

HP-71 Disassembler

An HP-71 Disassembler

1 Purpose & Design

A. Basic Function

The basic function of a disassembler is to present a block of assembly language code as comprehensible mnemonics which represent known opcodes. As the following paragraphs indicate, there are many features that can be added.

B. Applications

Typical applications of disassemblers might be to discover how HP does their low power key wait function or how new LEX words operate.

C. Where are opcodes stored?

The HP-71 uses four types of non-data files. Two of these contain opcodes that are conveniently shown by a disassembler.

BASIC	No, contains tokens for interpretation.
BIN	Yes, contains opcodes and name chains.
LEX	Yes, contains opcodes and lex tables.
FORTHRAM	No, contains mostly addresses.

The four types of data files, DATA, KEY, SDATA, and TEXT might contain anything, even opcodes, but generally don't.

D. HP-71 architecture necessitates certain features

The HP71 makes extensive use of files and linked lists to provide flexible memory management. The disassembler decodes these for the user's convenience.

- LEX files—For LEX files, the name chain, speed table and lex table are also shown (as macros).
- BIN files—For BIN files, the disassembler shows the name chain constructs as the macros used by the Forth Rom assembler to create them.

E. Specification of code for disassembly

File names

For both LEX and BIN words, the code may be specified by the file name which contains it. This generates a disassembly of all opcode segments within the file.

Hex addresses

For locations not within a standard 71 file type, the hexadecimal starting and ending addresses of the code may be given. This is useful for ROM or Forth words. Of course, the disassembler doesn't make very good sense out of data areas.

F. Output of disassembly

Format of output

The disassembler's output is compatible with the Forth ROM assembler since this aids comprehension and permits reassembly of the output.

Format of each line

Each line of the disassembler output represents one assembly language instruction and looks like:

address	mnemonic	arguments (in user radix)	ASCII representation.
---------	----------	------------------------------	--------------------------

The ASCII representation is included for convenience when looking at tables, etc. and is presented as a comment.

Cross reference

A listing of constants and addresses used by the program and their point of use is an aid to discovering the meaning of the program. A choice must be made as to which types of references to constants and addresses are included in the cross reference. We index jump addresses, call addresses, and load constants. Types not included are bit numbers, data offsets, field selectors, etc.

Where to?

Another important question in the design of a disassembler is where the information is put after decoding. Several options present themselves:

- display
- printer
- file

Assembler text file

The ability to place the disassembled source in a text file is a powerful convenience. It allows the user to customize BIN and LEX files easily by disassembling them, making some modifications, and reassembling. If symbol replacement is used, this option is particularly attractive.

File headers are not presented

File header information is not presented to the user since most pertinent file information is already shown in the catalog entry (length, date, etc.) and the global information (the next entry in the global file chain) doesn't say much about the file or its contents.

2 Opcode Structure & Implementation

A. Opcode symmetries

Their inevitability

No set of opcodes is a complete random arrangement of nibbles (or bytes) representing instructions. The design and construction of Central Processing Units (CPUs) necessitates an orderly approach to the opcode set. Some of these symmetries are obvious, like the field select nibble for many 71 instructions. Others are not as clear and are covered by the assembly language. Several examples are the 71 instructions of the form 11x, like 110 or A=R0. An inspection of the paper disassembler at the end of this paper shows that the third nibble, represented by x above, is divided into two sub-fields. The 2^0 through 2^2 bits are clearly the register number whereas the 2^3 bit indicates whether register A or C should be used as the destination.

Efficiency

A disassembler could be written by using one massive look-up table over all possible opcode combinations. However, this approach clearly requires a mountain of memory. The symmetries of the instruction set are the tools used to reduce the size of the disassembler. The 71 instructions 0Exx are an example of this. The fourth nibble of these instructions is divided into three subfields. Bits 2^0 through 2^1 indicate which register pair to use, vis:

AB	00
BC	01
CA	10
DC	11

Bit 2^2 indicates whether to use the pair in the order given (=0) or to reverse the order (=1). Predictably, this ordered register pair scheme is used many other times in the 71s instruction set. For these instructions, bit 2^3 indicates whether to do a logical AND (=0) or a logical OR (=1). With this knowledge, it isn't necessary to write a table of 16 instructions to look up, a few instructions will sort out the whole mess.

B. Writing a Disassembler

The first step is to define the options for input and output. Implementation of the user interface is the most time consuming. Once the instruction set is decoded and symmetries known, the actual disassembly is a classic example of parsing. A good method of learning symmetries is to build a paper disassembler. This is done by writing out the opcodes by hand and examining them.

3 Future Options

A. Reference type specification

One possible future enhancement is to provide the ability to indicate what types of references are to be cross referenced and/or replaced by symbols. This would be useful when looking only for particular types of uses of i/o devices or memory.

B. Reference table building

Another possible future option is to provide a data file which contains the reference table. This can be useful when working with multiple files since it provides the ability to combine reference sets into a global reference table.

4 Paper Disassembler

The chart on the following page is a paper disassembler which can be used to manually look up the mnemonics corresponding to opcodes. It also illustrates the structure of the opcode set.

5 Missing Op Codes

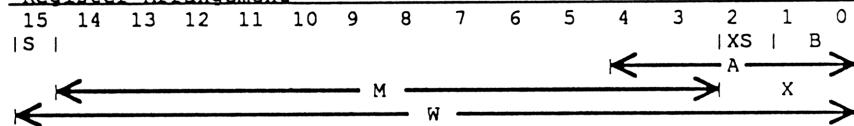
Examination of the Paper Disassembler will reveal a variety of missing op codes whose presence is suggested by symmetry elsewhere in the instruction set. These might prove an interesting subject for further exploration.

HP-71 Paper Disassembler

Copyright © 1985 Steinmetz & Brown, Ltd

Key to Symbols

Field select symbol		
Description	Field	a b
Pointer	P	0 8
Nib's 0-P	WP	1 9
Exp. sign	XS	2 A
Exponent	X	3 B
Mant. sign	S	4 C
Mantissa	M	5 D
Byte	B	6 E
Word	W	7 F

Register ArrangementNibble 1=0

00	RTNSXM	OC	P=P+1	OEaA	B=B!C a	OE6	A=A&C A
01	RTN	OD	P=P-1	OEaB	D=D!C a	OE7	C=C&D A
02	RTNSC	OEa0	A=A&B a	OEaC	B=B!A a	OE8	A=A!B A
03	RTNCC	OEa1	B=B&C a	OEaD	C=C!B a	OE9	B=B!C A
04	SETHEX	OEa2	C=C&A a	OEaE	A=A!C a	OEFA	C=C!A A
05	SETDEC	OEa3	D=D&C a	OEaF	C=C!D a	OEFB	D=D!C A
06	RSTK=C	OEa4	B=B&A a	OEFO	A=A&B A	OEFC	B=B!A A
07	C=RSTK	OEa5	C=C&B a	OEFl	B=B&C A	OEFD	C=C!B A
08	CLRST	OEa6	A=A&C a	OEF2	C=C&A A	OEFE	A=A!C A
09	C=ST	OEa7	C=C&D a	OEFl	D=D&C A	OEFF	C=C!D A
0A	ST=C	OEa8	A=A!B a	OEFO	B=B&A A	OF	RTI
0B	CSTEX	OEa9	B=B!C a	OEFl	C=C&B A		

Nibble 1=1

100	R0=A	123	AR3EX	140	DAT0=A A	156a	C=DAT0 a
101	R1=A	124	AR4EX	141	DAT1=A A	157a	C=DAT1 a
102	R2=A	128	CR0EX	142	A=DAT0 A	158x	DAT0=A d
103	R3=A	129	CR1EX	143	A=DAT1 A	159x	DAT1=A d
104	R4=A	12A	CR2EX	144	DAT0=C A	15Ax	A=DAT0 d
108	R0=C	12B	CR3EX	145	DAT1=C A	15Bx	A=DAT1 d
109	R1=C	12C	CR4EX	146	C=DAT0 A	15Cx	DAT0=C d
10A	R2=C	130	D0=A	147	C=DAT1 A	15Da	DAT1=C a
10B	R3=C	131	D1=A	148	DAT0=A B	15Ex	C=DAT0 d
10C	R4=C	132	ADOEX	149	DAT1=A B	15Fx	C=DAT1 d
110	A=R0	133	AD1EX	14A	A=DAT0 B	16x	D0=D0+ d
111	A=R1	134	D0=C	14B	A=DAT1 B	17x	D1=D1+ d
112	A=R2	135	D1=C	14C	DAT0=C B	18x	D0=D0- d
113	A=R3	136	CDOEX	14D	DAT1=C B	19nn	D0(2)=nn
114	A=R4	137	CD1EX	14E	C=DAT0 B	1Annn	D0(4)=nnnn
118	C=R0	138	D0=AS	14F	C=DAT1 B	1Bnnnn	D0(5)=nnnnn
119	C=R1	139	D1=AS	150a	DAT0=A a	1Cx	D1=D1- d
11A	C=R2	13A	ADOXS	151a	DAT1=A a	1Dnn	D1(2)=nn
11B	C=R3	13B	AD1XS	152a	A=DAT0 a	1Ennn	D1(4)=nnnn
11C	C=R4	13C	D0=CS	153a	A=DAT1 a	1Fnnnn	D1(5)=nnnnn
120	AR0EX	13D	D1=CS	154a	DAT0=C a		
121	AR1EX	13E	CDOXS	155a	DAT1=C a		
122	AR2EX	13F	CD1XS				

Nibble 1=2

2n P= n

Nibble 1 =3

3xn..n LC(x) n..n

Nibble 1=4

4aaaaa GOC aaaaaa

Nibble 1=5

5aaaaa GONC aaaaaa

Nibble 1=6

6aaaaa GOTO aaaaaa

Nibble 1=7

7aaaaa GOSUB aaaaaa

Nibble 1=8

800	OUT=CS		814	ASRC		87nyy	?ST#0	n		8B1yy	?B>C	A
801	OUT=C		815	BSRC		88nyy	?P#	n		8B2yy	?C>A	A
802	A=IN		816	CSRC		89nyy	?P=	n		8B3yy	?D>C	A
803	C=IN		817	DSRC		8A0yy	?A=B	A		8B4yy	?A<B	A
804	UNCNFG		81C	ASRB		8A1yy	?B=C	A		8B5yy	?B<C	A
805	CONFG		81D	BSRB		8A2yy	?C=A	A		8B6yy	?C<A	A
806	C=ID		81E	CSR		8A3yy	?D=C	A		8B7yy	?D<C	A
8080	INTON		81F	DSRB		8A4yy	?A#B	A		8B8yy	?A>=B	A
808F	INTOFF		821	XM=0		8A5yy	?B#C	A		8B9yy	?B>=C	A
809	C+P+1		822	SB=0		8A6yy	?C#A	A		8BAyy	?C>=A	A
80A	RESET		824	SR=0		8A7yy	?D#C	A		8BByy	?D>=C	A
80B	BUSCC		828	MP=0		8A8yy	?A=0	A		8BCyy	?A<=B	A
80Cn	C=P	n	82F	CLRHST		8A9yy	?B=0	A		8BDyy	?B<=C	A
80Dn	P=C	n	831yy	?XM=0		8AAyy	?C=0	A		8BEyy	?C<=A	A
80E	SREQ?		832yy	?SB=0		8AByy	?D=0	A		8BFyy	?D<=C	A
80Fn	CPEX	n	834yy	?SR=0		8ACyy	?A#0	A		8Caaaa	GOLONG	aaaa
810	ASLC		838yy	?MP=0		8ADyy	?B#0	A		8Daaaaa	GOVLNG	aaaaaa
811	BSLC		84n	ST=0	n	8AEyy	?C#0	A		8Eaaaa	GOSUBL	aaaa
812	CSLC		85n	ST=1	n	8AFyy	?D#0	A		8Faaaaa	GOSBVL	aaaaa
813	DSLC		86nyy	?ST#1	n	8B0yy	?A>B	A				

Nibble 1=9

9a0yy	?A=B	a	9a8	?A=0	a	9b0	?A>B	b	9b8	?A>=B	b
9a1	?B=C	a	9a9yy	?B=0	a	9b1	?B>C	b	9b9	?B>=C	b
9a2	?C=A	a	9aA	?C=0	a	9b2	?C>A	b	9bA	?C>=A	b
9a3	?D=C	a	9aB	?D=0	a	9b3	?D>C	b	9bB	?D>=C	b
9a4	?A#B	a	9aCyy	?A#0	a	9b4	?A<B	b	9bCyy	?A<=B	b
9a5	?B#C	a	9aD	?B#0	a	9b5	?B<C	b	9bD	?B<=C	b
9a6	?C#A	a	9aE	?C#0	a	9b6yy	?C<A	b	9bE	?C<=A	b
9a7	?D#C	a	9aFyy	?D#0	a	9b7	?D<C	b	9bFyy	?D<=C	b

Nibble 1=A

Aa0	A=A+B	a	Aa8	B=B+A	a	Ab0	A=0	b	Ab8	B=A	b
Aa1	B=B+C	a	Aa9	C=C+B	a	Ab1	B=0	b	Ab9	C=B	b
Aa2	C=C+A	a	AaA	A=A+C	a	Ab2	C=0	b	AbA	A=C	b
Aa3	D=D+C	a	AaB	C=C+D	a	Ab3	D=0	b	AbB	C=D	b
Aa4	A=A+A	a	AaC	A=A-1	a	Ab4	A=B	b	AbC	ABEX	b
Aa5	B=B+B	a	AaD	B=B-1	a	Ab5	B=C	b	AbD	BCEX	b
Aa6	C=C+C	a	AaE	C=C-1	a	Ab6	C=A	b	AbE	ACEX	b
Aa7	D=D+D	a	AaF	D=D-1	a	Ab7	D=C	b	AbF	DCEX	b

Nibble 1=B

Ba0	A=A-B	a	Ba8	B=B-A	a	Bb0	ASL	b	Bb8	A=-A	b
Ba1	B=B-C	a	Ba9	C=C-B	a	Bb1	BSL	b	Bb9	B=-B	b
Ba2	C=C-A	a	BaA	A=A-C	a	Bb2	CSL	b	BbA	C=-C	b
Ba3	D=D-C	a	BaB	C=C-D	a	Bb3	DSL	b	BbB	D=-D	b
Ba4	A=A+1	a	BaC	A=B-A	a	Bb4	ASR	b	BbC	A=-A-1	b
Ba5	B=B+1	a	BaD	B=C-B	a	Bb5	BSR	b	BbD	B=-B-1	b
Ba6	C=C+1	a	BaE	C=C-A	a	Bb6	CSR	b	BbE	C=-C-1	b
Ba7	D=D+1	a	BaF	D=C-D	a	Bb7	DSR	b	BbF	D=-D-1	b

Nibble 1=C

C0	A=A+B	a
C1	B=B+C	a
C2	C=C+A	a
C3	D=D+C	a
C4	A=A+A	a
C5	B=B+B	a
C6	C=C+C	a
C7	D=D+D	a
C8	B=B+A	a
C9	C=C+B	a
CA	A=A+C	a
CB	C=C+D	a
CC	A=A-1	a
CD	B=B-1	a
CE	C=C-1	a
CF	D=D-1	a

Nibble 1=D

D0	A=0	a
D1	B=0	a
D2	C=0	a
D3	D=0	a
D4	A=B	a
D5	B=C	a
D6	C=A	a
D7	D=C	a
D8	B=A	a
D9	C=B	a
DA	A=C	a
DB	C=D	a
DC	ABEX	a
DD	BCEX	a
DE	ACEX	a
DF	DCEX	a

Nibble 1=E

E0	A=A-B	a
E1	B=B-C	a
E2	C=C-A	a
E3	D=D-C	a
E4	A=A+1	a
E5	B=B+1	a
E6	C=C+1	a
E7	D=D+1	a
E8	B=B-A	a
E9	C=C-B	a
EA	A=A-C	a
EB	C=C-D	a
EC	A=B-A	a
ED	B=C-B	a
EE	C=C-A	a
EF	D=C-D	a

Nibble 1=F

F0	ASL	a
F1	BSL	a
F2	CSL	a
F3	DSL	a
F4	ASR	a
F5	BSR	a
F6	CSR	a
F7	DSR	a
F8	A=-A	a
F9	B=-B	a
FA	C=-C	a
FB	D=-D	a
FC	A=-A-1	a
FD	B=-B-1	a
FE	C=-C-1	a
FF	D=-D-1	a

INDEX BASIC 4352 06/11/85 17:16

```
10 ! INDEX, program to disassemble LEX file header and tables (see PPC V1ZN2).
20 ! program needs DISLEX lex file.
30 !
40 IF NOT POS(VER$,'DIS:') THEN BEEP 1450,.2 @ DISP "NEED DISLEX LEX FILE" @ END

50 DESTROY ALL @ STD @ DELAY 0,0 @ CFLAG 1 @ CFLAG 63 @ DISP CHR$(27)&'=';
60 DIM L1$(23),L2$(23),L3$(23),L4$(23)
70 L1$="-----"
80 L2$="*****"
90 L3$="-----"
100 L4$="xxxxxxxxxxxxxxxxxxxxxx"
110 !
120 DEF FNI$(A$,N)=DTH$(HTD(A$)+N)
130 DEF FNPS$(A$,N)=REV$(PEEK$(A$,N))
140 DEF FND$(A$,N) @ I$=REV$(PEEK$(A$,N))
150 I$=I$&" - "&STR$(HTD(I$)) @ FND$=I$ @ END DEF
160 DEF FNT$(P$,A$,N) @ P=HTD(P$)
170 IF P>=16^N/2 THEN P=-(16^N-P)
180 FNT$=DTH$(HTD(A$)+P) @ END DEF
190 DEF FNW$(P$,N) @ I$="" @ FOR I=N TO 1 STEP -2
200 I$=I$&CHR$(HTD(P$[I-1,I])) @ NEXT I
210 FNW$=I$ @ END DEF
220 !
230 ! XROM 01 start at #1DD04 ..... XROM 00 start at #1E537
240 !
250 INPUT "FILE NAME:   ?,N$ @ N$=UPRC$(N$)
260 IF N$='?' THEN 'INHEX' ELSE A$=ADDR$(N$) @ GOTO 'HDR'
270 'INHEX': INPUT "HEX ADDRESS ? ",'1DD04';L$ @ SFLAG 1 @ DISP @ GOTO 1160
280 'HDR': DISP @ PRINT L2$ @ PRINT L1$
290 PRINT " --FILE HEADER--" @ PRINT TAB(10);#"@PRINT L1$
300 PRINT "File Name:   ";FNW$(FNPS$(A$,16),16) @ A$=FNI$(A$,16)
310 IF FNPS$(A$,4)="#E208" THEN BEEP 1450 @ DISP "NOT A LEX FILE" @ END
320 A$=FNI$(A$,4) @ PRINT "Flags=      ";FNPS$(A$,1) @ A$=FNI$(A$,1)
330 PRINT "Copy Code=    ";FNPS$(A$,1) @ A$=FNI$(A$,1)
340 P$=FNPS$(A$,4) @ PRINT "Creation Time  ";P$[1,2];":";P$[3,4] @ A$=FNI$(A$,4)
350 P$=FNPS$(A$,6) @ PRINT "Creation Date   ";P$[1,2];"/";P$[3,4];"/";P$[5,6] @ A$=FNI$(A$,6)
360 P$=FNPS$(A$,5) @ PRINT "Next File at:  #";FNT$(P$,A$,5) @ A$=FNI$(A$,5)
370 PRINT L1$ @ PRINT TAB(10);#"@DTH$(HTD(A$)-1)
380 PRINT " **END of FILE HEADER**" @ PRINT L1$
390 'AGAIN': IF PEEK$(A$,2)='00' THEN SFLAG 63
400 PRINT "LEX ID:      #";FND$(A$,2) @ A$=FNI$(A$,2)
410 K=HTD(FNPS$(A$,2)) @ A$=FNI$(A$,2) @ PRINT "Low Token=      ";K
420 K9=HTD(FNPS$(A$,2)) @ A$=FNI$(A$,2) @ PRINT "High Token=     ";K9
430 P$=FNPS$(A$,5) @ L$=FNT$(P$,A$,5) @ A$=FNI$(A$,5)
440 PRINT "LEX Table Link  #";
450 IF P$#000000 THEN SFLAG 1 @ PRINT L$ ELSE PRINT P$
460 P$=FNPS$(A$,1) @ A$=FNI$(A$,1) @ IF P$='F' THEN PRINT "NO Speed Table" @ GOTO 'TXTOFS'
470 PRINT "Speed Table from #";DTH$(HTD(A$)-1) @ A$=FNI$(A$,79)
480 PRINT TAB(15);"to #";DTH$(HTD(A$)-1)
490 'TXTOFS': P$=FNPS$(A$,4) @ T$=DTH$(HTD(FNT$(P$,A$,4))-1) @ A$=FNI$(A$,4)
500 IF PEEK$(T$,3)##1FF THEN PRINT "Text Table:      #";T$ ELSE PRINT "NO Text Table"
510 P$=FNPS$(A$,4)
```

```

520 IF FLAG(63) THEN PRINT "Mainframe Message Table" @ GOTO 'POLL'
530 IF P$='0000' THEN PRINT "NO Message Table" @ M$="" @ GOTO 'POLL'
540 M$=FNT$(P$,A$,4) @ PRINT "Message Table: ##";M$
550 'POLL': A$=FNI$(A$,4) @ P$=FNP$(A$,5)
560 IF P$='00000' THEN PRINT "NO Poll Handler" @ GOTO 'MAIN'
570 PRINT "Poll Handler: ##";FNT$(P$,A$,5)
580 'MAIN': A$=FNI$(A$,5) @ PRINT L2$ @ PRINT L2$
590 PRINT " --MAIN TABLE--" @ PRINT TAB(10);##;A$
600 'MAINT': N=3 @ PRINT L1$ @ P$=FNP$(A$,N)
610 K$="00"&STR$(K) @ K$=K$[LEN(K$)-2,LEN(K$)]
620 PRINT "KEYWORD ";K$;" at ##;FNT$(P$,T$,5) @ A$=FNI$(A$,3)
630 P$=FNP$(A$,5)
640 PRINT "EXECUTION start ##;FNT$(P$,A$,5) @ A$=FNI$(A$,5) @ K=K+1
650 PRINT "Characterization Nib ##;FNP$(A$,1) @ A$=FNI$(A$,1)
660 IF K<=K9 THEN 'MAINT'
670 PRINT L1$ @ PRINT TAB(10);##;DTH$(HTD(A$)-1)
680 PRINT " **END of MAIN TABLE**" @ A$=T$
690 IF PEEK$(A$,3)='1FF' THEN 'ETT' ELSE PRINT L2$ @ PRINT L2$
700 PRINT " --TEXT TABLE--" @ PRINT TAB(10);##;A$
710 'TEXTBL': PRINT L1$ @ N=HTD(PEEK$(A$,1))+1 @ A$=FNI$(A$,1)
720 IF N<4 THEN BEEP 1450 @ DISP "Keyword TOO SHORT" @ END
730 PRINT "Keyword ";FNW$(FNP$(A$,N),N) @ A$=FNI$(A$,N)
740 PRINT "Token= ";HTD(FNP$(A$,2)) @ A$=FNI$(A$,2)
750 IF PEEK$(A$,3)='1FF' THEN 'TEXTBL'
760 PRINT L1$ @ PRINT TAB(10);##;DTH$(HTD(A$)+2)
770 PRINT " **END of TEXT TABLE**"
780 'ETT': P$=FNP$(A$,2) @ A$=FNI$(A$,2)
790 IF FLAG(63) THEN M$='1F3E0'
800 IF M$="" THEN 'EXIT'
810 PRINT L2$ @ PRINT L2$ @ PRINT " --MESSAGE TABLE--" @ A$=M$
820 PRINT TAB(10);##;A$ @ PRINT L1$ 
830 P$=FND$(A$,2) @ IF FLAG(63) THEN P$="01 - 1"
840 PRINT "Low Message: ##;P$ @ A$=FNI$(A$,2)
850 P$=FND$(A$,2) @ IF FLAG(63,0) THEN P$="F8 - 248" @ A$='1F3E8'
860 PRINT "High Message: ##;P$ @ A$=FNI$(A$,2)
870 'MSGB': PRINT L4$ @ P$=FNP$(A$,2) @ A$=FNI$(A$,2)
880 IF P$='FF' THEN 'FEXIT'
890 PRINT "Block Length=";HTD(P$);nibs"
900 PRINT "Message ID: ##;FND$(A$,2) @ A$=FNI$(A$,2)
910 'CELL': P$=FNP$(A$,1) @ A$=FNI$(A$,1)
920 IF P$='C' THEN 'MSGB' ELSE PRINT L3$ 
930 IF P$='B' THEN 'LONG'
940 IF P$='E' THEN 'MFBB'
950 IF P$='D' THEN 'LBB'
960 IF P$&PEEK$(A$,1)='F0' THEN 'DLBB'
970 IF P$&PEEK$(A$,1)='F1' THEN 'IMC'
980 IF P$&PEEK$(A$,1)='F2' THEN 'ITNS'
990 IF P$&PEEK$(A$,1)='F3' THEN 'ITWS'
1000 PRINT " -Text Cell-"
1010 'TEXT': N=(HTD(P$)+1)*2 @ PRINT FNW$(FNP$(A$,N),N) @ A$=FNI$(A$,N) @ GOTO 'CELL'
1020 'LONG': PRINT " -Long Text Cell-" @ P$=FNP$(A$,1) @ A$=FNI$(A$,1) @ GOTO 'TEXT'
1030 'MFBB': PRINT "-Mainframe Bldg. Block-"
1040 'BBP': PRINT TAB(9);##;FND$(A$,2) @ A$=FNI$(A$,2) @ GOTO 'CELL'
1050 'LBB': PRINT " -Local Bldg. Block-" @ GOTO 'BBP'
1060 'DLBB': PRINT " -Ext LEX Bldg. Block-" @ A$=FNI$(A$,1)
1070 PRINT "Message ID: ##;FND$(A$,2) @ A$=FNI$(A$,2)

```

```
1080 PRINT "LEX ID:      #";FND$(A$,2) @ A$=FNI$(A$,2) @ GOTO 'CELL'
1090 'IMS': PRINT "-Indirect Message Cell-"
1100 'STR': A$=FNI$(A$,1) @ PRINT "-value passed in R2-" @ GOTO 'CELL'
1110 'ITNS': PRINT "-Insert Cell No Space-" @ GOTO 'STR'
1120 'ITWS': PRINT "-Insert Cell W/Space-" @ GOTO 'STR'
1130 'FEXIT': PRINT TAB(10);"#";DTH$(HTD(A$)-1)
1140 PRINT " *END of MESSAGE TABLE*"
1150 'EXIT': PRINT L2$
1160 IF FLAG(1,0) THEN PRINT L2$ @ PRINT TAB(10);"#";L$ @ A$=L$ @ GOTO 'AGAIN'
1170 END
```

```
*****
-----
--FILE HEADER--
#70A93
-----
File Name: DISLEX
Flags= 1
Copy Code= 0
Creation Time 10:32
Creation Date 85/06/13
Next File at: #70F17
-----
#70AB7
**END of FILE HEADER**
-----
LEX ID: #35 - 53
Low Token= 1
High Token= 7
LEX Table Link #00000
NO Speed Table
Text Table: #70B10
Message Table: #70BAE
Poll Handler: #70B78
*****
*****  
--MAIN TABLE--
#70AD1
-----
KEYWORD 001 at #70B3F
EXECUTION start #70C03
Characterization Nib #D
-----
KEYWORD 002 at #70B2C
EXECUTION start #70CF9
Characterization Nib #F
-----
KEYWORD 003 at #70B1D
EXECUTION start #70D30
Characterization Nib #F
-----
KEYWORD 004 at #70B4C
EXECUTION start #70D6D
Characterization Nib #F
-----
KEYWORD 005 at #70B10
EXECUTION start #70D7D
Characterization Nib #F
-----
KEYWORD 006 at #70B57
EXECUTION start #70DD7
Characterization Nib #D
-----
KEYWORD 007 at #70B66
EXECUTION start #70E6F
Characterization Nib #D
-----
#70B0F
**END of MAIN TABLE**
*****
```

```
--TEXT TABLE--
#70B10
-----
Keyword CMDST
Token= 5
-----
Keyword GETNIB
Token= 3
-----
Keyword KEYWAIT$
Token= 2
-----
Keyword RCOPY
Token= 1
-----
Keyword REV$
Token= 4
-----
Keyword SETCMD
Token= 6
-----
Keyword UNPRIV
Token= 7
-----
#70B77
**END of TEXT TABLE**
*****  
*****  
--MESSAGE TABLE--
#70BAE
-----
Low Message: #00 - 0
High Message: #03 - 3
xxxxxxxxxxxxxxxxxxxx
Block Length= 16 nibs
Message ID: #00 - 0
-----  
-Text Cell-
DISS
xxxxxxxxxxxxxxxxxxxx
Block Length= 8 nibs
Message ID: #01 - 1
-----  
-Mainframe Bldg. Block-
#0B - 11
xxxxxxxxxxxxxxxxxxxx
Block Length= 8 nibs
Message ID: #02 - 2
-----  
-Mainframe Bldg. Block-
#3A - 58
xxxxxxxxxxxxxxxxxxxx
Block Length= 37 nibs
Message ID: #03 - 3
-----  
-Long Text Cell-
File not in RAM
xxxxxxxxxxxxxxxxxxxx
#70BF8
*END of MESSAGE TABLE*
```

DISASS BASIC 4621 06/11/85 15:12

```
10 ! DISASS, program to disassemble HP71 machine language (see CHHU VZN1).
20 ! program needs DISLEX lex file.
30 !
40 IF NOT POS(VER$,'DIS:') THEN BEEP 1450,.2 @ DISP "NEED DISLEX LEX FILE" @ END

50 DESTROY ALL @ STD @ DELAY 0,0 @ CFLAG 63 @ DISP CHR$(27)&'=';
60 !
70 DEF FNY(Q$) @ DISP Q$
80 Y=POS('NY',UPRC$(KEYWAIT$))-1 @ IF Y<0 THEN 80
90 FNY=Y @ END DEF
100 !
110 INTEGER N,O,I @ REAL A,Z
120 DIM A0$[5],Z$[5],C$,M$ @ ON ERROR GOTO 140
130 !
140 INPUT "Start address: ','?';A0$ @ A=HTD(A0$)-1 @ ON ERROR GOTO 150
150 INPUT "End address: ','?';Z$ @ Z=HTD(Z$) @ GOTO 1220
160 DISP @ DISP "Initializing...." @ GOSUB 860
170 DISP @ DISP "DISASS from "&A0$ @ DISP
180 !
190 A=A+1 @ IF FLAG(0) THEN PAUSE
200 IF A>Z THEN BEEP 1500 @ DISP @ DISP "DONE" @ END
210 B=A @ O=GETNIB(A)+1 @ A=A+1 @ N=GETNIB(A) @ M$=""
220 ON O GOSUB 280,330,490,500,510,520,550,560,600,750,760,770,780,780,780,780
230 F0$=DTH$(B)&" "&C$&" " @ F0$=F0$[1,14]
240 F0$=F0$&M$ @ IF LEN(F0$)<24 THEN F0$=F0$&" " @ F0$=F0$[1,23]
250 IF FLAG(1) THEN PRINT #1;F0$
260 PRINT F0$ @ GOTO 190
270 !
280 IF N#14 THEN C$=R0$(N) @ RETURN
290 A=A+1 @ N=GETNIB(A) @ IF N>8 AND N#15 THEN 790
300 A=A+1 @ C$=L$(GETNIB(A))
310 IF N=15 THEN M$="A" ELSE M$=F$(N)
320 RETURN
330 A=A+1
340 IF N>7 THEN ON N-7 GOTO 390,410,420,430,390,410,420,430
350 ON N+1 GOTO 360,360,360,360,370,460,390,390
360 C$=R1$(N*16+GETNIB(A)) @ IF LEN(C$) THEN RETURN ELSE 790
370 N=GETNIB(A) @ C$=R1$(MOD(N,8)+64) @ M$="A" @ IF N>7 THEN M$="B"
380 RETURN
390 IF N=12 THEN N=9
400 C$=D$(N-6) @ M$=STR$(GETNIB(A)+1) @ RETURN
410 O=2 @ GOTO 440
420 O=4 @ GOTO 440
430 O=5
440 C$="D0=HEX" @ IF N>12 THEN C$="D1=HEX"
450 M$=REV$(PEEK$(DTH$(A),O)) @ A=A+O-1 @ RETURN
460 N=GETNIB(A) @ A=A+1 @ C$=R1$(MOD(N,8)+64) @ IF N>8 THEN 480
470 N=GETNIB(A) @ IF N>7 THEN 790 ELSE M$=F$(N) @ RETURN
480 M$=STR$(GETNIB(A)+1) @ RETURN
490 C$="P=" @ M$=STR$(N) @ RETURN
500 C$="LCHEX" @ M$=REV$(PEEK$(DTH$(A+1),N+1)) @ A=A+N+1 @ RETURN
510 C$="GOC" @ N=2 @ GOTO 530
520 C$="GONC" @ N=2
530 IF PEEK$(DTH$(A),2)<>"00" THEN 570 ELSE C$="RTNC" @ IF O=6 THEN C$="RTNCC"
540 A=A+1 @ RETURN
```

```

550 C$="GOTO" @ N=3 @ GOTO 570
560 C$="GOSUB" @ N=3 @ SFLAG 63
570 A3=HTD( REV$( PEEK$( DTH$( A ), N ) ) )
580 IF A3)=16^N/2 THEN A3=-(16^N-A3)
590 M$=DTH$( ABS(A+A3+N*FLAG(63)) ) @ CFLAG 63 @ A=A+N-1 @ RETURN
600 A=A+1 @ O=GETNIB(A)
610 IF N<8 THEN ON N+1 GOTO 630,650,650,660,670,670,680,680
620 ON N-7 GOTO 680,680,690,690,710,720,700,720
630 IF O=8 THEN 730 ELSE IF O#12 AND O#13 AND O#15 THEN 650
640 A=A+1 @ C$=R3$(0) @ M$=STR$(GETNIB(A)) @ RETURN
650 C$=R3$(N*16+0) @ IF LEN(C$) THEN RETURN ELSE 790
660 C$=R3$(N*16+0) @ IF LEN(C$) THEN 800 ELSE 790
670 C$=R4$(N-4) @ M$=STR$(0) @ RETURN
680 C$=R4$(N-4) @ M$=STR$(0) @ GOTO 800
690 C$=T$((N-10)*16+0) @ M$="A" @ GOTO 800
700 SFLAG 63
710 C$=R5$(N-12) @ N=4 @ GOTO 570
720 C$=R5$(N-12) @ M$=REV$(PEEK$(DTH$(A),5)) @ A=A+4 @ RETURN
730 A=A+1 @ C$="INTOFF" @ N=GETNIB(A)
740 IF N=15 THEN RETURN ELSE C$="INTON" @ IF N=0 THEN RETURN ELSE 790
750 A=A+1 @ C$=T$((N)7)*16+GETNIB(A) @ M$=F$(MOD(N,8)) @ GOTO 800
760 A=A+1 @ C$=G$((N)7)*16+GETNIB(A) @ M$=F$(MOD(N,8)) @ RETURN
770 A=A+1 @ C$=G$(32+(N)7)*16+GETNIB(A) @ M$=F$(MOD(N,8)) @ RETURN
780 C$=G$((GETNIB(A-1)-12)*16+N) @ M$="A" @ RETURN
790 C$="Unknown "&PEEK$(DTH$(B),A-B+1) @ RETURN
800 F0$=DTH$(B)&" "&C$&" " @ F0$=F0$[1,14]
810 F0$=F0$&M$ @ IF LEN(F0$)<24 THEN F0$=F0$&" " @ F0$=F0$[1,23]
820 PRINT F0$ @ IF FLAG(1) THEN PRINT #1;F0$
830 A=A+1 @ B=A @ IF FLAG(0) THEN PAUSE
840 IF GETNIB(A)+GETNIB(A+1) THEN C$="GOYES" @ N=2 @ GOTO 570
850 C$="RTNYES" @ A=A+1 @ RETURN
860 OPTION BASE 0
870 DIM R1$(71)[6],R3$(56)[6],R4$(5)[5],R5$(3)[6],T$(31)[5],L$(15)[5],F$(7)[2]
880 DIM G$(63)[6],D$(3)[6],R0$(15)[6]
890 READ R1$( ),R3$( ),R4$( ),R5$( ),T$( ),L$( ),F$( ),G$( ),D$( ),R0$( ) @ RETURN
900 DATA R0=A,R1=A,R2=A,R3=A,R4=A,,,R0=C,R1=C,R2=C,R3=C,R4=,,,
910 DATA A=R0,A=R1,A=R2,A=R3,A=R4,,,C=R0,C=R1,C=R2,C=R3,C=R4,,,
920 DATA AR0EX,AR1EX,AR2EX,AR3EX,AR4EX,,,CR0EX,CR1EX,CR2EX,CR3EX,CR4EX,,,
930 DATA D0=A,D1=A,AD0EX,AD1EX,D0=C,D1=C,CD0EX,CD1EX,D0=AS,D1=AS,AD0XS,AD1XS
940 DATA D0=CS,D1=CS,CD0XS,CD1XS
950 DATA DAT0=A,DAT1=A,A=DAT0,A=DAT1,DAT0=C,DAT1=C,C=DAT0,C=DAT1
960 DATA OUT=CS,OUT=C,A=IN,C=IN,UNCNFG,CONFIG,C=ID,SHUTDN,,C+P+1,RESET
970 DATA BUSCC,C=P,C=P,SREQ?,CPLEX
980 DATA ASLC,BSLC,CSLC,DSLC,ASRC,BSRC,CSRC,DSRC,,,,ASRB,BSRB,CSRB,DSRB
990 DATA ,XM=0,SB=0,,SR=0,,,MP=0,,,,,,CLRHST
1000 DATA ,?XM=0,?SB=0,,?SR=0,,,?MP=0
1010 DATA ST=0,ST=1,?ST=0,?ST=1,?P#,?P=
1020 DATA GOLONG,GOVLNG,GOSUBL,GOSBVL
1030 DATA ?A=B,?B=C,?C=A,?D=C,?A#B,?B#C,?C#A,?D#C,?A=0,?B=0,?C=0,?D=0
1040 DATA ?A#0,?B#0,?C#0,?D#0
1050 DATA ?A>B,?B>C,?C>A,?D>C,?A(B,?B(C,?C(A,?D(C
1060 DATA ?A)=B,?B)=C,?C)=A,?D)=C,?A<=B,?B<=C,?C<=A,?D<=C
1070 DATA A=A&B,B=B&C,C=C&A,D=D&C,B=A&B,C=B&C,A=C&A,C=D&C
1080 DATA A=A!B,B=B!C,C=C!A,D=D!C,B=A!B,C=B!C,A=C!A,C=D!C
1090 DATA P,WP,XS,X,S,M,B,W
1100 DATA A=A+B,B=B+C,C=C+A,D=D+C,A=A+A,B=B+B,C=C+C,D=D+D
1110 DATA B=A+B,C=B+C,A=C+A,C=D+C,A=A-1,B=B-1,C=C-1,D=D-1
1120 DATA A=0,B=0,C=0,D=0,A=B,B=C,C=A,D=C,B=A,C=B,A=C,C=D

```

```
1130 DATA ABEX,BCEX,CAEX,DCEX
1140 DATA A=A-B,B=B-C,C=C-A,D=D-C,A=A+1,B=B+1,C=C+1,D=D+1
1150 DATA B=B-A,C=C-B,A=A-C,C=C-D,A=B-A,B=C-B,C=A-C,D=C-D
1160 DATA ASL,BSL,CSL,DSL,ASR,BSR,CSR,DSR
1170 DATA A=-A,B=-B,C=-C,D=-D,A=-A-1,B=-B-1,C=-C-1,D=-D-1
1180 DATA D0=D0+,D1=D1+,D0=D0-,D1=D1-
1190 DATA RTNSXM,RTN,RTNSC,RTNCC,SETHEX,SETDEC,RSTK=C,C=RSTK
1200 DATA CLRST,C=ST,ST=C,CSTEX,P=P+1,P=P-1,,RTI
1210 !
1220 CFLAG 1 @ ON ERROR GOTO 1250
1230 INPUT "File name      ','?';Y$ @ IF Y$='?' THEN OFF ERROR @ GOTO 160
1240 CREATE TEXT Y$ @ ASSIGN #1 TO Y$ @ OFF ERROR @ SFLAG 1 @ GOTO 160
1250 IF ERRN#59 AND ERRN#255030 THEN OFF ERROR @ DISP ERRM$ @ BEEP @ END
1260 IF NOT FNY("overwrite file (Y/N)?"") THEN 1230
1270 PURGE Y$ @ GOTO 1240
```

R1\$(71)[6]

0 - R0=A	56 - D0=AS
1 - R1=A	57 - D1=AS
2 - R2=A	58 - AD0XS
3 - R3=A	59 - AD1XS
4 - R4=A	60 - D0=CS
5 -	61 - D1=CS
6 -	62 - CD0XS
7 -	63 - CD1XS
8 - R0=C	64 - DAT0=A
9 - R1=C	65 - DAT1=A
10 - R2=C	66 - A=DAT0
11 - R3=C	67 - A=DAT1
12 - R4=C	68 - DAT0=C
13 -	69 - DAT1=C
14 -	70 - C=DAT0
15 -	71 - C=DAT1
16 - A=R0	
17 - A=R1	
18 - A=R2	
19 - A=R3	
20 - A=R4	
21 -	
22 -	
23 -	
24 - C=R0	
25 - C=R1	
26 - C=R2	
27 - C=R3	
28 - C=R4	
29 -	
30 -	
31 -	
32 - AR0EX	
33 - AR1EX	
34 - AR2EX	
35 - AR3EX	
36 - AR4EX	
37 -	
38 -	
39 -	
40 - CR0EX	
41 - CR1EX	
42 - CR2EX	
43 - CR3EX	
44 - CR4EX	
45 -	
46 -	
47 -	
48 - D0=A	
49 - D1=A	
50 - AD0EX	
51 - AD1EX	
52 - D0=C	
53 - D1=C	
54 - CD0EX	
55 - CD1EX	

R4\$(5)[5]

0 - ST=0
1 - ST=1
2 - ?ST=0
3 - ?ST=1
4 - ?P#
5 - ?P=
R5\$(3)[6]
0 - GOLONG
1 - GOVLNG
2 - GOSUBL
3 - GOSBVL
T\$(31)[5]
0 - ?A=B
1 - ?B=C
2 - ?C=A
3 - ?D=C
4 - ?A#B
5 - ?B#C
6 - ?C#A
7 - ?D#C
8 - ?A=0
9 - ?B=0
10 - ?C=0
11 - ?D=0
12 - ?A#0
13 - ?B#0
14 - ?C#0
15 - ?D#0
16 - ?A>B
17 - ?B>C
18 - ?C>A
19 - ?D>C
20 - ?A<B
21 - ?B<C
22 - ?C<A
23 - ?D<C
24 - ?A>=B
25 - ?B>=C
26 - ?C>=A
27 - ?D>=C
28 - ?A<=B
29 - ?B<=C
30 - ?C<=A
31 - ?D<=C

R3\$(56)[6]

0 - OUT=CS
1 - OUT=C
2 - A=IN
3 - C=IN
4 - UNCNFG
5 - CONFIG
6 - C=ID
7 - SHUTDOWN
8 -
9 - C+P+1
10 - RESET
11 - BUSCC
12 - C=P
13 - C=P
14 - SREQ?
15 - CPEX
16 - ASLC
17 - BSLC
18 - CSLC
19 - DSLC
20 - ASRC
21 - BSRC
22 - CSRC
23 - DSRC
24 -
25 -
26 -
27 -
28 - BSRB
29 - BSRB
30 - CSR8
31 - DSRB
32 -
33 - XM=0
34 - SB=0
35 -
36 - SR=0
37 -
38 -
39 -
40 - MP=0
41 -
42 -
43 -
44 -
45 -
46 -
47 - CLRHST
48 -
49 - ?XM=0
50 - ?SB=0
51 -
52 - ?SR=0
53 -
54 -
55 -
56 - ?MP=0

G\$(63)[6]

0 - A=A+B
 1 - B=B+C
 2 - C=C+A
 3 - D=D+C
 4 - A=A+A
 5 - B=B+B
 6 - C=C+C
 7 - D=D+D
 8 - B=A+B
 9 - C=B+C
 10 - A=C+A
 11 - C=D+C
 12 - A=A-1
 13 - B=B-1
 14 - C=C-1
 15 - D=D-1
 16 - A=0
 17 - B=0
 18 - C=0
 19 - D=0
 20 - A=B
 21 - B=C
 22 - C=A
 23 - D=C
 24 - B=A
 25 - C=B
 26 - A=C
 27 - C=D
 28 - ABEX
 29 - BCEX
 30 - CAEX
 31 - DCEX
 32 - A=A-B
 33 - B=B-C
 34 - C=C-A
 35 - D=D-C
 36 - A=A+1
 37 - B=B+1
 38 - C=C+1
 39 - D=D+1
 40 - B=B-A
 41 - C=C-B
 42 - A=A-C
 43 - C=C-D
 44 - A=B-A
 45 - B=C-B
 46 - C=A-C
 47 - D=C-D
 48 - ASL
 49 - BSL
 50 - CSL
 51 - DSL
 52 - ASR
 53 - BSR
 54 - CSR
 55 - DSR
 56 - A=-0

L\$(15)[5]

0 - A=A&B
 1 - B=B&C
 2 - C=C&A
 3 - D=D&C
 4 - B=A&B
 5 - C=B&C
 6 - A=C&A
 7 - C=D&C
 8 - A=A!B
 9 - B=B!C
 10 - C=C!A
 11 - D=D!C
 12 - B=A!B
 13 - C=B!C
 14 - A=C!A
 15 - C=D!C

F\$(7)[2]

0 - P
 1 - WP
 2 - XS
 3 - X
 4 - S
 5 - M
 6 - B
 7 - W

D\$(3)[6]

0 - D0=D0+
 1 - D1=D1+
 2 - D0=D0-
 3 - D1=D1-

R0\$(15)[6]

0 - RTNSXM
 1 - RTN
 2 - RTNSC
 3 - RTNCC
 4 - SETHEX
 5 - SETDEC
 6 - RSTK=C
 7 - C=RSTK
 8 - CLRST
 9 - C=ST
 10 - ST=C
 11 - CSTEX
 12 - P=P+1
 13 - P=P-1
 14 -
 15 - RTI

Disassembled listing of DISLEX lex file made by program DISASS.

POLHND * entry point manually inserted
70B78 ?B=0 B
70B7B GOYES 70B80
70B7D GONC 70BAC
70B80 C=R3
70B83 D1=C
70B86 A=R2
70B89 D1=D1- 12
70B8C CD1EX
70B8F ?C>A A
70B92 GOYES 70BAC
70B94 D1=C
70B97 R3=C
70B9A LCHEX 204449533A42
70BA8 DAT1=C 12
70BAC RTNSXM

70BF9 dOFFST 000F0 * offset to decompile routine
70BFE pOFFST 000F2 * offset to parse routine
CPY * entry point, all three manually inserted.
70C03 P= 1
70C05 GOSBVL 01531
70C0C GOC 70C62
70C0F D=0 W
70C12 LCHEX 2020
70C18 R0=C
70C1B A=DAT0 B
70C1E LCHEX F3
70C22 ?C#A B
70C25 GOYES 70C39
70C27 D0=D0+ 2
70C2A ?ST=1 3
70C2D GOYES 70C4A
70C2F GOSBVL 01131
70C36 GONC 70C7D
70C39 LCHEX F0
70C3D ?C>A B
70C40 GOYES 70C4A
70C42 ?ST=1 3
70C45 GOYES 70C59
70C47 GONC 70C2F
70C4A LCHEX D0
70C4E ?C#A B
70C51 GOYES 70C69
70C53 LCHEX 7
70C56 D=C P
70C59 D0=D0+ 2
70C5C A=0 W

```

70C5F GONC 70C80
70C62 GOVLNG 0939A
70C69 GOSBVL 0A35D
70C70 GONC 70C7D
70C73 GOSBVL 09F2D
70C7A GOC 70C62
70C7D DSLC
70C80 GOSBVL 0845A
70C87 ?ST=1 3
70C8A GOYES 70C93
70C8C ST=1 3
70C8F GOTO 70C0F
70C93 GOSUB 70CA1
70C97 GOC 70C62
70C9A GOVLNG 08A48
70CA1 GOSBVL 011CE
70CA8 ?ST=0 0
70CAB GOYES 70CBA
70CAD SETHEX
70CAF P= 0
70CB1 LCHEX 3502
70CB7 GONC 70C62
70CBA GOSBVL 08448
70CC1 GOC 70CE2
70CC4 CD1EX
70CC7 R3=C
70CCA CD1EX
70CCD GOSBVL 06BFB
70CD4 GOVLNG 082EB
70CDB GOVLNG 08401
70CE2 GOVLNG 08407
70CE9 GOVLNG 05722
70CF0 GOVLNG 03B0C
*****
70CF7 FNCTN 00 * function parameter definition
KEYW * entry point, both manually inserted.
70CF9 CD0EX
70FCF R0=C
70cff CD1EX
70D02 R1=C
70D05 GOSBVL 006C2
70D0C GOC 70D22
70D0F C=R1
70D12 D1=C
70D15 C=R0
70D18 D0=C
70D1B GOVLNG 1ACA8
70D22 GOSBVL 00721
70D29 GOTO 70D05
*****
70D2D FNCTN 11-8 * function parameter definition
PEEK * entry point, both manually inserted.
70D30 GOSBVL 0E8FD
70D37 GOSBVL 1B223
70D3E GOC 70D44
70D41 GONC 70DA3
70D44 CD1EX
70D47 CR0EX

```

70D4A D1=A
70D4D A=0 A
70D4F A=DAT1 1
70D53 GOSBVL 1B31B
70D5A CAEX W
70D5D AR0EX
70D60 AD1EX
70D63 GOVLNG 0F238

70D6A FNCTN 11-4 * function parameter definition
REV * entry point, both manually inserted.
70D6D GOSBVL 1B38E
70D74 GOVLNG 0F23C

70D7B FNCTN 00 * function parameter definition
CMDS * entry point, both manually inserted.
70D7D CD0EX
70D80 D0=HEX 2F976
70D87 A=0 A
70D89 A=DAT0 1
70D8D CD0EX
70D90 A=A+1 A
70D92 GOSBVL 1B31B
70D99 C=A W
70D9C GOVLNG 0F216

70DA3 SETHEX
70DA5 P= 0
70DA7 LCHEX 3501
70DAD GOVLNG 0939A
70DB4 GOVLNG 05493
70DBB GOSBVL 0369D
70DC2 GOVLNG 03172
70DC9 GOTO 70DA3
70DCD dOFFST FFFE7 * offset to decompile routine
70DD2 pOFFST FFFE9 * offset to parse routine
SCMD * entry point, all three manually inserted.
70DD7 GOSBVL 0F186
70DDE GOSBVL 0BD1C
70DE5 GOSBVL 1B223
70DEC GOC 70DF2
70DEF GONC 70DC9
70DF2 C=0 W
70DF5 ?A=0 A
70DF8 GOYES 70DC9
70DFA A=A-1 A
70DFC LCHEX F
70DFF ?C<A A
70E02 GOYES 70DC9
70E04 D1=HEX 2F976
70E0B DAT1=A 1
70E0F C=A A
70E11 C=C+1 A
70E13 C=C+C A
70E15 B=C A
70E17 C=C+C A
70E19 B=B+C A
70E1B D1=HEX 2F576

```

70E22 C=DAT1 A
70E25 D1=HEX 2F580
70E2C C=B+C A
70E2E DAT1=C A
70E31 D1=D1+ 5
70E34 DAT1=C A
70E37 D1=D1+ 5
70E3A DAT1=C A
70E3D D1=HEX 2F576
70E44 C=DAT1 A
70E47 D1=C
70E4A LCHEX 003000
70E52 DAT1=C B
70E56 D1=D1+ B
70E59 A=A-1 A
70E5B GONC 70E52
70E5E GOVLNG 08A48
*****  

70E65 dOFFST 000AB      * offset to decompile routine
70E6A pOFFST 0009F      * offset to parse routine
UNPRV          * entry point, all three manually inserted.
70E6F GOSBVL 0A35D
70E76 GONC 70EA0
70E79 GOSBVL 09F2D
70E80 GOC 70EC0
70E83 C=D S
70E86 C=C+1 S
70E89 GOC 70EA0
70E8C C=C+1 S
70E8F CSRB
70E92 CSRB
70E95 ?C=0 S
70E98 GOYES 70EA0
70E9A GOLONG 70CAD
70EA0 GOSBVL 09F63
70EA7 GOC 70EC0
70EAA GOSBVL 0A5F7
70EB1 GONC 70EFC
70EB4 GOSBVL 06BFB
70EBB ?SB=0
70EBE GOYES 70EC7
70EC0 GOVLNG 0939A
70ECT7 GOSBVL 11053
70ECE GONC 70EDD
70ED1 GOSBVL 06A25
70ED8 ?C#0 P
70EDB GOYES 70EE4
70EDD GOVLNG 0A5CA
70EE4 D1=D1+ 4
70EE7 A=DAT1 B
70EEA LCHEX D
70EED A=C&A P
70EF1 DAT1=A 1
70EF5 GOVLNG 0A060
70EFC SETHEX
70EFE P= 0
70F00 LCHEX 3503
70F06 GONC 70EC0

```

70F09 GOVLNG 03B82

70F10 GOVLNG 054E5

Syntax description for functions and statements in DISLEX lex file.

CMDST function (no parameters).

This function returns the present size of the command stack.

SYNTAX: CMDST

GETNIB function (one numeric parameter).

Given a decimal address (0(=ADDR(=FFFFF hex) function returns the decimal value of the nibble at this address.

Function errors out if address negative or beyond 1048575 (dec) with error message 53001: DISS ERR: Invalid Arg

SYNTAX: N=GETNIB(A)
 DISP GETNIB(12345) 8

KEYWAIT\$ function (no parameters).

Function operates the same as mainframe function KEY\$ except CPU is put in low power mode in between keystrokes.

SYNTAX: Q\$=KEYWAIT\$

RCOPY statement (one argument: file specifier).

Statement is a partial duplicate of the mainframe statement COPY with the same syntax, except external file specifiers are illegal.

If illegal file specifier is used, statement errors out with error message 53002: DISS ERR: Invalid Filespec

Only allowed file specifiers are 'none', :MAIN, :PORT(x).

Statement is designed to copy files from RAM --> RAM or
ROM --> RAM,

without respecting the private status of the source file.

Statement works on mainframe files only, no poll handler incorporated.

SYNTAX: RCOPY DCLEX
 RCOPY AMPILEX:PORT(3) TO AMPI
 RCOPY FORTHROM:PORT(4)

REV\$ function (one string parameter).

Function returns the reverse of the source string.

SYNTAX: DISP REV\$("DISASS") SSASID
 A\$=REV\$(F\$)

```
*****
SETCMD      statement      (one numeric argument).

Statement set maximum size of command stack to value specified. (0<value<=16)
Command stack will be cleared when setting size by this statement.
If value negative or beyond 16 errors out with error message 53001:
        DISS ERR: Invalid Arg

SYNTAX:      SETCMD 15
*****

UNPRIV      statement      (one argument: filespecifier).

Statement clears private status of file in RAM.
Errors out when an external file or a file in ROM is specified.
Only allowed file specifiers are: 'none', :MAIN, :PORT(x).
If illegal device is specified, error message 53002 will appear:
        DISS ERR: Invalid Filespec
If file is not in RAM, error message 53003 will appear:
        DISS ERR: File not in RAM
Statement works on mainframe files only, no poll handler incorporated.

SYNTAX:      UNPRIV  filename
              UNPRIV  filename:PORT
*****
```

TOOLS7C TEXT 8960 06/13/85 17:00

Source file for DISLEX lex file.

```
LEX      'DISLEX'
TITLE   DISLEX,    J.G.Buitenhuis
ID      #35
MSG     MSGTBL
POLL    POLHND
ALINFO  EQU      #01531
DEFFIL  EQU      #01131
BSERR   EQU      #0939A
tKYSck  EQU      #0A35D
FSPECx  EQU      #09F2D
SVINFO  EQU      #0845A
NXTSTM  EQU      #08A48
FILFIL  EQU      #011CE
CPYERR  EQU      #08407
FINDFS  EQU      #08448
GETPR1  EQU      #06BFB
REENT   EQU      #082EB
CPYEXT  EQU      #08401
COPYDC  EQU      #05722
COPYP   EQU      #03B0C
SLEEP   EQU      #006C2
KEY$    EQU      #1ACA8
CKSREQ  EQU      #00721
POP1R   EQU      #0E8FD
FLTDH   EQU      #1B223
HDFLT   EQU      #1B31B
FNRTN4  EQU      #0F238
REV$    EQU      #1B38E
EXPR    EQU      #0F23C
MAXCMD  EQU      #2F976
FNRTN1  EQU      #0F216
NUMCK   EQU      #0369D
RESPTR  EQU      #03172
FIXDC   EQU      #05493
EXPEXC  EQU      #0F186
POP1N   EQU      #0BD1C
CLCBFR  EQU      #2F576
RAWBFR  EQU      #2F580
eIVARG  EQU      #0000B
eFSPEC  EQU      #0003A
IFTYPH  EQU      #00004
FINDF+  EQU      #09F63
RAMROM  EQU      #0A5F7
FTYPFD  EQU      #11053
RDENTY  EQU      #06A25
EDITXE  EQU      #0A5CA
PURGDN  EQU      #0A060
PRIVTP  EQU      #03B82
NAMEdc  EQU      #054E5
```

```

ENTRY  CPY
CHAR   #D
ENTRY  KEYW
CHAR   #F
ENTRY  PEEK
CHAR   #F
ENTRY  REV
CHAR   #F
ENTRY  CMDS
CHAR   #F
ENTRY  SCMD
CHAR   #D
ENTRY  UNPRV
CHAR   #D
*****
KEY   'CMDST'
TOKEN 5
KEY   'GETNIB'
TOKEN 3
KEY   'KEYWAIT$'
TOKEN 2
KEY   'RCOPY'
TOKEN 1
KEY   'REV$'
TOKEN 4
KEY   'SETCMD'
TOKEN 6
KEY   'UNPRIV'
TOKEN 7
ENDTXT
*****
POLHND ?B=0    B
      GOYES VER0
      GONC  VER1
VER0   C=R3
      D1=C
      A=R2
      D1=D1-(Ve)-(Vs)-2
      CD1EX
      ?A>C   A
      GOYES VER1
      D1=C
      R3=C
Vs    LCASC  'DIS:B'
Ve    DAT1=C (Ve)-(Vs)-2
VER1  RTNSXM
*****
MSGTBL CON(2) 00
      CON(2) 03
msg00 CON(2) (msg01)-(*)
      CON(2) 00
      CON(1) 4
      NIBASC 'DISS '
      CON(1) 12
msg01 CON(2) (msg02)-(*)
      CON(2) 01
      CON(1) 14
      CON(2) eIVARG

```

```

        CON(1) 12
msg02  CON(2) (msg03)-(*)
        CON(2) 02
        CON(1) 14
        CON(2) eFSPEC
        CON(1) 12
msg03  CON(2) (msgend)-(*)
        CON(2) 03
        CON(1) 11
        CON(1) 14
        NIBASC 'File not'
        NIBASC ' in RAM'
        CON(1) 12
msgend NIBHEX FF
*****
REL(5) dCOPY
REL(5) pCOPY
CPY
P=      1
GOSBVL =ALINFO
GOC     CPYERX
CPY010 D=0    W
LCASC   , ,
R0=C
A=DAT0 B
LCHEX  F3
?C#A   B
GOYES  CPY030
D0=D0+ 2
?ST=1  3
GOYES  CPY040
CPY020 GOSBVL =DEFFIL
GONC   CPY065
CPY030 LCHEX F0
?C>A   B
GOYES  CPY040
?ST=1  3
GOYES  CPY050
GONC   CPY020
CPY040 LCHEX D0
?C#A   B
GOYES  CPY060
LCHEX  7
D=C    P
CPY050 D0=D0+ 2
A=0    W
GONC   CPY070
CPYERX GOVLNG =BSERR
CPY060 GOSBVL =tKYSck
GONC   CPY065
GOSBVL =FSPECx
GOC    CPYERX
CPY065 DSLC
CPY070 GOSBVL =SVINFO
?ST=1  3
GOYES  CPY081
ST=1   3
GOTO   CPY010

```

```

CPY081 GOSUB  COPYu
      GOC    CPYERX
nxtstm GOVLNG =NXTSTM
COPYu GOSBVL =FILFIL
      ?ST=0  0
      GOYES CPY100
illegl SETHEX
      P=    0
      LCHEX 3502
      GONC  CPYERX
CPY100 GOSBVL =FINDFS
      GOC    CPYER1
      CD1EX
      R3=C
      CD1EX
      GOSBVL =GETPR1
      GOVLNG =REENT
EXIT   GOVLNG =CPYEXT
CPYER1 GOVLNG =CPYERR
dCOPY  GOVLNG =COPYDC
pCOPY  GOVLNG =COPYP
*****
NIBHEX 00
KEYW
      CD0EX
      R0=C
      CD1EX
      R1=C
LBL01 GOSBVL =SLEEP
      GOC    LBL02
      C=R1
      D1=C
      C=R0
      D0=C
      GOVLNG =KEY$
LBL02 GOSBVL =CKSREQ
      GOTO  LBL01
*****
NIBHEX 811
PEEK
      GOSBVL POP1R
      GOSBVL FLTDX
      GOC    ok
      GONC  sorry
ok    CD1EX
      CR0EX
      D1=A
      A=0    A
      A=DAT1 1
      GOSBVL HDFLT
      ACEX  W
      AR0EX
      AD1EX
      GOVLNG FNRTN4
*****
NIBHEX 411
REV
      GOSBVL =REV$
```

```

GOVLNG =EXPR
*****
NIBHEX 00
CMDS
CD0EX
D0=(5) MAXCMD
A=0 A
A=DAT0 1
CD0EX
A=A+1 A
GOSBVL HDFLT
C=A W
GOVLNG FNRTN1
*****
sorry SETHEX
P= 0
LCHEX 3501
GOVLNG =BSERR
dSCMD GOVLNG =FIXDC
pSCMD GOSBVL =NUMCK
GOVLNG =RESPTR
SORRY GOTO sorry
REL(5) dSCMD
REL(5) pSCMD
SCMD
GOSBVL =EXPEXC
GOSBVL =POP1N
GOSBVL =FLTDH
GOC ONWARD
GONC SORRY
ONWARD C=0 W
?A=0 A
GOYES SORRY
A=A-1 A
LCHEX F
?A>C A
GOYES SORRY
D1=(5) =MAXCMD
DAT1=A 1
C=A A
C=C+1 A
C=C+C A
B=C A
C=C+C A
B=B+C A
D1=(5) =CLCBFR
C=DAT1 A
D1=(5) =RAWBFR
C=C+B A
DAT1=C A
D1=D1+ 5
DAT1=C A
D1=D1+ 5
DAT1=C A
D1=(5) =CLCBFR
C=DAT1 A
D1=C
LCHEX 003000

```

```

LOOP  DAT1=C 6
      D1=D1+ 6
      A=A-1  A
      GONC  LOOP
      GOVLNG =NXTSTM
*****
      REL(5) dNAME
      REL(5) pUNPV
UNPRV
      GOSBVL =tKY5ck
      GONC  SCR10
      GOSBVL =FSPECx
      GOC   SCERR
      C=D   S
      C=C+1 S
      GOC   SCR10
      C=C+1 S
      CSRB
      CSRB
      ?C=0  S
      GOYES SCR10
      GOLONG illegal
SCR10 GOSBVL =FINDF+
      GOC   SCERR
      GOSBVL =RAMROM
      GONC  INROM
      GOSBVL =GETPR1
      ?SB=0
      GOYES SCR20
SCERR  GOVLNG =BSERR
SCR20 GOSBVL =FTYPFD
      GONC  PRV20
      GOSBVL =RDENTY
      ?C#0  P
      GOYES SCR65
PRV20 GOVLNG =EDITXE
SCR65 D1=D1+ =1FTYPH
      A=DAT1 B
      LCHEX  D
      A=A&C P
      DAT1=A 1
      GOVLNG =PURGDN
INROM  SETHEX
      P=    0
      LCHEX  3503
      GONC  SCERR
pUNPV  GOVLNG =PRIVTP
dNAME  GOVLNG =NAMEdc
*****
END

```

0001 00000 LEX 'DISLEX'
0001 00000 449435C4 NIBASC 'DISLEX '
00008 54850202
0001 00010 802E NIBHEX 802E
0001 00014 00 CON(2) 0
0001 00016 04 CON(2) #40
0001 00018 90 CON(2) #09
0001 0001A 31 CON(2) #13
0001 0001C 60 CON(2) #06
0001 0001E 58 CON(2) #85
0001 00020 46400 REL(5) FiLeNd
0002 00025 TITLE DISLEX, J.G.Buitenhuis
0003 00025 ID #35
0003 00025 53 CON(2) #35
0003 00027 10 CON(2) 0001
0003 00029 70 CON(2) 0007
0003 0002B 00000 CON(5) 0
0003 00030 F NIBHEX F
0003 00031 D400 REL(4) 1+TxTbSt
0004 00035 MSG MSGTBL
0004 00035 6E00 REL(4) MSGTBL
0005 00039 POLL POLHND
0005 00039 CA000 REL(5) POLHND
0006 0003E ALINFO EQU #01531
0007 0003E DEFFIL EQU #01131
0008 0003E BSERR EQU #0939A
0009 0003E tKYSck EQU #0A35D
0010 0003E FSPECx EQU #09F2D
0011 0003E SVINFO EQU #0845A
0012 0003E NXTSTM EQU #08A48
0013 0003E FILFIL EQU #011CE
0014 0003E CPYERR EQU #08407
0015 0003E FINDFS EQU #08448
0016 0003E GETPR1 EQU #06BFB
0017 0003E REENT EQU #082EB
0018 0003E CPYEXT EQU #08401
0019 0003E COPYDC EQU #05722
0020 0003E COPYP EQU #03B0C
0021 0003E SLEEP EQU #006C2
0022 0003E KEY\$ EQU #1ACA8
0023 0003E CKSREQ EQU #00721
0024 0003E POP1R EQU #0E8FD
0025 0003E FLTDH EQU #1B223
0026 0003E HDFLT EQU #1B31B
0027 0003E FNRTN4 EQU #0F238
0028 0003E REV\$ EQU #1B38E
0029 0003E EXPR EQU #0F23C
0030 0003E MAXCMD EQU #2F976
0031 0003E FNRTN1 EQU #0F216
0032 0003E NUMCK EQU #0369D
0033 0003E RESPTR EQU #03172
0034 0003E FIXDC EQU #05493
0035 0003E EXPPEXC EQU #0F186
0036 0003E POP1N EQU #0BD1C
0037 0003E CLCBFR EQU #2F576
0038 0003E RAWBFR EQU #2F580

```
0039 0003E      eIVARG EQU      #0000B
0040 0003E      eFSPEC EQU      #0003A
0041 0003E      1FTYPH EQU      #00004
0042 0003E      FINDF+ EQU      #09F63
0043 0003E      RAMROM EQU      #0A5F7
0044 0003E      FTYPFD EQU      #11053
0045 0003E      RDENTY EQU      #06A25
0046 0003E      EDITXE EQU      #0A5CA
0047 0003E      PURGDN EQU      #0A060
0048 0003E      PRIVTP EQU      #03B82
0049 0003E      NAMEEdc EQU      #054E5
0050 0003E      ****
0051 0003E      ENTRY CPY
0051 0003E      * * * M A I N   T A B L E * * *
0051 0003E 000    CON(3) (TxEn01)-(TxTbSt)
0051 00041 F2100 REL(5) CPY
0052 00046       CHAR  #D
0052 00046 D     CON(1) #D
0053 00047       ENTRY KEYW
0053 00047 D00   CON(3) (TxEn02)-(TxTbSt)
0053 0004A C1200 REL(5) KEYW
0054 0004F       CHAR  #F
0054 0004F F     CON(1) #F
0055 00050       ENTRY PEEK
0055 00050 C10   CON(3) (TxEn03)-(TxTbSt)
0055 00053 A4200 REL(5) PEEK
0056 00058       CHAR  #F
0056 00058 F     CON(1) #F
0057 00059       ENTRY REV
0057 00059 F20   CON(3) (TxEn04)-(TxTbSt)
0057 0005C E7200 REL(5) REV
0058 00061       CHAR  #F
0058 00061 F     CON(1) #F
0059 00062       ENTRY CMDS
0059 00062 C30   CON(3) (TxEn05)-(TxTbSt)
0059 00065 58200 REL(5) CMDS
0060 0006A       CHAR  #F
0060 0006A F     CON(1) #F
0061 0006B       ENTRY SCMD
0061 0006B 740   CON(3) (TxEn06)-(TxTbSt)
0061 0006E 6D200 REL(5) SCMD
0062 00073       CHAR  #D
0062 00073 D     CON(1) #D
0063 00074       ENTRY UNPRV
0063 00074 650   CON(3) (TxEn07)-(TxTbSt)
0063 00077 56300 REL(5) UNPRV
0064 0007C       CHAR  #D
0064 0007C D     CON(1) #D
0065 0007D      ****
0066 0007D      KEY    'CMDST'
0066 0007D      * * * T E X T   T A B L E * * *
0066 0007D      TxTbSt
0066 0007D      TxEn01
```

PAGE 003 DISLEX, J.G.Buitenhuis
Forth Assembler

0066 0007D 9 CON(1) 09
0066 0007E 34D44435 NIBASC 'CMDST'
00086 45
0067 00088 TOKEN 5
0067 00088 50 CON(2) 5
0068 0008A KEY 'GETNIB'
0068 0008A TxEn02
0068 0008A B CON(1) 11
0068 0008B 745445E4 NIBASC 'GETNIB'
00093 9424
0069 00097 TOKEN 3
0069 00097 30 CON(2) 3
0070 00099 KEY 'KEYWAIT\$'
0070 00099 TxEn03
0070 00099 F CON(1) 15
0070 0009A B4549575 NIBASC 'KEYWAIT\$'
000A2 14944542
0071 000AA TOKEN 2
0071 000AA 20 CON(2) 2
0072 000AC KEY 'RCOPY'
0072 000AC TxEn04
0072 000AC 9 CON(1) 09
0072 000AD 2534F405 NIBASC 'RCOPY'
000B5 95
0073 000B7 TOKEN 1
0073 000B7 10 CON(2) 1
0074 000B9 KEY 'REV\$'
0074 000B9 TxEn05
0074 000B9 7 CON(1) 07
0074 000BA 25546542 NIBASC 'REV\$'
0075 000C2 TOKEN 4
0075 000C2 40 CON(2) 4
0076 000C4 KEY 'SETCMD'
0076 000C4 TxEn06
0076 000C4 B CON(1) 11
0076 000C5 35544534 NIBASC 'SETCMD'
000CD D444
0077 000D1 TOKEN 6
0077 000D1 60 CON(2) 6
0078 000D3 KEY 'UNPRIV'
0078 000D3 TxEn07
0078 000D3 B CON(1) 11
0078 000D4 55E40525 NIBASC 'UNPRIV'
000DC 9465
0079 000E0 TOKEN 7
0079 000E0 70 CON(2) 7
0080 000E2 ENDTXT
0080 000E2 1FF NIBHEX 1FF
0081 000E5 *****
0082 000E5 969 POLHND ?B=0 B
0083 000E8 50 GOYES VER0
0084 000EA 5E2 GONC VER1
0085 000ED 11B VER0 C=R3

PAGE 004
Forth Assembler

DISLEX, J.G.Buitenhuis

0086 000F0 135	D1=C
0087 000F3 112	A=R2
0088 000F6 1CB	D1=D1-(Ve)-(Vs)-2
0089 000F9 137	CD1EX
0090 000FC 8B6	?A>C A
0091 000FF A1	GYES VER1
0092 00101 135	D1=C
0093 00104 10B	R3=C
0094 00107 3B24A335 Vs	LCASC ' DIS:B'
0095 00115 15DB	Ve DAT1=C (Ve)-(Vs)-2
0096 00119 00	VER1 RTNSXM
0097 0011B	*****
0098 0011B 00	MSGTBL CON(2) 00
0099 0011D 30	CON(2) 03
0100 0011F 01	msg00 CON(2) (msg01)-(*)
0101 00121 00	CON(2) 00
0102 00123 4	CON(1) 4
0103 00124 44943535	NIBASC 'DISS ,
0012C 02	
0104 0012E C	CON(1) 12
0105 0012F 80	msg01 CON(2) (msg02)-(*)
0106 00131 10	CON(2) 01
0107 00133 E	CON(1) 14
0108 00134 B0	CON(2) eIVARG
0109 00136 C	CON(1) 12
0110 00137 80	msg02 CON(2) (msg03)-(*)
0111 00139 20	CON(2) 02
0112 0013B E	CON(1) 14
0113 0013C A3	CON(2) eFSPEC
0114 0013E C	CON(1) 12
0115 0013F 52	msg03 CON(2) (msgend)-(*)
0116 00141 30	CON(2) 03
0117 00143 B	CON(1) 11
0118 00144 E	CON(1) 14
0119 00145 6496C656	NIBASC 'File not'
0014D 02E6F647	
0120 00155 0296E602	NIBASC ' in RAM'
0015D 2514D4	
0121 00163 C	CON(1) 12
0122 00164 FF	msgend NIBHEX FF
0123 00166	*****
0124 00166 0F000	REL(5) dCOPY
0125 0016B 2F000	REL(5) pCOPY
0126 00170	COPY
0127 00170 21	P= 1
0128 00172 8F13510	GOSBVL =ALINFO
0129 00179 455	60C CPYERX
0130 0017C AF3	CPY010 D=0 W
0131 0017F 330202	LCASC , ,
0132 00185 108	R0=C
0133 00188 14A	A=DAT0 B
0134 0018B 313F	LCHEX F3

PAGE 005
Forth Assembler

DISLEX, J.G.Buitenhuis

0135 0018F 966	?C#A	B
0136 00192 41	GOYES	CPY030
0137 00194 161	D0=D0+	2
0138 00197 873	?ST=1	3
0139 0019A D1	GOYES	CPY040
0140 0019C 8F13110	CPY020	GOSBVL =DEFFIL
0141 001A3 564	GONC	CPY065
0142 001A6 310F	CPY030	LCHEX F0
0143 001AA 9E2	?C>A	B
0144 001AD A0	GOYES	CPY040
0145 001AF 873	?ST=1	3
0146 001B2 41	GOYES	CPY050
0147 001B4 57E	GONC	CPY020
0148 001B7 310D	CPY040	LCHEX D0
0149 001BB 966	?C#A	B
0150 001BE 81	GOYES	CPY060
0151 001C0 307	LCHEX	7
0152 001C3 A87	D=C	P
0153 001C6 161	CPY050	D0=D0+ 2
0154 001C9 AF0	A=0	W
0155 001CC 502	GONC	CPY070
0156 001CF 8DA9390	CPYERX	GOVLNG =BSERR
0157 001D6 8FD53A0	CPY060	GOSBVL =tKYSck
0158 001DD 5C0	GONC	CPY065
0159 001E0 8FD2F90	GOSBVL	=FSPECx
0160 001E7 47E	GOC	CPYERX
0161 001EA 813	CPY065	DSLC
0162 001ED 8FA5480	CPY070	GOSBVL =SVINFO
0163 001F4 873	?ST=1	3
0164 001F7 90	GOYES	CPY081
0165 001F9 853	ST=1	3
0166 001FC 6F7F	GOTO	CPY010
0167 00200 7A00	CPY081	GOSUB COPYu
0168 00204 4AC	GOC	CPYERX
0169 00207 8D84A80	nxtstm	GOVLNG =NXTSTM
0170 0020E 8FEC110	COPYu	GOSBVL =FILFIL
0171 00215 860	?ST=0	0
0172 00218 F0	GOYES	CPY100
0173 0021A 04	illegal	SETHEX
0174 0021C 20	P=	0
0175 0021E 332053	LCHEX	3502
0176 00224 5AA	GONC	CPYERX
0177 00227 8F84480	CPY100	GOSBVL =FINDFS
0178 0022E 402	GOC	CPYER1
0179 00231 137	CD1EX	
0180 00234 10B	R3=C	
0181 00237 137	CD1EX	
0182 0023A 8FBFB60	GOSBVL	=GETPR1
0183 00241 8DBE280	GOVLNG	=REENT
0184 00248 8D10480	EXIT	GOVLNG =CPYEXT
0185 0024F 8D70480	CPYER1	GOVLNG =CPYERR
0186 00256 8D22750	DCOPY	GOVLNG =COPYDC
0187 0025D 8DC0B30	PCOPY	GOVLNG =COPYP

PAGE 006
Forth Assembler

DISLEX, J.G.Buitenhuis

```
0188 00264 *****  
0189 00264 00 NIBHEX 00  
0190 00266 KEYW  
0191 00266 136 CD0EX  
0192 00269 108 R0=C  
0193 0026C 137 CD1EX  
0194 0026F 109 R1=C  
0195 00272 8F2C600 LBL01 GOSBVL =SLEEP  
0196 00279 451 GOC LBL02  
0197 0027C 119 C=R1  
0198 0027F 135 D1=C  
0199 00282 118 C=R0  
0200 00285 134 D0=C  
0201 00288 8D8ACA1 GOVLNG =KEY$  
0202 0028F 8F12700 LBL02 GOSBVL =CKSREQ  
0203 00296 6BDF GOTO LBL01  
0204 0029A *****  
0205 0029A 811 NIBHEX 811  
0206 0029D PEEK  
0207 0029D 8FDF8E0 GOSBVL POP1R  
0208 002A4 8F322B1 GOSBVL FLTDH  
0209 002AB 450 GOC ok  
0210 002AE 516 GONC sorry  
0211 002B1 137 ok CD1EX  
0212 002B4 128 CR0EX  
0213 002B7 131 D1=A  
0214 002BA D0 A=0 A  
0215 002BC 15B0 A=DAT1 1  
0216 002C0 8FB13B1 GOSBVL HDFLT  
0217 002C7 AFE ACEX W  
0218 002CA 120 AR0EX  
0219 002CD 133 AD1EX  
0220 002D0 8D832F0 GOVLNG FNRTN4  
0221 002D7 *****  
0222 002D7 411 NIBHEX 411  
0223 002DA REV  
0224 002DA 8FE83B1 GOSBVL =REV$  
0225 002E1 8DC32F0 GOVLNG =EXPR  
0226 002E8 *****  
0227 002E8 00 NIBHEX 00  
0228 002EA CMDS  
0229 002EA 136 CD0EX  
0230 002ED 1B679F2 D0=(5) MAXCMD  
0231 002F4 D0 A=0 A  
0232 002F6 15A0 A=DAT0 1  
0233 002FA 136 CD0EX  
0234 002FD E4 A=A+1 A  
0235 002FF 8FB13B1 GOSBVL HDFLT  
0236 00306 AF6 C=A W  
0237 00309 8D612F0 GOVLNG FNRTN1  
0238 00310 *****  
0239 00310 04 sorry SETHEX  
0240 00312 20 P= 0
```

0241 00314 331053	LCHEX 3501
0242 0031A 8DA9390	GOVLNG =BSERR
0243 00321 8D39450	dSCMD GOVLNG =FIXDC
0244 00328 8FD9630	pSCMD GOSBVL =NUMCK
0245 0032F 8D27130	GOVLNG =RESPTR
0246 00336 69DF	SORRY GOTO sorry
0247 0033A 7EFFF	REL(5) dSCMD
0248 0033F 9EFFF	REL(5) pSCMD
0249 00344	SCMD
0250 00344 8F681F0	GOSBVL =EXPEXC
0251 0034B 8FC1DB0	GOSBVL =POP1N
0252 00352 8F322B1	GOSBVL =FLTDH
0253 00359 450	GOC ONWARD
0254 0035C 59D	GONC SORRY
0255 0035F AF2	ONWARD C=0 W
0256 00362 8A8	?A=0 A
0257 00365 1D	GOYES SORRY
0258 00367 CC	A=A-1 A
0259 00369 30F	LCHEX F
0260 0036C 8B6	?A>C A
0261 0036F 7C	GOYES SORRY
0262 00371 1F679F2	D1=(5) =MAXCMD
0263 00378 1590	DAT1=A 1
0264 0037C D6	C=A A
0265 0037E E6	C=C+1 A
0266 00380 C6	C=C+C A
0267 00382 D5	B=C A
0268 00384 C6	C=C+C A
0269 00386 C1	B=B+C A
0270 00388 1F675F2	D1=(5) =CLCBFR
0271 0038F 147	C=DAT1 A
0272 00392 1F085F2	D1=(5) =RAWBFR
0273 00399 C9	C=C+B A
0274 0039B 145	DAT1=C A
0275 0039E 174	D1=D1+ 5
0276 003A1 145	DAT1=C A
0277 003A4 174	D1=D1+ 5
0278 003A7 145	DAT1=C A
0279 003AA 1F675F2	D1=(5) =CLCBFR
0280 003B1 147	C=DAT1 A
0281 003B4 135	D1=C
0282 003B7 35000300	LCHEX 003000
0283 003BF 15D5	LOOP DAT1=C 6
0284 003C3 175	D1=D1+ 6
0285 003C6 CC	A=A-1 A
0286 003C8 56F	GONC LOOP
0287 003CB 8D84A80	GOVLNG =NXTSTM
0288 003D2	*****
0289 003D2 BA000	REL(5) dNAME
0290 003D7 F9000	REL(5) pUNPV
0291 003DC	UNPRV
0292 003DC 8FD53A0	GOSBVL =tKYSck
0293 003E3 592	GONC SCR10

PAGE 008
Forth Assembler

DISLEX, J.G.Buitenhuis

0294 003E6 8FD2F90	GOSBVL	=FSPECx
0295 003ED 4F3	GOC	SCERR
0296 003F0 ACB	C=D	S
0297 003F3 B46	C=C+1	S
0298 003F6 461	GOC	SCR10
0299 003F9 B46	C=C+1	S
0300 003FC 81E	CSRB	
0301 003FF 81E	CSRB	
0302 00402 94A	?C=0	S
0303 00405 80	GOYES	SCR10
0304 00407 8C11EF	GOLONG	illegal
0305 0040D 8F36F90	SCR10	GOSBVL =FINDF+
0306 00414 481	GOC	SCERR
0307 00417 8F7F5A0	GOSBVL	=RAMROM
0308 0041E 5A4	GONC	INROM
0309 00421 8FBFB60	GOSBVL	=GETPR1
0310 00428 832	?SB=0	
0311 0042B 90	GOYES	SCR20
0312 0042D 8DA9390	SCERR	GOVLNG =BSERR
0313 00434 8F35011	SCR20	GOSBVL =FTYPFD
0314 0043B 5E0	GONC	PRV20
0315 0043E 8F52A60	GOSBVL	=RDENTY
0316 00445 90E	?C#0	P
0317 00448 90	GOYES	SCR65
0318 0044A 8DAC5A0	PRV20	GOVLNG =EDITXE
0319 00451 173	SCR65	D1=D1+ =IFTYPH
0320 00454 14B	A=DAT1	B
0321 00457 30D	LCHEX	D
0322 0045A 0E06	A=A&C	P
0323 0045E 1590	DAT1=A	1
0324 00462 8D060A0	GOVLNG	=PURGDN
0325 00469 04	INROM	SETHEX
0326 0046B 20	P=	0
0327 0046D 333053	LCHEX	3503
0328 00473 59B	GONC	SCERR
0329 00476 8D28B30	pUNPV	GOVLNG =PRIVTP
0330 0047D 8D5E450	dNAME	GOVLNG =NAMEdc
0331 00484		*****
0332 00484		END

PAGE 009
Forth Assembler

DISLEX, J.G.Buitenhuis
**** SYMBOL TABLE ****

ALINFO 01531
BSERR 0939A
CKSREQ 00721
CLCBFR 2F576
CMDS 002EA
COPYDC 05722
COPYP 03B0C
COPYu 0020E
CPY 00170
CPY010 0017C
CPY020 0019C
CPY030 001A6
CPY040 001B7
CPY050 001C6
CPY060 001D6
CPY065 001EA
CPY070 001ED
CPY081 00200
CPY100 00227
CPYER1 0024F
CPYERR 08407
CPYERX 001CF
CPYEXT 08401
DEFFIL 01131
EDITXE 0A5CA
EXIT 00248
EXPEXC 0F186
EXPR 0F23C
FILFIL 011CE
FINDF+ 09F63
FINDFS 08448
FIXDC 05493
FLTDH 1B223
FNRTN1 0F216
FNRTN4 0F238
FSPECx 09F2D
FTYPFD 11053
FiLeNd 00484
GETPR1 06BFB
HDFLT 1B31B
INROM 00469
KEY\$ 1ACA8
KEYW 00266
LBL01 00272
LBL02 0028F
LOOP 003BF
MAXCMD 2F976
MSGTBL 0011B
NAMEdc 054E5
NUMCK 0369D
NXTSTM 08A48
ONWARD 0035F
PEEK 0029D

PAGE 010
Forth Assembler

DISLEX, J.G.Buitenhuis
**** SYMBOL TABLE ****

POLHND	00229
POP1N	48412
POP1R	59645
PRIVTP	15234
PRV20	01098
PURGDN	41056
RAMROM	42487
RAWBFR	93920
RDENTY	27173
REENT	33515
RESPTR	12658
REV	00730
REV\$	11502
SCERR	01069
SCMD	00836
SCR10	01037
SCR20	01076
SCR65	01105
SLEEP	01730
SORRY	00822
SVINFO	33882
TxEn01	00125
TxEn02	00138
TxEn03	00153
TxEn04	00172
TxEn05	00185
TxEn06	00196
TxEn07	00211
TxTbSt	00125
UNPRV	00988
VER0	00237
VER1	00281
Ve	00277
Vs	00263
dCOPY	00598
dNAME	01149
dSCMD	00801
eFSPEC	00058
eIVARG	00011
illegal	00538
IFTYPh	00004
msg00	00287
msg01	00303
msg02	00311
msg03	00319
msgend	00356
nxtstm	00519
ok	00689
pCOPY	00605
pSCMD	00808
pUNPV	01142
sorry	00784
tKYSck	41821

PAGE 011 DISLEX, J.G.Buitenhuis
Forth Assembler **** SYMBOL TABLE ****

SOURCE : TOOLS7C

OBJECT : DISLEX

LISTING : :HP2225B

DATE : 09:53:24 on 85/06/13

ERRORS : 000

--TEXT TABLE--
#36700

Keyword CMDST
Token= 5

Keyword GETNIB
Token= 3

Keyword KEYWAIT\$
Token= 2

Keyword RCOPY
Token= 1

Keyword REV\$
Token= 4

Keyword SETCMD
Token= 6

Keyword UNPRIV
Token= 7

#36767
END of TEXT TABLE

--MESSAGE TABLE--
#3679E

Low Message: #00 - 0
High Message: #03 - 3
xxxxxxxxxxxxxxxxxxxxxx
Block Length= 16 nibs
Message ID: #00 - 0

-Text Cell-
DISS
xxxxxxxxxxxxxxxxxxxxxx
Block Length= 8 nibs
Message ID: #01 - 1

-Mainframe Bldg. Block-
#0B - 11
xxxxxxxxxxxxxxxxxxxxxx
Block Length= 8 nibs
Message ID: #02 - 2

-Mainframe Bldg. Block-
#3A - 58
xxxxxxxxxxxxxxxxxxxxxx
Block Length= 37 nibs
Message ID: #03 - 3

-Long Text Cell-
File not in RAM
xxxxxxxxxxxxxxxxxxxxxx
#367E8
END of MESSAGE TABLE

--FILE HEADER--
#36683

File Name: DISLEX
Flags= 0
Copy Code= 0
Creation Time 10:16
Creation Date 85/06/13
Next File at: #36807

#36607
END of FILE HEADER

LEX ID: #35 - 53
Low Token= 1
High Token= 7
LEX Table Link #000000
No Speed Table
Text Table: #36700
Message Table: #3679E
Poll Handler: #36768

--MAIN TABLE--
#366C1

KEYWORD 001 at #36700
EXECUTION start #367F3
Characterization Nib #D

KEYWORD 002 at #36760
EXECUTION start #368E9
Characterization Nib #F

KEYWORD 003 at #36710
EXECUTION start #36920
Characterization Nib #F

KEYWORD 004 at #3672F
EXECUTION start #3695D
Characterization Nib #F

KEYWORD 005 at #36730
EXECUTION start #3696D
Characterization Nib #F

KEYWORD 006 at #36747
EXECUTION start #369C7
Characterization Nib #D

KEYWORD 007 at #36756
EXECUTION start #36A5F
Characterization Nib #D

#366FF
END of MAIN TABLE

SORTER BASIC 636 06/12/85 09:55

```
10 ! SORTER, program to set offset in MAIN TABLE according to place
20 ! of keyword in TEXT TABLE.
30 DESTROY ALL @ STD @ DELAY 0,0 @ OPTION BASE 1
40 DISP CHR$(27) & "=";
50 INPUT " LEX File name : ",?';F$
60 INPUT " NR. OF FUNCTIONS : ",?';N
70 A$=ADDR$(F$) @ A0=HTD(A$)
80 Q$=PEEK$(DTH$(A0+16),4) @ DISP
90 IF Q$#1802E' THEN BEEP @ DISP " NOT A LEX FILE" @ END
100 Q$=PEEK$(DTH$(A0+48),1)
110 IF Q$="F" THEN A0=A0+62 ELSE A0=A0+141
120 DIM P$(N)[3] @ A=A0
130 FOR X=1 TO N
140 P$(X)=PEEK$(DTH$(A),3) @ A=A+9
150 NEXT X @ DISP
160 FOR X=1 TO N
170 X$='0000'&STR$(X) @ X$=X$[LEN(X$)-2,LEN(X$)]
180 DISP "Sequence of KEYWORD ";X$;" in TEXT TABLE ---> ";
190 INPUT ',?';P
200 W$=P$(P) @ Z$=DTH$(A0)
210 DISP "Address: ";Z$;"      offset: ";REV$(W$) @ A0=A0+9
220 POKE Z$,W$ @ NEXT X
230 DISP @ BEEP 1500 @ DISP "DONE"
240 END
```

LEX File name : DISLEX
NR. OF FUNCTIONS : 7

```
Sequence of KEYWORD 001 in TEXT TABLE ---> 4
Address: 2FA99      offset: 02F
Sequence of KEYWORD 002 in TEXT TABLE ---> 3
Address: 2FAA2      offset: 01C
Sequence of KEYWORD 003 in TEXT TABLE ---> 2
Address: 2FAAB      offset: 00D
Sequence of KEYWORD 004 in TEXT TABLE ---> 5
Address: 2FAB4      offset: 03C
Sequence of KEYWORD 005 in TEXT TABLE ---> 1
Address: 2FABD      offset: 000
Sequence of KEYWORD 006 in TEXT TABLE ---> 6
Address: 2FAC6      offset: 047
Sequence of KEYWORD 007 in TEXT TABLE ---> 7
Address: 2FACF      offset: 056
```

DONE

LEXMAIN TEXT 460 06/12/85 15:40

!!! Do NOT use this function!!!
You will be heading for MEMORY LOST.
Have to check with Nick Reid what is
wrong here.
If I can get this function to work it will
replace my "normal" basic program SORTER

: CHECKLEX
FINDF DUP 0=
ABORT" File not found "
DUP 10 + @ E208 - DUP @ = (???)
ABORT" Wrong file type "
;

: GETSTART
27 + DUP C@ SWAP Z+ DUP C@ SWAP 7 + DUP N@
IF 1+ ELSE 80 + THEN DUP DUP 4N@ 1- + SWAP D + ;

: OFFSET
2DUP SWAP - SWAP DUP N@ Z+ + DUP R> Z+ SWAP R> C@ ;

: UPDATE
5 PICK - 9 * 6 PICK + DUP @ FF000 AND ROT + SWAP ! ;

: LEXMAIN
CHECKLEX GETSTART 4 ROLL ROT DUP 5 ROLL 1+ 6 PICK
DO OFFSET UPDATE LOOP SP! ;

--TEXT TABLE--

#36700

--FILE HEADER--

#36683

File Name: DISLEX
Flags= 0
Copy Code= 0
Creation Time 10:16
Creation Date 85/06/13
Next File at: #36807

#366A7

END of FILE HEADER

LEX ID: #35 - 53
Low Token= 1
High Token= 7
LEX Table Link #00000
HO Speed Table
Text Table: #36700
Message Table: #3679E
Poll Handler: #36768

--MAIN TABLE--

#366C1

KEYWORD 001 at #3672F
EXECUTION start #367F3
Characterization Nib #D

KEYWORD 002 at #36710
EXECUTION start #368E9
Characterization Nib #F

KEYWORD 003 at #3678D
EXECUTION start #36920
Characterization Nib #F

KEYWORD 004 at #36730
EXECUTION start #3695D
Characterization Nib #F

KEYWORD 005 at #36700
EXECUTION start #3696D
Characterization Nib #F

KEYWORD 006 at #36747
EXECUTION start #369C7
Characterization Nib #D

KEYWORD 007 at #36756
EXECUTION start #36A5F
Characterization Nib #D

#366FF

END of MAIN TABLE

Keyword CMDST

Token= 5

Keyword GETNIB

Token= 3

Keyword KEYWAIT\$

Token= 2

Keyword RCOPY

Token= 1

Keyword REV\$

Token= 4

Keyword SETCMD

Token= 6

Keyword UNPRIV

Token= 7

#36767

END of TEXT TABLE

--MESSAGE TABLE--

#3679E

Low Message: #00 - 0
High Message: #03 - 3
xxxxxxxxxxxxxxxxxxxxxx
Block Length= 16 nibs
Message ID: #00 - 0

- - - - -

-Text Cell-

DISS

xxxxxxxxxxxxxxxxxxxxxx
Block Length= 8 nibs
Message ID: #01 - 1

- - - - -
-Mainframe Blds. Block-
#0B - 11

xxxxxxxxxxxxxxxxxxxxxx
Block Length= 8 nibs
Message ID: #02 - 2

- - - - -
-Mainframe Blds. Block-
#3A - 58

xxxxxxxxxxxxxxxxxxxxxx
Block Length= 37 nibs
Message ID: #03 - 3

- - - - -
-Long Text Cell-
File not in RAM
xxxxxxxxxxxxxxxxxxxxxx
#367E8

END of MESSAGE TABLE

DUMPLEX BASIC 453 06/12/85 09:52

```
10 INPUT "Filename: ";F$ @ A=HTD(ADDR$(F$)) @ T$=PEEK$(DTH$(A+16),2)
20 IF T$#"40" AND T$#"80" THEN DISP "Wrong Filetype" @ BEEP @ GOTO 10
30 N=HTD(REV$(PEEK$(DTH$(A+32),5)))+32 @ Y=MOD(N,16)
40 PRINT F$;TAB(11);ID#"&REV$(PEEK$(DTH$(A+37),2));TAB(18);IP((N-31)/2);"bytes"

50 PRINT @ PRINT @ PRINT "      0123456789ABCDEF ck" @ PRINT
60 FOR X=0 TO N DIV 16-1
70 A$=PEEK$(DTH$(A),16)
80 PRINT DTH$(X)[3]&" : "&A$&" "&FNS$(A$) @ A=A+16
90 NEXT X
100 IF Y=0 THEN 120
110 A$=PEEK$(DTH$(A),Y) @ PRINT DTH$(X)[3]&" : "&A$;TAB(23);FNS$(A$)
120 PRINT @ END
130 DEF FNS$(A$) @ S=0 @ FOR Z=1 TO LEN(A$) @ S=IP(NUM(A$[Z,Z])*Z+S) @ NEXT Z
140 FNS$=DTH$(MOD(S,256))[4] @ END DEF
```

MAKELEX BASIC 659 06/12/85 09:49

```
10 SFLAG -1 @ ON ERROR GOTO 30 @ DESTROY ALL
20 PURGE DUMMY
30 ON ERROR GOTO 220
40 INPUT '# of bytes: ';N
50 CREATE TEXT DUMMY,N
60 A=HTD(ADDR$('DUMMY')) @ A1=A @ P$="-----"
70 Q=1 @ X=0 @ INPUT '000: ',P$,A$ @ C$=A$ @ GOSUB 200
80 Q=2 @ X=1 @ GOSUB 190
90 A$=A$&C$ @ A=A+37 @ N=N*2+31 @ Q=3 @ SFLAG 5
100 FOR X=2 TO N DIV 16-1
110 GOSUB 190
120 IF FLAG(5) THEN C$=C$[6]
130 POKE DTH$(A),C$ @ A=A+16-5*FLAG(5,0) @ NEXT X @ Q=4
140 DISP DTH$(X)[3]; @ INPUT ': ',P$[1,MOD(N,16)];C$
150 GOSUB 200
160 L=LEN(C$) @ IF C$[L,L]='-' THEN C$=C$[1,L-1] @ GOTO 160
170 POKE DTH$(A),C$ @ POKE DTH$(A1),A$
180 OFF ERROR @ CFLAG -1 @ END
190 DISP DTH$(X)[3]; @ INPUT ': ',P$;C$
200 DISP DTH$(X)[3]; @ INPUT ' ck ','---';C1$
210 S=0 @ FOR Z=1 TO LEN(C$) @ S=IP(NUM(C$[Z,Z])*Z+S) @ NEXT Z
220 IF C1$=DTH$(MOD(S,256))[4] THEN RETURN
230 DISP 'Checksum Error' @ BEEP @ POP @ ON Q GOTO 70,80,110,140
240 DISP 'Error:'&ERRM$ @ BEEP @ GOTO 180
```

DISLEX

ID#35 562 bytes

0123456789ABCDEF ck

000: 449435C454850202 AE
001: 802E102301316058 5D
002: 4640053107000000 1D
003: FD4006E00CA000F2 64
004: 0F2100DC10C1200F 3A
005: D00A4200FC30E720 2E
006: 0F00058200F7406D 26
007: 200D65056300D934 79
008: D444354550B74544 66
009: 5E4942430FB45495 55
00A: 7514944542209253 87
00B: 4F40595107255465 F1
00C: 4240B35544534D44 BA
00D: 460B55E405259465 9D
00E: 701FF969505E211B AF
00F: 1351121CB1378B6A 39
010: 113510B3B24A3359 15
011: 4440215DB0000300 37
012: 10044494353502C8 7F
013: 010EB0C8020EA3C5 79
014: 230BE6496C65602E EF
015: 6F6470296E602251 28
016: 4D4CFF0F0002F000 DC
017: 218F13510455AF33 3F
018: 3020210814A313F9 FA
019: 6641161873D18F13 32
01A: 110564310F9E2A08 AD
01B: 734157E310D96681 06
01C: 307A87161AF05028 E9
01D: DA93908FD53A05C0 28
01E: 8FD2F9047E8138FA 22
01F: 5480873908536F7F 4A
020: 7A004AC8D84A808F 24
021: EC110860F0042033 89
022: 20535AA8F8448040 E9
023: 213710B1378FBFB6 81
024: 08DBE2808D104808 00
025: D704808D227508DC 2D
026: 0B30001361081371 1C
027: 098F2C6004511191 C5
028: 351181348D8ACA18 9F
029: F127006BDF8118FD 6F
02A: F8E08F322B145051 53
02B: 6137128131D015B0 60
02C: 8FB13B1AFE120133 1E
02D: 8D832F04118FE83B C3
02E: 18DC32F0001361B6 C7
02F: 79F2D015A0136E48 6E
030: FB13B1AF68D612F0 6D
031: 04203310538DA939 61
032: 08D394508FD96308 A4
033: D2713069DF7EFFF9 37

034: EFFF8F681F08FC1D 2D
035: B08F322B145059DA 1E
036: F28A81DCC30F8B67 47
037: C1F679F21590D6E6 83
038: C6D5C6C11F675F21 1D
039: 471F085F2C914517 28
03A: 41451741451F675F BD
03B: 2147135350003001 B2
03C: 5D5175CC56F8D84A 99
03D: 80BA000F90008FD5 1B
03E: 3A05928FD2F904F3 B2
03F: ACBB46461B4681E8 1B
040: 1E94A808C11EF8F3 5F
041: 6F904818F7F5A05A CA
042: 48FBFB60832908DA B6
043: 93908F350115E08F CD
044: 52A6090E908DAC5A 8D
045: 017314B30D0E0615 CD
046: 908D060A00420333 B8
047: 05359B8D28B308D5 50
048: E450 0C

main entry table

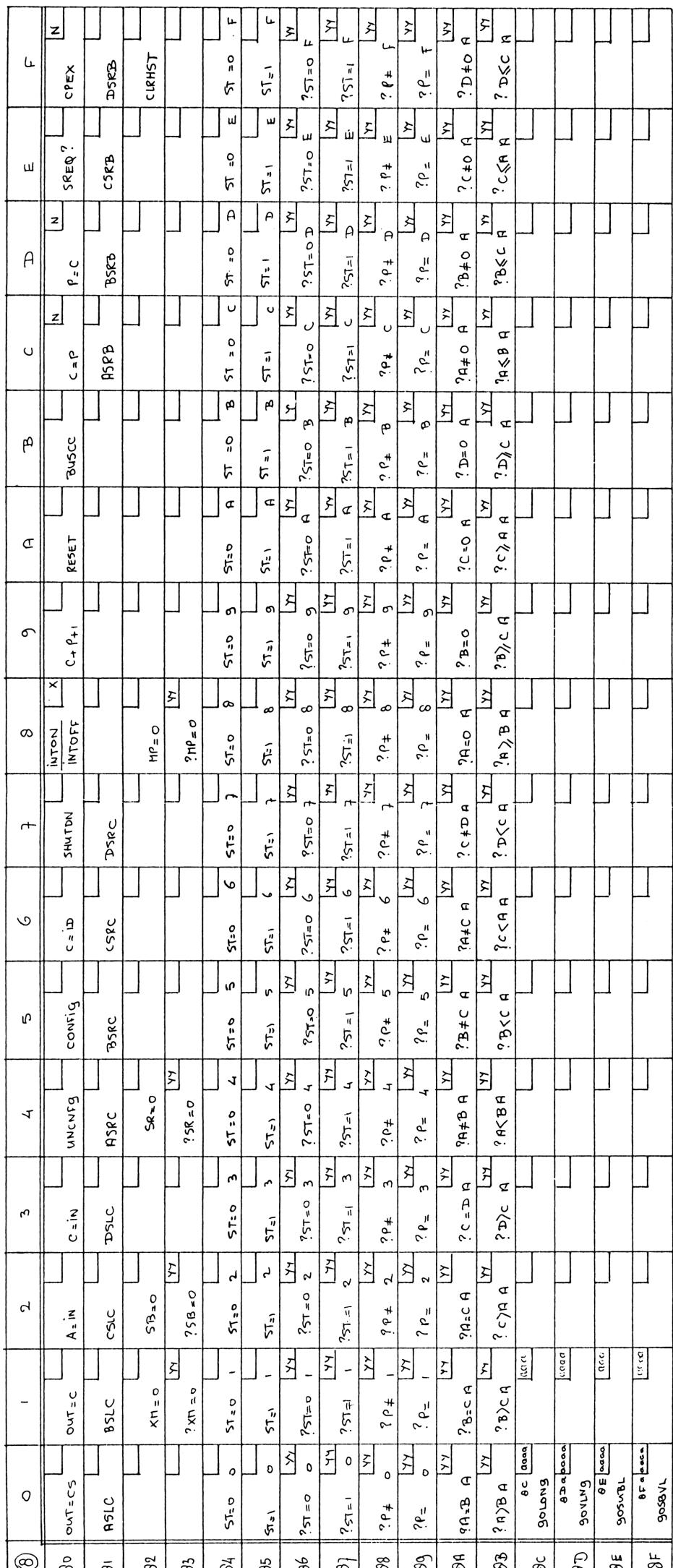
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
O	RTNSM RTNEY'S	RTN	RTNSC	RTNCC	SETHEX	SETDEC	RSTK=C	C=RSTK	CLEST	C=ST	ST=C	CSTEX	P=P+1	P=P-1	Q nib logicals	RTI
-1	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
2	P=Φ	P=1	P=2	P=3	P=4	P=5	P=6	P=7	P=8	P=9	P=A..0	P=B..1	P=C..2	P=D..3	P=E..4	P=F..5
3	LC 1	LC 2	LC 3	LC 4	LC 5	LC 6	LC 7	LC 8	LC 9	LC 10	LC 11	LC 12	LC 13	LC 14	LC 15	LC 16
4	LCHEX 1	LCHEX 2	LCHEX 3	LCHEX 4	LCHEX 5	LCHEX 6	LCHEX 7	LCHEX 8	LCHEX 9	LCHEX 10	LCHEX 11	LCHEX 12	LCHEX 13	LCHEX 14	LCHEX 15	LCHEX 16
5	RTN	RTNC	RTNCC	RTNCC	NOP3	NOP4	NOP5	NOP6	NOP7	NOP8	NOP9	NOP10	NOP11	NOP12	NOP13	NOP14
6	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
7	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
8	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
9	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
A	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
B	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
C	A=A+B A	B=B-C A	C=C+A A	D=D+C A	A=A+A A	B=B+A A	C=C+C A	D=D+D A	B=B+A A	C=C+D A	A=A-1 A	B=B-1 A	C=C-1 A	D=D-1 A	C=C-1 A	D=D-1 A
D	A=0 A	B=0 A	C=0 A	D=0 A	A=B A	B=C A	C=A A	D=C A	B=C A	C=D A	A=C A	B=C A	A=C A	B=C A	A=C A	B=C A
E	A=A-B A	B=B-C A	C=C+A A	D=D-C A	A=A+1 A	B=B+1 A	C=C+1 A	D=D+1 A	B=B-A A	C=C-B A	A=A-C A	B=B-A A	C=A-C A	D=C-D A	C=A-C A	D=C-D A
F	ASL A	BSL A	CSL A	DSL A	DSR A	CSR A	BSR A	DSR A	CSR A	BSR A	A=-A A	D=-D A	C=-C A	D=-D-1 A	C=-C-1 A	D=-D-1 A

γ	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F		
0	$A = A \& B$	P	$B = B \& C$	P	$C = C \& A$	P	$D = D \& C$	P	$A = A \& B$	P	$B = B \& A$	P	$C = C \& B$	P	$A = A \& C$	P	$C = C \& D$	P
1	$A = A \& B$	$W\Gamma$	$B = B \& C$	$W\Gamma$	$C = C \& A$	$W\Gamma$	$D = D \& C$	$W\Gamma$	$A = A \& C$	$W\Gamma$	$B = B \& A$	$W\Gamma$	$C = C \& B$	$W\Gamma$	$A = A \& C$	$W\Gamma$	$C = C \& D$	$W\Gamma$
2	$A = A \& B$	$X\Gamma$	$B = B \& C$	$X\Gamma$	$C = C \& A$	$X\Gamma$	$D = D \& C$	$X\Gamma$	$A = A \& C$	$X\Gamma$	$B = B \& A$	$X\Gamma$	$C = C \& B$	$X\Gamma$	$A = A \& C$	$X\Gamma$	$C = C \& D$	$X\Gamma$
3	$A = A \& B$	X	$B = B \& C$	X	$C = C \& A$	X	$D = D \& C$	X	$B = B \& A$	X	$C = C \& B$	X	$A = A \& C$	X	$B = B \& A$	X	$C = C \& D$	X
4	$A = A \& B$	S	$B = B \& C$	S	$C = C \& A$	S	$D = D \& C$	S	$B = B \& S$	S	$C = C \& B$	S	$A = A \& C$	S	$B = B \& A$	S	$A = A \& C$	S
5	$A = A \& B$	H	$B = B \& C$	H	$C = C \& A$	H	$D = D \& C$	H	$B = B \& H$	H	$C = C \& B$	H	$A = A \& C$	H	$B = B \& H$	H	$C = C \& D$	H
6	$A = A \& B$	B	$B = B \& C$	B	$C = C \& A$	B	$D = D \& C$	B	$B = B \& B$	B	$C = C \& B$	B	$A = A \& C$	B	$B = B \& B$	B	$C = C \& D$	B
7	$A = A \& B$	W	$B = B \& C$	W	$C = C \& A$	W	$D = D \& C$	W	$B = B \& W$	W	$C = C \& B$	W	$A = A \& C$	W	$B = B \& W$	W	$C = C \& D$	W
8																		
9																		
A																		
B																		
C																		
D																		
E																		
F	$A = A \& B$	A	$B = B \& C$	A	$C = C \& A$	A	$D = D \& C$	A	$B = B \& A$	A	$C = C \& B$	A	$D = D \& C$	A	$B = B \& A$	A	$C = C \& B$	A

①	O	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
10	RΦ = A	R1 = A	R2 = A	R3 = A	R4 = A						RΦ = C	R1 = C	R2 = C	R3 = C	R4 = C	
11	A = RΦ	A = R1	A = R2	A = R3	A = R4						C = RΦ	C = R1	C = R2	C = R3	C = R4	
12	ARΦEX	ARIEX	AR2EX	AR3EX	AR4EX						CRΦEX	CR1EX	CR2EX	CR3EX	CR4EX	
13	DΦ = A	D1 = A	DΦEX	D1EX	DΦ = C	D1 = C	CΙΦEX	CΙ1EX	DΦ = RS	D1 = RS	ADΦXS	AD1XS	DΦ = CS	D1 = CS	CΙDΦXS	
14	DATΦ = A	A = DAT1	DATΦ = A	DATΦ = A	DATΦ = A	DAT1 = C	A = DAT1	A = DAT1	A = DAT1	A = DAT1	DATΦ = R	DAT1 = R	A = DATΦ	B	DATΦ < C	C = DATΦ
15	DATΦ = A -	A = DAT1 -	DATΦ = A -	DATΦ = A -	DATΦ = A -	DAT1 = C -	DATΦ = C -	DAT1 = C -	DATΦ = C -	DAT1 = C -	DATΦ = R	DAT1 = R	A = DATΦ -	A = DAT1 -	DATΦ = C -	C = DATΦ -
16	DΦ = DΦ + 1	DΦ = DΦ + 2	DΦ = DΦ + 3	DΦ = DΦ + 4	DΦ = DΦ + 5	DΦ = DΦ + 6	DΦ = DΦ + 7	DΦ = DΦ + 8	DΦ = DΦ + 9	DΦ = DΦ + 10	DΦ = DΦ + 11	DΦ = DΦ + 12	DΦ = DΦ + 13	DΦ = DΦ + 14	DΦ = DΦ + 15	DΦ = DΦ + 16
17	D1 = D1 + 1	D1 = D1 + 2	D1 = D1 + 3	D1 = D1 + 4	D1 = D1 + 5	D1 = D1 + 6	D1 = D1 + 7	D1 = D1 + 8	D1 = D1 + 9	D1 = D1 + 10	D1 = D1 + 11	D1 = D1 + 12	D1 = D1 + 13	D1 = D1 + 14	D1 = D1 + 15	D1 = D1 + 16
18	DΦ = DΦ - 1	DΦ = DΦ - 2	DΦ = DΦ - 3	DΦ = DΦ - 4	DΦ = DΦ - 5	DΦ = DΦ - 6	DΦ = DΦ - 7	DΦ = DΦ - 8	DΦ = DΦ - 9	DΦ = DΦ - 10	DΦ = DΦ - 11	DΦ = DΦ - 12	DΦ = DΦ - 13	DΦ = DΦ - 14	DΦ = DΦ - 15	DΦ = DΦ - 16
19																
1A																
1B																
1C	D1 = D1 - 1	D1 = D1 - 2	D1 = D1 - 3	D1 = D1 - 4	D1 = D1 - 5	D1 = D1 - 6	D1 = D1 - 7	D1 = D1 - 8	D1 = D1 - 9	D1 = D1 - 10	D1 = D1 - 11	D1 = D1 - 12	D1 = D1 - 13	D1 = D1 - 14	D1 = D1 - 15	D1 = D1 - 16
1D																
1E																
1F																

$$a = P_{wp}, x_s, x_s, n, n \\ x = d_{-1} \quad (f_1 = d)$$

$$1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7$$



GUARINYES.

INTON : $x = 0$ (888F)
INTOFF : $x = F$ (848F)

⑥	o	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F		
90	?A=B	p	?B=c	p	?C=D	p	?A!=C	p	?B!=C	p	?A!=0	p	?B!=0	p	?C!=0	p	?D!=0	p
91	?A=B	wp	?B=c	wp	?C=D	wp	?A!=C	wp	?B!=C	wp	?A!=0	wp	?B!=0	wp	?C!=0	wp	?D!=0	wp
92	?A=B	xs	?B=c	xs	?A:C	xs	?A!=C	xs	?B!=C	xs	?A!=0	xs	?B!=0	xs	?C!=0	xs	?D!=0	xs
93	?A=B	x	?B=c	x	?C=D	x	?A!=B	x	?B!=C	x	?A!=0	x	?B!=0	x	?C!=0	x	?D!=0	x
94	?A=B	c	?B=c	c	?C=D	s	?A!=B	s	?B!=C	s	?A!=0	s	?B!=0	s	?C!=0	s	?D!=0	s
95	?A=B	h	?B=c	h	?C=D	h	?A!=B	h	?B!=C	h	?A!=0	h	?B!=0	h	?C!=0	h	?D!=0	h
96	?A=B	b	?B=c	b	?C=D	b	?A!=B	b	?B!=C	b	?A!=0	b	?B!=0	b	?C!=0	b	?D!=0	b
97	?A=B	w	?B=c	w	?C=D	w	?A!=B	w	?B!=C	w	?A!=0	w	?B!=0	w	?C!=0	w	?D!=0	w
98	?A=B	s	?B=c	s	?C=D	s	?A!=B	s	?B!=C	s	?A!=0	s	?B!=0	s	?C!=0	s	?D!=0	s
99	?A=B	wp	?B=c	wp	?C=D	wp	?A!=B	wp	?B!=C	wp	?A!=0	wp	?B!=0	wp	?C!=0	wp	?D!=0	wp
9E	?A=B	g	?B=c	g	?C=D	g	?A!=B	g	?B!=C	g	?A!=0	g	?B!=0	g	?C!=0	g	?D!=0	g
9F	?A=B	w	?B=c	w	?C=D	w	?A!=B	w	?B!=C	w	?A!=0	w	?B!=0	w	?C!=0	w	?D!=0	w

\textcircled{A}	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F		
A0	$A = A \cdot B$	P	$B = B + C$	P	$D = D \cdot C$	P	$A = A + A$	P	$B = B + B$	P	$D = D + D$	P	$C = C \cdot C$	P	$A = A + C$	P	$B = B \cdot B$	P
A1	$A = A \cdot B$	WP	$B = B + C$	WP	$D = D \cdot C$	WP	$A = A + A$	WP	$B = B + B$	WP	$D = D \cdot D$	WP	$C = C \cdot C$	WP	$A = A + C$	WP	$B = B \cdot B$	WP
A2	$A = A \cdot B$	X_S	$B = B + C$	X_S	$D = D \cdot C$	X_S	$A = A + A$	X_S	$B = B + B$	X_S	$D = D \cdot D$	X_S	$B = B + A$	WP	$A = A + C$	WP	$B = B \cdot B$	WP
A3	$A = A \cdot B$	X	$B = B + C$	X	$D = D \cdot C$	X	$A = A + A$	X	$B = B + B$	X	$D = D \cdot D$	X	$B = B + A$	X	$A = A + C$	X	$B = B \cdot B$	X
A4	$A = A \cdot B$	S	$B = B + C$	S	$D = D \cdot C$	S	$A = A + A$	S	$B = B + B$	S	$D = D \cdot D$	S	$B = B + A$	X_S	$A = A + C$	X_S	$C = C \cdot B$	X_S
A5	$A = A \cdot B$	H	$B = B + C$	H	$D = D \cdot C$	H	$A = A + A$	H	$B = B + B$	H	$D = D \cdot D$	H	$B = B + A$	H	$A = A + C$	H	$B = B \cdot B$	H
A6	$A = A \cdot B$	B	$B = B + C$	B	$D = D \cdot C$	B	$A = A + A$	B	$B = B + B$	S	$D = D \cdot D$	S	$B = B + A$	S	$A = A + C$	S	$C = C \cdot B$	S
A7	$A = A + B$	W	$B = B + C$	W	$D = D \cdot C$	W	$A = A + A$	W	$B = B + B$	W	$D = D \cdot D$	W	$B = B + A$	H	$A = A + C$	H	$B = B \cdot B$	H
A8	$A = 0$	P	$B = 0$	P	$D = 0$	P	$A = B$	P	$B = C$	P	$D = C$	P	$B = C$	WP	$A = C$	WP	$B = B \cdot B$	P
A9	$A = 0$	WP	$B = 0$	WP	$D = 0$	WP	$A = B$	WP	$B = C$	WP	$D = C$	WP	$B = C$	WP	$A = C$	WP	$B = B \cdot B$	WP
AA	$A = 0$	X_S	$B = 0$	X_S	$D = 0$	X_S	$A = B$	X_S	$B = C$	X_S	$D = C$	X_S	$B = C$	XS	$A = C$	XS	$B = B \cdot B$	XS
AB	$A = 0$	X	$B = 0$	X	$D = 0$	X	$A = B$	X	$B = C$	X	$D = C$	X	$B = C$	X	$A = C$	X	$B = B \cdot B$	X
AC	$A = 0$	S	$B = 0$	S	$D = 0$	S	$A = B$	S	$B = C$	S	$D = C$	S	$B = C$	S	$A = C$	S	$B = B \cdot B$	S
AD	$A = 0$	H	$B = 0$	H	$D = 0$	H	$A = B$	H	$B = C$	H	$D = C$	H	$B = C$	H	$A = C$	H	$B = B \cdot B$	H
AE	$A = 0$	B	$B = 0$	B	$D = 0$	B	$A = B$	B	$B = C$	B	$D = C$	B	$A = C$	B	$A = C$	B	$B = B \cdot B$	B
AF	$A = 0$	W	$B = 0$	W	$D = 0$	W	$A = B$	W	$B = C$	W	$D = C$	W	$A = C$	W	$A = C$	W	$B = B \cdot B$	W

B	O	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F				
B0	A=A-B	P	B=B-C	P	C=C-A	P	A=A ₁	P	B=B ₁	P	D=D ₁	P	A=B-A	P	B=c-B	P	C=A-C	P	D=c-D	P
B1	A=A-B	WP	B=B-C	WP	C=C-A	WP	A=A ₁	WP	B=B ₁	WP	D=D ₁	WP	A=B-A	WP	B=c-B	WP	C=A-C	WP	D=c-D	WP
B2	A=A-B	X5	B=B-C	X5	C=C-A	X5	A=A ₁	X5	B=B ₁	X5	D=D ₁	X5	A=B-A	X5	B=c-B	X5	C=R-C	X5	D=c-D	X5
B3	A=A-B	X	B=B-C	X	C=C-A	X	A=A ₁	X	B=B ₁	X	D=D ₁	X	A=B-A	X	B=c-B	X	C=R-C	X	D=c-D	X
B4	A=A-B	S	B=B-C	S	C=C-A	S	A=A ₁	S	B=B ₁	S	D=D ₁	S	A=B-A	S	B=c-B	S	C=R-C	S	D=c-D	S
B5	A=A-B	M	B=B-C	M	C=C-A	M	A=A ₁	M	B=B ₁	M	D=D ₁	M	A=B-A	M	B=c-B	M	C=R-C	M	D=c-D	M
B6	A=A-B	B	B=B-C	B	C=C-A	B	A=A ₁	B	B=B ₁	B	D=D ₁	B	A=B-A	B	B=c-B	B	C=R-C	B	D=c-D	B
B7	A=A-B	W	B=B-C	W	C=C-A	W	A=A ₁	W	B=B ₁	W	D=D ₁	W	A=B-A	W	B=c-B	W	C=R-C	W	D=c-D	W
B8	ASL	P	B ₁ SL	P	C ₁ SL	P	DSL	P	BSR	P	DSR	P	A=-A	P	B=-B	P	C=-C	P	D=-D	P
B9	ASL	WP	B ₁ SL	WP	C ₁ SL	WP	DSL	WP	ASR	WP	BSR	WP	DSR	WP	A=-A	WP	B=-B	WP	C=-C	WP
B10	ASL	X5	B ₁ SL	X5	C ₁ SL	X5	DSL	X5	ASR	X5	DSR	X5	A=-A	X5	B=-B	X5	C=-C	X5	D=-D	X5
B11	ASL	X	B ₁ SL	X	C ₁ SL	X	DSL	X	ASR	X	DSR	X	A=-A	X	B=-B	X	C=-C	X	D=-D	X
B12	ASL	S	B ₁ SL	S	C ₁ SL	S	DSL	S	ASR	S	DSR	S	A=-A	S	B=-B	S	C=-C	S	D=-D	S
B13	ASL	H	B ₁ SL	H	C ₁ SL	H	DSL	H	ASR	H	DSR	H	A=-A	H	B=-B	H	C=-C	H	D=-D	H
B14	ASL	3	B ₁ SL	3	C ₁ SL	3	DSL	3	ASR	3	DSR	3	A=-A	3	B=-B	3	C=-C	3	D=-D	3
B15	ASL	W	B ₁ SL	W	C ₁ SL	W	DSL	W	ASR	W	DSR	W	A=-A	W	B=-B	W	C=-C	W	D=-D	W

