



DECUS

PROGRAM LIBRARY

DECUS NO.	FOCAL8-60
TITLE	A SYSTEM FOR PRODUCTION OF PROBLEM SETS WITH INDIVIDUALIZED DATA
AUTHOR	H. Bradford Thompson
COMPANY	Department of Chemistry University of Toledo Toledo, Ohio
DATE	October 1969
SOURCE LANGUAGE	FOCAL

anc

qs

m

m

600-41-11

10/10/10 10:10 AM 10/10/10 10:10 AM

10/10/10 10:10 AM

10/10/10 10:10 AM

10/10/10

10/10/10

A SYSTEM FOR PRODUCTION OF PROBLEM SETS WITH INDIVIDUALIZED DATA

DECUS Program Library Write-up

DECUS No. FOCAL8-60

ABSTRACT

The system produces problem sets for use in science and mathematics instruction, in which input data are changed for each student. Two programs are involved, (1) a FOCAL program into which the instructor inserts the algebra required to perform the calculations, and (2) a program which accepts a text with data positions marked, and then inserts individualized data from the FOCAL program (without the answers) and prints the copies. The system will work on any family-of-8 machine for which FOCAL is available.

1.0 INTRODUCTION

This system enables an instructor in a course involving calculations to prepare a number of sets of problems for his students. The problems have the same text except for varying input data. The type and complexity of problem possible is limited only by the imagination of the instructor and the very considerable capacity of the FOCAL programming system.

2.0 SYSTEM REQUIREMENTS

2.1 Hardware: Required: Basic PDP-8/I or equivalent plus ASR 33.

Useful: Extended memory. (See Section 4.3)

2.2 Software: Required: FOCAL (Any version having a random number function (but see sec. 4.1).

Useful: Symbolic Editor; 8K FOCAL.

3.0 PROGRAM DESCRIPTIONS

Two programs are involved, a FOCAL program which calculates data and answers, and a Text-Assembly Program which introduces the data into a prepared text. The FOCAL program produces a tape bearing both data and answers. The Text-Assembly Program (TAP) discards from this tape everything but the answers, which it inserts into copies of the text printed on the teletype. Detailed descriptions of the two programs follow. Examples are presented in Section 5.

3.1 Problem-Set Generator (FOCAL)

3.1.1 Description: The FOCAL program consists of two sections (1 and 2) of basic text, plus statements prepared by the user for particular problems. Section 1 controls over-all program flow, determines the number of data sets desired, punches out leader, provides the loop containing the user's prepared statements, punches a terminating clue, punches trailer, then quits. Section 2 is essentially a random-number generator (See 4.1), producing a random number HI and a complementary value LO=1-HI.

3.1.2 Problem Preparation: The user must supply for the FOCAL program the limits to be set on the input variables, the algebraic and logical statements by which answers are obtained, and statements which print a table and punch a tape containing data and answers. The user-prepared statements must begin with section 3, and from there can use all of the higher section numbers allowed by the user's version of FOCAL.

3.1.2.1 Random Data Generation. For each input variable, the user prepares a statement of the form

3.xx D0 2; SET X=HI*(upper limit)+LO*(lower limit)

thus producing a random value of X lying between these limits.

3.1.2.2 Answer Determination. Most commonly, the user will simply calculate each answer by SET statements. For complex problems, he has at his disposal the full arithmetic and logical power of FOCAL.

3.1.2.3 Data Output. The format of the output table must contain those clues required by the program TAP to distinguish data from format characters and answers. The basic requirement is quite simple: each item of data must be preceded by an = (as is normal in FOCAL), and each answer must be preceded by the letter A ahead of the = . The symbols A and = should not occur elsewhere on the tape. Further specifics regarding tapes are given in section 3.2.4. The symbols : and ; should not be used immediately following the last digit of a datum, since they serve as special clues. They may be used elsewhere on a data tape. A typical data tape generation statement might be:

4.xx TYPE %5.02, I, ?X?, ? Y?, ? Z?, " A", V

where X, Y, and Z are data to be inserted in the problem texts, while the answer V will be printed in the FOCAL output, but omitted from the texts.

The user may, of course, choose different formats for different numbers. He may output the data and answers intermixed, and have any number of data and answers, provided that (1) the data are typed in the order they are to appear in the text, and (2) each answer is preceded by an A. (Two or more A's will cause the program TAP to skip the corresponding number of output numbers.)

3.1.2.4 Program Testing. The algebraic algorithm and answer printing sections can easily be tested with data giving known answers if the program is written with this in mind. Most simply, if all the random data generation statements are included in section 3, followed by a "DO 4" statement, section 4, containing the calculation and output sections can be tested on-line. The test input data can be specified by an on-line (i.e., unnumbered) SET statement, followed by an on-line GOTO 4. An example of this technique is given in section 5.1.

3.1.3 Usage: After preparation of the required FOCAL sections, the program is started by a FOCAL GO command. In response to NO. OF SETS DESIRED: the user gives the number of individualized data sets wanted. The paper tape punch should be turned on immediately after typing the terminating character (space or carriage return) following this number.

The program will immediately begin turning out problem set data and answers. The punched tape will consist of

Leader: spaces (Code 240)
Data and Answers
A termination clue (= 99999:)
Trailer: spaces

The colon(:) terminating a number will halt the Text Assembly Program.

3.2 Text-Assembly Program

3.2.1 Description: TAP is a PAL III program which accepts a text tape in which the positions reserved for answers are indicated by @, then reads in the data tape from the FOCAL Problem Set Generator, abstracts the data, and inserts these values in copies of the text produced on the teletype.

3.2.2 Text Preparation: The text is printed using the DEC program DEC-1S-U. Allowed symbols are:

Letters A to Z Digits 0 to 9 Symbols !"\$%&'()*+,-./:;<=>?

Other symbols will cause errors if included as text. Particular note should be taken that the symbols % and + cannot be used. The symbol @ is to be used only at the point where a datum is to be inserted. (The Form Feed tape symbol will also cause a break for a datum.) Text tape can easily be prepared using the DEC Symbolic Editor. Leader tape should be blank (produced by HERE IS). (Leader with the code 200 will produce a string of Y's.)

3.2.3 Usage: TAP is provided as a binary tape loaded in the usual manner with the BIN loader. It has two starting points, one for text readin and one for data readin and text output.

3.2.3.1 Text Readin.

1. Insert the text tape in the tape reader and turn on the reader.
2. Set SR to 0200, LOAD ADDRESS, START. Text will read in up to first @ symbol.
3. Press CONTINUE to read in subsequent text segments.

3.2.3.2 Data Readin, Text Output.

1. Insert data tape in the tape reader and turn on the reader.
2. Set SR to 0220, LOAD ADDRESS, START. Check first two or three text copies, then go to coffee.

3.2.4 Summary of Allowed Symbol Usage: The table below gives the allowed symbol usage on text tapes and in two different cases on data tapes. "Search Mode" means the data tape is being inspected for the letter A and the symbol =. "Number Mode" means an = sign has been encountered, and the data tape is being inspected for a datum.

<u>Symbols</u>	<u>In Text</u>	<u>On Data Tape</u>	
		<u>Search Mode</u>	<u>Number Mode ***</u>
Letter A	Typed	Skip one = *	Terminate
Letter E	Typed	Ignored	Typed
B, C, D, F thru Z	Typed	Ignored	Terminate
+ - .	Typed	Ignored	Typed
Digits 0 - 9	Typed	Ignored	Typed
=	Typed	Change Mode	Terminate
Space	Typed	Ignored	**
@	End Segment	Ignored	Terminate
:	Typed	Ignored	Halt Program
;	Typed	Ignored	Terminate Set
Other Symbols	See 3.2.2	Ignored	Terminate

* Each time an A is encountered in Search Mode a counter is incremented which causes one = to be ignored. Thus "AAA" would cause the next three = to be ignored, and could cause the next three punched numbers to be bypassed.

** Spaces in Number Mode are ignored if they follow the = and precede any other symbol. Otherwise, spaces terminate the number.

*** When Number Mode is terminated, the program returns to the next segment of text, and the terminating symbol is saved. When the next @ is encountered, the terminating symbol is recalled, and if it is an A or an =, it is used as indicated above under Search Mode. However, if the end of the text is encountered, the saved symbol is discarded.

When the terminating symbol is : the program halts. When the terminating symbol is ; the rest of the text is skipped and a new problem set is begun.

4.0 NOTES

4.1 Generation of the Random Number HI.

The FOCAL function FRAN is inadequate in two versions tested. In FOCAL.W (1968) FRAN had an annoying habit of repeating a value several times. In FOCAL, 1969, FRAN was apparently bounded in the rather unusual interval 0.5 to 1.0, and occurred nearly twice as often at the lower end as at the upper. In either case, the simple definition of HI in terms of FRAN given in statement 2.1 of this program produces a random number much better suited to this program.

If a version of FOCAL without FRAN is employed, statement 2.1 may be used with FRAN () replaced by some arbitrary six-number fraction.

4.2 Timing:

TAP is I/O bound by reader and printer speeds. A large number of copies will require essentially the teletype time needed to type them. The FOCAL program will be similarly limited by printer-punch speed unless the answer calculation is quite complex.

4.3 Versions and Modifications Tested:

The Program Generation System has been tested using FOCAL.W (1968) and FOCAL, 1969 in the 4K versions, and FOCAL, 1969 with the 8K patch. A standard tape has been constructed including the 8K patch and FWAT (see above), which is the present working version at Toledo. 8K FOCAL allows for a very extended set of problems, and is ideal for this system. However, the 4K version functions in a very satisfactory manner.

A modification using a fast tape reader and punch is planned.

		/KEYBOARD READ	
		*2	
0002	0000	READ,	0
0003	6031		KSF
0004	5003		JMP .-1
0005	6036		KRB
0006	5402		JMP I READ
		/ALPHAMERIC MESSAGE INPUT (TO MATCH DEC 8-18-U	
		/CALL: JMS ALPHIN	
		/ (ADDRESS)	
		/ RETURN HERE	
		*20	
0020	0000	ALPHIN,	0
0021	7320		CLA STL
0022	1420		TAD I ALPHIN
0023	3071		DCA ADR
0024	2020		ISZ ALPHIN
0025	4002	READIN,	JMS READ
0026	6046		TLS
0027	7450		SNA
0030	5025		JMP READIN
0031	1073		TAD CONST
0032	7450		SNA
0033	5061		JMP CR
0034	7001		IAC
0035	7450		SNA
0036	5063		JMP TSYMR
0037	7510		SPA
0040	5061		JMP CR
0041	1074		TAD CONST+1
0042	0075		AND MASK77
0043	7450		SNA
0044	5063		JMP TSYMR
0045	7430	JOIN,	SZL
0046	5054		JMP RIGHT
0047	7106		CLL RTL
0050	7006		RTL
0051	7006		RTL
0052	3072		DCA CHAR
0053	5025		JMP READIN
0054	1072	RIGHT,	TAD CHAR
0055	3471		DCA I ADR
0056	2071		ISZ ADR

	/OBTAIN STORAGE ADDRESS
	/READ A CHARACTER
	/BLANK TAPE?
	/YES. IGNORE LEADER
	/NO
	/CR?
	/YES
	/NO
	/FORM FEED?
	/YES
	/ NO. LESS THAN 214?
	/YES. ASSUME LINE FEED
	/NO
	/OBTAIN 8-BIT CODE
	/0?
	/YES
	/RIGHT OR LEFT 8 BITS?

0057	7120		STL	
0060	5025		JMP READIN	
0061	1070	CR,	TAD CRCHAR	/GENERATE DEC 8-18-U
0062	5045		JMP JOIN	/CR AND LF CODES
0063	7430	TSYMR,	SZL	/TERMINATE THIS INPUT
0064	1072		TAD CHAR	
0065	3471		DCA I ADR	
0066	1071		TAD ADR	
0067	5420		JMP I ALPHIN	
0070	0045	CRCHAR,	45	
0071	0000	ADR,	0	
0072	0000	CHAR,	0	
0073	7563	CONST,	-215	
0074	0214		214	
0075	0077	MASK77,	77	
0076	0000	CHARIN,	0	
		*175		
0175	0020	QALFIN,	ALPHIN	/PLANT ADDRESS

ADR	0071
ALPHIN	0020
CHAR	0072
CHARIN	0076
CONST	0073
CR	0061
CRCHAR	0070
JOIN	0045
MASK77	0075
QALFIN	0175
READ	0002
READIN	0025
RIGHT	0054
TSYMR	0063

5.0 EXAMPLES

5.1 Gas Law Calculation: In this simple problem the number of moles of an ideal gas are calculated, given the pressure, volume, and temperature

*C SAMPLE GAS LAW PROBLEM

*

*W

C-FOCAL,1969

01.20 ASK ! "NO. OF SETS DESIRED", SQ

01.25 T !; F NQ=1,70; T " "

01.30 FOR NQ=1,SQ; DO 3

01.40 T %6.00, 99999, ":",D 1.25;Q

02.10 S HI=67*HI+FRAN(); S HI=HI-FITR(HI); S LO=1-HI

03.10 DO 2; SET P=.8*LO+1.2*HI

03.20 DO 2; SET V=22*LO+25*HI

03.30 DO 2; SET T=288*LO+295*HI

03.40 DO 4

04.10 SET N=P*V/(.08206*T)

04.20 TYPE %6.03,! ? P?, ? T?, ? V?, " A", N

*

*C TEST RUN WITH CHOSEN INPUT DATA

*SET P=1; SET V=22.4; SET T=273

*GOTO 4.1

P= 1.000 T= 273.000 V= 22.400 A= 1.000*

*

*C DATA RUN

*GO

NO. OF SETS DESIRED:10

P=	0.882	T=	293.867	V=	24.160	A=	0.884
P=	1.151	T=	292.377	V=	24.109	A=	1.157
P=	1.015	T=	289.675	V=	24.565	A=	1.049
P=	1.046	T=	290.498	V=	24.575	A=	1.078
P=	0.971	T=	291.238	V=	22.658	A=	0.920
P=	1.172	T=	293.041	V=	24.554	A=	1.197
P=	0.835	T=	291.331	V=	23.247	A=	0.812
P=	1.053	T=	294.756	V=	23.048	A=	1.004
P=	0.933	T=	290.606	V=	22.241	A=	0.870
P=	1.017	T=	288.414	V=	24.963	A=	1.072= 99999:

*

SAMPLE PROBLEM PRODUCED WITH INDIVIDUALIZED DATA ON THE PDP-8/I

HOW MANY MOLES OF AN IDEAL GAS AT A PRESSURE OF @ ATM
AND A TEMPERATURE OF @ DEGREES KELVIN WILL BE REQUIRED TO FILL
A @ LITER CONTAINER?

@

SAMPLE PROBLEM PRODUCED WITH INDIVIDUALIZED DATA ON THE PDP-8/I

HOW MANY MOLES OF AN IDEAL GAS AT A PRESSURE OF 1.053 ATM
AND A TEMPERATURE OF 294.756 DEGREES KELVIN WILL BE REQUIRED TO FILL
A 23.048 LITER CONTAINER?

SAMPLE PROBLEM PRODUCED WITH INDIVIDUALIZED DATA ON THE PDP-8/I

HOW MANY MOLES OF AN IDEAL GAS AT A PRESSURE OF 0.933 ATM
AND A TEMPERATURE OF 290.606 DEGREES KELVIN WILL BE REQUIRED TO FILL
A 22.241 LITER CONTAINER?

SAMPLE PROBLEM PRODUCED WITH INDIVIDUALIZED DATA ON THE PDP-8/I

HOW MANY MOLES OF AN IDEAL GAS AT A PRESSURE OF 1.017 ATM
AND A TEMPERATURE OF 288.414 DEGREES KELVIN WILL BE REQUIRED TO FILL
A 24.963 LITER CONTAINER?

SAMPLE PROBLEM PRODUCED WITH INDIVIDUALIZED DATA ON THE PDP-8/I

HOW MANY MOLES OF AN IDEAL GAS AT A PRESSURE OF 99999

5.2 Gas Sample Collected over Water: This example, provided by Dr. Duane F. Burow, uses two modifications worthy of attention.

```

*****
*C-FOCAL,1969
*
*01.20 ASK ! "NO. OF SETS DESIRED", SQ
*01.22 FOR TC=10,40 ; TYPE %2.00,?TC?;ASK ?PT(TC)?
*01.25 T !; F NQ=1,70; T " "
*01.30 FOR NQ=1,SQ; DO 3
*01.40 T %6.00, 99999, ":";D 1.25;Q
*
*02.10 S HI=67*HI+FRAN(); S HI=HI-FITR(HI); S LO=1-HI
*
*03.10 DO 2; SET X=FITR(10.0*LO+300.0*HI)
*03.20 DO 2; SET Y=FITR(700*LO+750*HI)
*03.30 DO 2; SET TC=FITR(10*LO+40*HI)
*03.31 SET PW=PT(TC)
*03.40 DO 4
*
*04.10 SET V=<273*X*(Y-PW)>/((TC+273)*760
*04.20 TYPE %3.00,! ?NQ?, ? X?, ? Y?, ? TC?, " A", V
*****
*

```

Note that the user wished to read in a table (vapor pressures of water at specific temperatures), and so introduced the line 01.22. He also desired to round off all data, and used the FITR function for this purpose.

Note also how Dr. Burow obtained problem sets with consecutive numbering by including " ?NQ? " in his TYPE statement.

6.1 Problem-Set Generator (FOCAL):

```

*****

```

```

C-FOCAL , 8/68

```

```

01.20 ASK ! "NO. OF SETS DESIRED", SQ

```

```

01.25 T !; F NQ=1,70; T " "

```

```

01.30 FOR NQ=1,SQ; DO 3

```

```

01.40 T %6.00, 99999, ":";D 1.25;Q

```

```

02.10 S HI=67*HI+FRAN(); S HI=HI-FITR(HI); S LO=1-HI

```

```

*****

```

6.2 Text-Assembly Programs: An ASCII tape of TAP is available. It consists of three segments, a master segment at 0200, a version of DEC-8-18-u at 0100, and a text-input subroutine starting at 0006. Core from 0367 on is a text buffer, capable of holding approximately 7200 characters.

/PROBLEM SET TYPEOUT PROGRAM

/TEXT READIN SEGMENT

*200

0200	6046		TLS	
0201	3213		DCA SEGCNT	/CLEAR SEGMENT COUNT
0202	1215		TAD BUFFER	
0203	4341	INLOOP,	JMS CONALF	/READ A SEGMENT
0204	2213		ISZ SEGCNT	
0205	3214		DCA BUF2	
0206	7402		HLT	/AWAIT ANOTHER SEGMENT
0207	1214		TAD BUF2	
0210	5203		JMP INLOOP	
0211	0000	BUFPTR,	0	
0212	0000	SEG,	0	
0213	0000	SEGCNT,	0	
0214	0000	BUF2,	0	
0215	0367	BUFFER,	BFAREA	

*220

0220	7200		CLA	
0221	3333		DCA CHATMP	/CLEAR CHARACTER HOLDOVER
0222	6046		TLS	
0223	7300	BGN2,	CLA CLL	
0224	1213		TAD SEGCNT	
0225	7041		CIA	
0226	3212		DCA SEG	/PLANT SEGMENT COUNTER
0227	1215		TAD BUFFER	
0230	4361	NXTSEG,	JMS CONTYP	/PRINT A SEGMENT
0231	3211		DCA BUFPTR	
0232	2212		ISZ SEG	
0233	5235		JMP INDIN	/GET A NUMBER
0234	5223		JMP BGN2	
0235	7240	INDIN,	STA	
0236	3331		DCA AKLUE	/INITIALIZE COUNT OF A'S
0237	1333		TAD CHATMP	
0240	1332		TAD M301	
0241	7440		SZA	/TEST FOR A
0242	5250		JMP NOTANS	
0243	7240		STA	/ANSWER SIGNAL RECEIVED
0244	1331		TAD AKLUE	
0245	3331		DCA AKLUE	/INCREMENT COUNT
0246	4324	SEARCH,	JMS READR	/INPUT A CHARACTER
0247	5240		JMP INDIN+3	
0250	1337	NOTANS,	TAD P4	
0251	7450		SNA	/SKIP UNLESS AN =
0252	2331		ISZ AKLUE	/IF =, TEST FOR ANS
0253	5246		JMP SEARCH	
0254	4324	NUMBER,	JMS READR	
0255	3333		DCA CHATMP	
0256	1333		TAD CHATMP	
0257	1334		TAD M240	/SPACE?
0260	7650		SNA CLA	
0261	5254		JMP NUMBER	/SKIP INITIAL SPACES

PAUSE

0262	1333	NJOIN,	TAD CHATMP	
0263	1335		TAD M305	/TEST FOR E
0264	7450		SNA	
0265	5314		JMP DIGPRT	
0266	1340		TAD P12	
0267	7450		SNA	/TEST FOR;
0270	5223		JMP BGN2	/TERMINATE PROBLEM SET
0271	7001		IAC	
0272	7450		SNA	/TEST FOR :
0273	7402		HLT	
0274	7500		SMA	/TEST FOR CODES GT 273
0275	5322		JMP NXTSG2	/ TERMINATE NUMBER
0276	1340		TAD P12	
0277	7500		SMA	/TEST FOR DIGIT
0300	5314		JMP DIGPRT	
0301	7001		IAC	
0302	7450		SNA	
0303	5322		JMP NXTSG2	
0304	1336		TAD P3	/AC=CHAR CODE-254
0305	7450		SNA	
0306	5322		JMP NXTSG2	
0307	7500		SMA	
0310	5314		JMP DIGPRT	/- OR .
0311	7001		IAC	
0312	7640		SZA CLA	/TEST FOR +
0313	5322		JMP NXTSG2	
0314	7200	DIGPRT,	CLA	/PRINT CHAR
0315	1333		TAD CHATMP	
0316	6046		TLS	
0317	4324		JMS READR	
0320	3333		DCA CHATMP	
0321	5262		JMP NJOIN	
0322	1211	NXTSG2,	TAD BUFPTR	
0323	5230		JMP NXTSEG	
0324	0000	READR,	0	
0325	6031		KSF	
0326	5325		JMP .-1	
0327	6036		KRB	
0330	5724		JMP I READR	
0331	0000	AKLUE,	0	
0332	7477	M301,	-301	
0333	0000	CHATMP,	0	
0334	7540	M240,	-240	
0335	7473	M305,	-305	
0336	0003	P3,	3	
0337	0004	P4,	4	
0340	0012	P12,	12	

PAUSE

/CONSECUTIVE ALPHAMERIC ARRAY READ AND TYPE

/CALL: JMS CONALF WITH STORAGE ADR IN AC

/(AC) ON EXIT = FIRST UNUSED ADDRESS

QMESAG=176

QALFIN=175

0341	0000	CONALF,	0	
0342	3347		DCA ALFPTR	
0343	1357		TAD INSTR1	/PLANT PRINTOUT CALL
0344	3747		DCA I ALFPTR	
0345	2347		ISZ ALFPTR	
0346	4575		JMS I QALFIN	
0347	0000	ALFPTR,	0	
0350	7001		IAC	
0351	3347		DCA ALFPTR	
0352	1360		TAD INSTR2	/ PLANT PRINTOUT RETURN
0353	3747		DCA I ALFPTR	
0354	1347		TAD ALFPTR	
0355	7001		IAC	
0356	5741		JMP I CONALF	
0357	4576	INSTR1,	JMS I QMESAG	
0360	4577	INSTR2,	JMS I QALPTR	

/CALL: JMS I CONTYP WITH STORAGE ADDRESS IN AC

/(AC) ON EXIT = NEXT BUFFER ADDRESS

0361	0000	CONTYP,	0
0362	3364		DCA ALPTR
0363	5764		JMP I ALPTR
0364	0000	ALPTR,	0
0365	1364		TAD .-1
0366	5761		JMP I CONTYP
0367	0000	BFAREA,	0
		*177	
0177	0364	QALPTR,	ALPTR

PAUSE

AKLUE	0331
ALFPTR	0347
ALPTR	0364
BFAREA	0367
BGN2	0223
BUFFER	0215
BUFPTR	0211
BUF2	0214
CHATMP	0333
CONALF	0341
CONTYP	0361
DIGPRT	0314
INDIN	0235
INLOOP	0203
INSTR1	0357
INSTR2	0360
M240	0334
M301	0332
M305	0335
NJOIN	0262
NOTANS	0250
NUMBER	0254
NXTSEG	0230
NXTSG2	0322
P12	0340
P3	0336
P4	0337
QALFIN	0175
QALPTR	0177
QMESAG	0176
READR	0324
SEARCH	0246
SEG	0212
SEGCNT	0213

/DIGITAL 8-18-U MODIFIED
/MESSAGE TYPE-OUT

			*100
0100	0000	MESSAGE,	0
0101	7240		CLA CMA
0102	1100		TAD MESSAGE
0103	3010		DCA 10
0104	1410		TAD I 10
0105	3116		DCA MSRGHT
0106	1116		TAD MSRGHT
0107	7012		RTR
0110	7012		RTR
0111	7012		RTR
0112	4117		JMS TYPECH
0113	1116		TAD MSRGHT
0114	4117		JMS TYPECH
0115	5104		JMP MESSAGE+4
0116	0000	MSRGHT,	0
0117	0000	TYPECH,	0
0120	0150		AND MASK77
0121	7450		SNA
0122	5410		JMP I 10
0123	1151		TAD M40
0124	7500		SMA
0125	5130		JMP .+3
0126	1152		TAD C340
0127	5177		JMP MTP
0130	1153		TAD M3
0131	7440		SZA
0132	5135		JMP .+3
0133	1154		TAD C212
0134	5177		JMP MTP
0135	1155		TAD M2
0136	7440		SZA
0137	5142		JMP .+3
0140	1156		TAD C215
0141	5177		JMP MTP
0142	1157		TAD C245
0143	6177	MTP	TSF
0144	5143		JMP .-1
0145	6046		TLS
0146	7200		CLA
0147	5517		JMP I TYPECH

0150	0077	MASK77,	77
0151	7740	M40,	-40
0152	0340	C340,	340
0153	7775	M3,	-3
0154	0212	C212,	212
0155	7776	M2,	-2
0156	0215	C215,	215
0157	0245	C245,	245
			*176
0176	0100	QMESAG,	MESSAGE
		PAUSE	

C212	0154
C215	0156
C245	0157
C340	0152
MASK77	0150
MESSAGE	0100
MSRGHT	0116
MTP	0177
M2	0155
M3	0153
M40	0151
QMESAG	0176
TYPECH	0117