape deck break-down while reading or writing a block.	Deck address in digit 6 of Register B.	If restart from a dump point is to be made, set M.I.27 and press Start. (Stop 11 1819 will then occur.)
11 1808	Transport break-down while repositioning data tape.	Deck address in digit 6 of Register B.	If fault can be corrected immediately, do so and press Start. Otherwise re-load Restart routine and use alternative transports.
11 1811	Writing ring stop.	Deck address in digit 6 of Register B.	Either a writing ring is fitted which should be removed, or one is required but not fitted. Correct the fault and press Start.
11 1813	Incorrect tape fitted for repositioning	Deck address in digit 6 of Register B.	Correct fault and press Start.
11 1814	Incorrect read tape fitted.	Deck address in digit 6 of Register A.	Load correct tape and press Start.
11 1815	Incorrect write tape fitted.	Deck address in digit 6 of Register A.	Load correct tape and press Start.
11 1817	Invalid block detected in dump.	The dump tape will be unloaded.	Restart from a previous dump point.

Stop	Reason	Relevant Information	Restart Procedure
11 1818	Restart completed.		Restart according to programmer's instructions.
11 1819	Serious error condition.		Restart according to programmer's instructions.
11 1820	Dump completed.	Deck address in digit 6 of Register C. Dump number in digits 10 to 12 of Register C.	Record the state of console and peripherals and press Start.
11 1821	Writing a Beginning of File or Reel Label unsuccessful after 9 attempts.	Deck address in digit 6 of Register B.	Check read/write head cover properly shut, set M.I.29 and press Start. If M.I.29 is unset, restart will be made from a previous dump point.
11 1822	Writing a dump block unsuccessful after 9 attempts.	Deck address in digit 6 of Register B.	If a further 9 attempts are to be made, set M.I.29 and press Start. Otherwise restart from a previous dump point.
11 1823	Writing data block unsuccessful after 9 attempts.		If a further 9 attempts are to be made, set M.I.29 and press Start. If restart from a dump point to be made, set M.I.27, unset M.I.28 and M.I.29 and press Start. (Stop 11 1819 will occur.) If tape to be repositioned on a different deck, encode new deck address in digit 12 of Register C, unset M.I's 27 and 29 and press Start.
11 1824	Entry to Write Exceptions routine for no apparent reason.	Deck address in digit 6 of Register B.	If program under test, abandon, and give I.A.S. and tape printouts as requested by programmer. If proved program, call an engineer.
11 1825	Reading dump block unsuccessful after 6 attempts.	Deck address in digit 6 of Register B.	If further 6 attempts are to be made, press Start. Otherwise restart from previous dump point.
11 1826	Deck break-down while reading a dump block.		Reload Restart, possibly using different transports.
11 1827	Incorrect dump tape fitted.	Deck address in digit 1 of Register B. Correct reel serial number in digits 7 to 12.	Load correct tape and press Start.
11 1828	Required dump not on tape fitted.	Deck address in digit 1 of Register B. Dump number in digits 10 to 12.	Reload Restart using the correct tape.
11 1829	Reading unsuccessful after 6 attempts while repositioning data tape.	Deck address in digit 6 of Register B.	If further 6 attempts are to be made, press Start. Otherwise reload Restart, specifying previous dump number or using different transports.

Stop	Reason	Relevant Information	Restart Procedure
11 1830	Manual indicator setting incorrect.		Check M.I's are as at the appropriate dump point and press Start.
11 1832	Reading a block unsuccessful after 6 attempts during repositioning.		If further attempts to be made, set M.I.29 and press Start. If restart to be made, set M.I.27 and press Start. If repositioning to recommence on another deck, encode new deck address in digit 12 of Register C, unset M.I's 29 and 27, set M.I.26 and press Start. If repositioning to recommence on the same deck unset M.I's 26, 27 and 29 and press Start.
11 1833	Reading a block unsuccessful after 6 attempts.		Restart as for any of the first three cases for 11 1832 stop.
11 1834	Entry to Read Exceptions routine for no apparent reason.		See restart procedure for 11 1824 stop.
11 1841	End of error condition operation.		Reset M.I's to normal state and press Start.
11 1842	Errors accumulated during writing are greater than permissible.		If reel is to be closed, set M.I.28 and press Start. Otherwise unset M.I.28, and press Start to store errors and exit via link.
11 1843	Data block count discrepancy.		No restart possible.
11 1857	Tape transport address not mechanically ready.	Deck address in digit 2 of Register B.	Rectify, and press Start.
11 1858	Rewind failure during 'renewal'.		Restart from dump point.
11 1859	Wrong reel of file fitted.	Deck address *plus one* in digit 12 of Register A. Reel number in Register B.	Load first reel of file and press Start.
11 1860	File addressed not under program control.		No restart. Can only be arrived at through logical error in program.
11 1910	Short block indicator set while new reel being opened.		Fully rewind tape manually and press Start.
11 1911	No beginning label on new write file.		No restart possible.