

- 1. IDENTIFICATION
- 1.1 Digital-8-20-F-Bin
- 1.2 Four-Word Floating-Point Package
- 1.3 June 26, 1965

2. ABSTRACT

This program is almost identical to the 3-word Floating-Point Package (Digital-8-5-S) except that accuracy is carried to 35 bits, and 4 12-bit words are used for storage.

3. REQUIREMENTS

3.1 Storage

This program occupies registers 7; 40-61; 5600-7577 (octal).

4. USAGE

4.1 Loading

Binary Loader (Digital-8-2-U) or DECtape System.

4.2 Calling Sequence

Identical to Digital-8-5-S.

5. RESTRICTIONS

See Digital-8-5-S.

6. DESCRIPTION

The floating accumulator resides in memory locations 44, 45, 46, and 47. The instructions FGET, FPUT use 4-word arguments (11-bit exponent + sign; 35-bit mantissa + sign). The 4-word package contains all operations except for square root (0002) and square (0001).

7. METHODS

See Digital-8-5-S.

8. FORMAT (Not Applicable)

9. EXECUTION TIME

9.3 Average

Execution times are very difficult to estimate as they greatly depend upon the data on which the floating-point package is operating. Generally speaking:

FADD	=	382 μ sec + 42(N) where N is the number of shifts to align binary points.
FSUB	=	FADD time + 42 μ sec
FDIV	=	3.4 msec (approximately)

FMPY = 3.3 msec (approximately)
 FGET = 156 μ sec
 FPUT = 172 μ sec
 FNOR = $168 + N(42)$ μ sec where N is number of shifts;
 +84 μ sec if argument <0.
 FEXT = 140.5 μ sec

10. PROGRAM

10.4 Program Listing

/4 WORD FLOATING POINT
 /ARITHMETIC INTERPRETER
 /PAGE 1

*40

0040	0000	EX1,	0
0041	0000	HIGH1,	0
0042	0000	MID1,	0
0043	0000	LOW1,	0
0044	0000	EXP,	0
0045	0000	HORDER,	0
0046	0000	MIDDL,	0
0047	0000	LORDER,	0
0050	0000	OVER2,	0

0051	0000	OVER1,	0
		*61	
0061	0000	FLAG,	0

/ARITHMETIC ERROR FLAG

*5600

5600	0000	FPNT,	0
5601	7300	CLA CLL	
5602	3051	DCA OVER1	
5603	3050	DCA OVER2	
5604	1600	TAD I FPNT	
5605	3257	DCA JUMP	
5606	1257	TAD JUMP	
5607	0265	AND PAGENO	
5610	7650	SNA CLA	
5611	5214	JMP .+3	
5612	1267	TAD MASK5	
5613	0200	AND FPNT	
5614	3262	DCA ADDRS	
5615	1270	TAD MASK7	
5616	0257	AND JUMP	
5617	1262	TAD ADDRS	
5620	3262	DCA ADDRS	

/GET INSTRUCTION

/PAGE 0??

/YES

/NO - GET PAGE BITS

/GET 7 BIT ADDRESS

5621	1266	TAD INDRCT	/BIT3=1??
5622	0257	AND JUMP	
5623	7650	SNA CLA	
5624	5227	JMP LOOP01	
5625	1662	TAD I ADDRS	/YES - DEFER
5626	3262	DCA ADDRS	
5627	2200	ISZ FPNT	
5630	1662	TAD I ADDRS	
5631	3040	DCA EX1	/EXPONENT
5632	1262	TAD ADDRS	
5633	3263	DCA SAVE	
5634	2263	ISZ SAVE	
5635	1663	TAD I SAVE	/HIGH ORDER
5636	3041	DCA HIGH1	
5637	2263	ISZ SAVE	
5640	1663	TAD I SAVE	
5641	3042	DCA MID1	/MIDDLE BITS
5642	2263	ISZ SAVE	
5643	1663	TAD I SAVE	
5644	3043	DCA LOW1	/LOWER BITS
5645	1257	TAD JUMP	
5646	7106	CLL RTL	
5647	7006	RTL	
5650	0264	AND MASK3	/LOOK-UP ON TABLE
5651	1271	TAD TABLE	
5652	3260	DCA JUMP2	
5653	1660	TAD I JUMP2	
5654	3260	DCA JUMP2	
5655	4660	JMS I JUMP2	/EXECUTE
5656	5201	JMP FPNT+1	/GET NEXT
5657	0000	JUMP,	0
5660	0000	JUMP2,	0
5661	0000	GO2,	0
5662	0000	ADDRS,	0
5663	0000	SAVE,	0
5664	0017	MASK3,	0017
5665	0200	PAGENO,	0200
5666	0400	INDRCT,	0400
5667	7600	MASK5,	7600
5670	0177	MASK7,	0177
5671	5672	TABLE,	+.1
5672	5714	EXIT	
5673	6000	FLAD	
5674	6026	FLSU	
5675	6367	FLMY	
5676	6600	FLDV	
5677	5702	FLGT	
5700	5733	FLPT	
5701	6200	FNORM	

		/FLOATING GET=5000	
5702	0000	FLGT,	0
5703	1040		TAD EX1
5704	3044		DCA EXP
5705	1041		TAD HIGH1
5706	3045		DCA HORDER
5707	1042		TAD MID1
5710	3046		DCA MIDDLE
5711	1043		TAD LOW1
5712	3047		DCA LORDER
5713	5201		JMP FPNT+1
		/FLOATING EXIT OR SUBROUTINE=00XX	
5714	0000	EXIT,	0
5715	1257		TAD JUMP
5716	0264		AND MASK3
5717	7450		SNA
5720	5600		JMP I FPNT
5721	1350		TAD TABLE6
5722	3260		DCA JUMP2
5723	1660		TAD I JUMP2
5724	3260		DCA JUMP2
5725	1200		TAD FPNT
5726	3261		DCA G02
5727	4660		JMS I JUMP2
5730	1261		TAD G02
5731	3200		DCA FPNT
5732	5201		JMP FPNT+1
		/FLOATING PUT=6000	
5733	0000	FLPT,	0
5734	1044		TAD EXP
5735	3662		DCA I ADDRS
5736	1045		TAD HORDER
5737	2262		ISZ ADDRS
5740	3662		DCA I ADDRS
5741	1046		TAD MIDDLE
5742	2262		ISZ ADDRS
5743	3662		DCA I ADDRS
5744	1047		TAD LORDER
5745	2262		ISZ ADDRS
5746	3662		DCA I ADDRS
5747	5201		JMP FPNT+1
5750	5750	TABLE6,	.
5751	5770		EXIT6
5752	5770		EXIT6
5753	5770		EXIT6
5754	5770		EXIT6
5755	5770		EXIT6
5756	5770		EXIT6

/BITS 8-11=0??
/YES:FEXT
/NO:LOOKUP BITS 8-11
/ON SUBROUTINE TABLE

/SAVE PSEUDO PC

/RESTORE PSEUDO PC

/RETURN

/SUBROUTINE TABLE
/ABSOLUTE ADDRESSES
/OF SUBROUTINES

/EXIT6=DUMMY OR NOP

5757	5770	EXIT6
5760	5770	EXIT6
5761	5770	EXIT6
5762	5770	EXIT6
5763	5770	EXIT6
5764	5770	EXIT6
5765	5770	EXIT6
5766	5770	EXIT6
5767	5770	EXIT6

5770	0000	EXIT6,	0
5771	5770	JMP I	EXIT6

/FLOATING ADD=1000
*6000

6000	0000	FLAD,	0	
6001	4231	JMS	ALIGN	/ALIGN WORDS
6002	5600	JMP I	FLAD	/NO ALIGNMENT
6003	4312	JMS	SCALE	
6004	7300	CLA	CLL	/TRIPLE ADDITION
6005	1051	TAD	OVER1	
6006	1050	TAD	OVER2	
6007	3050	DCA	OVER2	
6010	7004	RAL		/CARRY
6011	1043	TAD	LOW1	
6012	1047	TAD	LORDER	
6013	3047	DCA	LORDER	
6014	7004	RAL		
6015	1042	TAD	MID1	
6016	1046	TAD	MIDDL	
6017	3046	DCA	MIDDL	
6020	7004	RAL		
6021	1041	TAD	HIGH1	
6022	1045	TAD	HORDER	
6023	3045	DCA	HORDER	
6024	4705	JMS	I NORMAL	
6025	5600	JMP I	FLAD	

/FLOATING SUBTRACT=2000

6026	0000	FLSU,	0	
6027	4706	JMS	I OPINS	/NEGATE OPERAND
6030	5201	JMP	FLAD+1 /ADD	

/ALIGN BINARY POINTS

6031	0000	ALIGN,	0	
6032	1045	TAD	HORDER	
6033	7640	SZA	CLA	
6034	5240	JMP	.+4	

6035	1040	TAD EX1	/C(FAC)=0
6036	3044	DCA EXP	
6037	5272	JMP DONE	
6040	1041	TAD HIGH1	
6041	7650	SNA CLA	
6042	5631	JMP I ALIGN	/OPERAND=0
6043	1040	TAD EX1	
6044	7041	CMA IAC	
6045	1044	TAD EXP	
6046	7450	SNA	
6047	5272	JMP DONE	/EXPONENTS EQUAL - EXIT
6050	7500	SMA	
6051	7041	CMA IAC	
6052	3304	DCA AMOUNT	/NUMBER OF PLACES
6053	1304	TAD AMOUNT	
6054	1307	TAD TEST1	
6055	7710	SPA CLA	
6056	5274	JMP NOGO	/NO SHIFTING POSSIBLE
6057	1040	TAD EX1	
6060	7041	CMA IAC	
6061	1044	TAD EXP	
6062	7004	RAL	
6063	7620	SNL CLA	
6064	1310	TAD TCON1	/SHIFT OPERAND RIGHT
6065	1311	TAD TCON2	/SHIFT FAC RIGHT
6066	3303	DCA POINT	
6067	4703	JMS I POINT	
6070	2304	ISZ AMOUNT	
6071	5267	JMP .-2	
6072	2231	DONE, ISZ ALIGN	
6073	5631	JMP I ALIGN	
6074	1040	NOGO, TAD EX1	
6075	7041	CMA IAC	
6076	1044	TAD EXP	
6077	7700	SMA CLA	
6100	5631	JMP I ALIGN	
6101	5702	JMP I .+1	
6102	5703	FLGT+1	
6103	0000	POINT, 0	
6104	0000	AMOUNT, 0	
6105	6200	NORMAL, FNORM	
6106	6306	OPMINS, OPNEG	
6107	0045	TEST1, 0045	
6110	0023	TCON1, SHFTOP-SHFTAC	
6111	6116	TCON2, SHFTAC	

```

/SCALE BOTH RIGHT
6112 0000 SCALE, 0
6113 4341 JMS SHFTOP
6114 4316 JMS SHFTAC
6115 5712 JMP I SCALE

/SCALE FLOATING AC RIGHT
6116 0000 SHFTAC, 0
6117 7300 CLA CLL
6120 1045 TAD HORDER
6121 7510 SPA
6122 7020 CML
6123 7010 RAR
6124 3045 DCA HORDER
6125 1046 TAD MIDDLE
6126 7010 RAR
6127 3046 DCA MIDDLE
6130 1047 TAD LORDER
6131 7010 RAR
6132 3047 DCA LORDER
6133 1050 TAD OVER2
6134 7010 RAR
6135 3050 DCA OVER2
6136 2044 ISZ EXP
6137 7000 NOP
6140 5716 JMP I SHFTAC

```

```

/SCALE OPERAND RIGHT
6141 0000 SHFTOP, 0
6142 7300 CLA CLL
6143 1041 TAD HIGH1
6144 7510 SPA
6145 7020 CML
6146 7010 RAR
6147 3041 DCA HIGH1
6150 1042 TAD MID1
6151 7010 RAR
6152 3042 DCA MID1
6153 1043 TAD LOW1
6154 7010 RAR
6155 3043 DCA LOW1
6156 1051 TAD OVER1
6157 7010 RAR
6160 3051 DCA OVER1
6161 2040 ISZ EX1
6162 7000 NOP
6163 5741 JMP I SHFTOP

```


/NORMALIZE FLOATING ACCUMULATOR

*6200

6200	0000	FNORM,	0	
6201	7300		CLA CLL	
6202	3361		DCA MP1	/0 # OF SHIFTS
6203	3363		DCA MP3	/RESET SWITCH
6204	1045		TAD HORDER	
6205	7510		SPA	/INPUT<0
6206	2363		ISZ MP3	/YES-SET SWITCH
6207	7640		SZA CLA	/FAC=0?
6210	5224		JMP G06	/NO
6211	1046		TAD MIDDLE	
6212	7640		SZA CLA	
6213	5224		JMP G06	
6214	1047		TAD LORDER	
6215	7640		SZA CLA	
6216	5224		JMP G06	/NO
6217	1050		TAD OVER2	
6220	7640		SZA CLA	
6221	5224		JMP G06	/NO
6222	3044		DCA EXP	/YES
6223	5600		JMP I FNORM	/EXIT
6224	1363	G06,	TAD MP3	
6225	7640		SZA CLA	/WAS INPUT <0
6226	4261		JMS ACNEG	/YES
6227	1045	SHIFT,	TAD HORDER	
6230	7104		CLL RAL	
6231	7710		SPA CLA	/TOO FAR?
6232	5251		JMP NOREXT	/YES:EXIT ROUTINE
6233	1050		TAD OVER2	/NO
6234	7104		CLL RAL	
6235	3050		DCA OVER2	/SHIFT LEFT
6236	1047		TAD LORDER	
6237	7004		RAL	
6240	3047		DCA LORDER	
6241	1046		TAD MIDDLE	
6242	7004		RAL	
6243	3046		DCA MIDDLE	
6244	1045		TAD HORDER	
6245	7004		RAL	
6246	3045		DCA HORDER	
6247	2361		ISZ MP1	/ADD 1 TO COUNT
6250	5227		JMP SHIFT	/CONTINUE
6251	1361	NOREXT,	TAD MP1	/SUBTRACT COUNT FROM
6252	7041		CMA IAC	/EXPONENT
6253	1044		TAD EXP	
6254	3044		DCA EXP	
6255	1363		TAD MP3	/WAS INPUT<0??
6256	7640		SZA CLA	
6257	4261		JMS ACNEG	/YES
6260	5600		JMP I FNORM	/EXIT

```

/NEGATE FLOATING AC
6261 0000 ACNEG, 0
6262 7300 CLA CLL
6263 1050 TAD OVER2
6264 7041 CMA IAC
6265 3050 DCA OVER2
6266 1047 TAD LORDER
6267 7040 CMA
6270 7430 SZL
6271 7101 CLL IAC
6272 3047 DCA LORDER
6273 1046 TAD MIDDLE
6274 7040 CMA
6275 7430 SZL
6276 7101 CLL IAC
6277 3046 DCA MIDDLE
6300 1045 TAD HORDER
6301 7040 CMA
6302 7430 SZL
6303 7101 CLL IAC
6304 3045 DCA HORDER
6305 5661 JMP I ACNEG
/NEGATE OPERAND

```

```

6306 0000 OPNEG, 0
6307 7300 CLA CLL
6310 1051 TAD OVER1
6311 7041 CMA IAC
6312 3051 DCA OVER1
6313 1043 TAD LOW1
6314 7040 CMA
6315 7430 SZL
6316 7101 CLL IAC
6317 3043 DCA LOW1
6320 1042 TAD MID1
6321 7040 CMA
6322 7430 SZL
6323 7101 CLL IAC
6324 3042 DCA MID1
6325 1041 TAD HIGH1
6326 7040 CMA
6327 7430 SZL
6330 7101 CLL IAC
6331 3041 DCA HIGH1
6332 5706 JMP I OPNEG

```

```

6333 0000 MULTIP, 0
6334 3361 DCA MP1
6335 3364 DCA MPSCON
6336 1365 TAD THIR
6337 3363 DCA MP3
6340 7100 CLL
6341 1361 TAD MP1
6342 7010 RAR
6343 3361 DCA MP1
6344 1364 TAD MPSCON
6345 7420 SNL
6346 5351 JMP .+3
6347 7100 CLL
6350 1362 TAD MP2CON
6351 7010 RAR
6352 3364 DCA MPSCON
6353 2363 ISZ MP3
6354 5341 JMP MULTIP+6
6355 1361 TAD MP1
6356 7010 RAR
6357 7100 CLL
6360 5733 JMP I MULTIP
6361 0000 MP1, 0
6362 0000 MP2CON, 0
6363 0000 MP3, 0
6364 0000 MPSCON, 0
6365 7764 THIR, -14
6366 6400 FMULT1, FMULT
6367 0000 FLMY, 0
6370 4766 JMS I FMULT1
6371 4200 JMS FNORM
6372 3050 DCA OVER2
6373 2777 ISZ I SIGN1
6374 5767 JMP I FLMY
6375 4261 JMS ACNEG
6376 5767 JMP I FLMY
6377 6750 SIGN1, SGTST

```

*6400

/FLOATING MULTIPLY

/(A*2↑24+B*2↑12+C)*(D*2↑24+E*2↑12+F)

```

6400 0000 FMULT, 0
6401 7201 CLA IAC
6402 1040 TAD EX1
6403 1044 TAD EXP
6404 3044 DCA EXP
6405 1377 TAD SMACLA
6406 3772 DCA I SGNSW
6407 4773 JMS I SIGNP

```

/ADD EXPONENTS

/SET UP SIGN ROUTINE
/GO THERE

6410	1043	TAD LOW1	
6411	3775	DCA I MP2	
6412	1047	TAD LORDER	/C*F
6413	4774	JMS I DMULT	
6414	7200	CLA	
6415	1776	TAD I MP5	
6416	3371	DCA MUL5	
6417	1046	TAD MIDDLE	
6420	3775	DCA I MP2	
6421	1043	TAD LOW1	/B*F
6422	4774	JMS I DMULT	
6423	1371	TAD MUL5	
6424	3371	DCA MUL5	
6425	7004	RAL	
6426	1776	TAD I MP5	
6427	3370	DCA MUL4	
6430	7004	RAL	
6431	3367	DCA MUL3	
6432	1042	TAD MID1	
6433	3775	DCA I MP2	
6434	1047	TAD LORDER	/C*E
6435	4774	JMS I DMULT	
6436	1371	TAD MUL5	
6437	3371	DCA MUL5	
6440	7004	RAL	
6441	1370	TAD MUL4	
6442	1776	TAD I MP5	
6443	3370	DCA MUL4	
6444	7004	RAL	
6445	1367	TAD MUL3	
6446	3367	DCA MUL3	
6447	1045	TAD HORDER	
6450	3775	DCA I MP2	
6451	1043	TAD LOW1	/A*F
6452	4774	JMS I DMULT	
6453	1370	TAD MUL4	
6454	3370	DCA MUL4	
6455	7004	RAL	
6456	1367	TAD MUL3	
6457	1776	TAD I MP5	
6460	3367	DCA MUL3	
6461	7004	RAL	
6462	3366	DCA MUL2	
6463	1041	TAD HIGH1	
6464	3775	DCA I MP2	
6465	1047	TAD LORDER	/D*C
6466	4774	JMS I DMULT	
6467	1370	TAD MUL4	
6470	3370	DCA MUL4	
6471	7004	RAL	

6472	1367	TAD MUL3	
6473	1776	TAD I MP5	
6474	3367	DCA MUL3	
6475	7004	RAL	
6476	1366	TAD MUL2	
6477	3366	DCA MUL2	
6500	1046	TAD MIDDLE	
6501	3775	DCA I MP2	
6502	1042	TAD MIDI	/B*D
6503	4774	JMS I DMULT	
6504	1370	TAD MUL4	
6505	3370	DCA MUL4	
6506	7004	RAL	
6507	1367	TAD MUL3	
6510	1776	TAD I MP5	
6511	3367	DCA MUL3	
6512	7004	RAL	
6513	1366	TAD MUL2	
6514	3366	DCA MUL2	
6515	1045	TAD HORDER	
6516	3775	DCA I MP2	
6517	1042	TAD MIDI	/A*E
6520	4774	JMS I DMULT	
6521	1367	TAD MUL3	
6522	3367	DCA MUL3	
6523	7004	RAL	
6524	1366	TAD MUL2	
6525	1776	TAD I MP5	
6526	3366	DCA MUL2	
6527	7004	RAL	
6530	3365	DCA MUL1	
6531	1041	TAD HIGH1	
6532	3775	DCA I MP2	
6533	1046	TAD MIDDLE	/B*D
6534	4774	JMS I DMULT	
6535	1367	TAD MUL3	
6536	3367	DCA MUL3	
6537	7004	RAL	
6540	1366	TAD MUL2	
6541	1776	TAD I MP5	
6542	3366	DCA MUL2	
6543	7004	RAL	
6544	1365	TAD MUL1	
6545	3365	DCA MUL1	
6546	1045	TAD HORDER	
6547	3775	DCA I MP2	
6550	1041	TAD HIGH1	/A*D
6551	4774	JMS I DMULT	
6552	1366	TAD MUL2	

6553	3046		DCA MIDDLE
6554	7004		RAL
6555	1365		TAD MUL1
6556	1776		TAD I MP5
6557	3045		DCA HORDER
6560	1367		TAD MUL3
6561	3047		DCA LORDER
6562	1370		TAD MUL4
6563	3050		DCA OVER2
6564	5600		JMP I FMULT
6565	0000	MUL1,	0
6566	0000	MUL2,	0
6567	0000	MUL3,	0
6570	0000	MUL4,	0
6571	0000	MUL5,	0
6572	6740	SGNSW,	SGNSWT
6573	6727	SIGNP,	SIGNCL
6574	6333	DMULT,	MULTIP
6575	6362	MP2,	MP2CON
6576	6364	MP5,	MP5CON
6577	7700	SMACLA,	SMA CLA

/FLOATING DIVIDE=4000

*6600

6600	0000	FLDV,	0
6601	1040		TAD EX1
6602	7041		CMA IAC
6603	1044		TAD EXP
6604	7001		IAC
6605	3044		DCA EXP
6606	1326		TAD SPACLA
6607	3340		DCA SGNSWT
6610	4327		JMS SIGNCL
6611	1041		TAD HIGH1
6612	7650		SNA CLA
6613	5303		JMP DVER
6614	7300		CLA CLL
6615	3320		DCA QUOL
6616	3321		DCA QUOH
6617	1325		TAD MIF
6620	3324		DCA DIVCNT
6621	5233		JMP DVX
6622	1047	DV3,	TAD LORDER
6623	7004		RAL
6624	3047		DCA LORDER
6625	1046		TAD MIDDLE
6626	7004		RAL

/SUBTRACT EXPONENTS

/SET UP SIGNS

/DIVISOR=0?

/YES - ERROR

6627	3046		DCA MIDDLE	
6630	1045		TAD HORDER	
6631	7004		RAL	
6632	3045		DCA HORDER	
6633	1043	DVX,	TAD LOW1	/PARTIAL SUBTRACT
6634	1047		TAD LORDER	
6635	3322		DCA DTEM1	
6636	7004		RAL	
6637	1042		TAD MID1	
6640	1046		TAD MIDDLE	
6641	3323		DCA DTEM2	
6642	7004		RAL	
6643	1041		TAD HIGH1	
6644	1045		TAD HORDER	
6645	7420		SNL	/DIVISOR<DIVIDEND?
6646	5254		JMP DV2-1	/NO
6647	3045		DCA HORDER	/YES:C(L)=QUOTIENT BIT
6650	1323		TAD DTEM2	
6651	3046		DCA MIDDLE	
6652	1322		TAD DTEM1	
6653	3047		DCA LORDER	
6654	7200		CLA	
6655	1320	DV2,	TAD QUOL	/SHIFT BIT INTO
6656	7004		RAL	/QUOTIENT
6657	3320		DCA QUOL	
6660	1321		TAD QUOH	
6661	7004		RAL	
6662	3321		DCA QUOH	
6663	1050		TAD OVER2	
6664	7004		RAL	
6665	3050		DCA OVER2	
6666	2324		ISZ DIVCNT	/DONE?
6667	5222		JMP DV3	/NO
6670	1320		TAD QUOL	
6671	3047		DCA LORDER	
6672	1321		TAD QUOH	
6673	3046		DCA MIDDLE	
6674	1050		TAD OVER2	
6675	3045		DCA HORDER	
6676	3050		DCA OVER2	
6677	4717		JMS I NORMIT	
6700	2350	DEXIT,	ISZ SGNTST	
6701	4746		JMS I FACNEG	
6702	5600		JMP I FLDV	

6703	7240	DVER,	CLA CMA	/DIVIDE ERROR
6704	3047		DCA LORDER	
6705	7240		CLA CMA	
6706	3046		DCA MIDDLE	
6707	7040		CMA	
6710	7110		CLL RAR	
6711	3045		DCA HORDER	
6712	1045		TAD HORDER	
6713	3044		DCA EXP	
6714	2061		ISZ FLAG	
6715	7000		NOP	
6716	5300		JMP DEXIT	

6717	6200	NORMIT,	FNORM	
6720	0000	QUOL,	0	
6721	0000	QUOH,	0	
6722	0000	DTEM1,	0	
6723	0000	DTEM2,	0	
6724	0000	DIVCNT,	0	
6725	7735	MIF,	-43	/STEP COUNT
6726	7710	SPACLA,	SPA CLA	

/TEST SIGN SUBROUTINE

6727	0000	SIGNCL,	0	
6730	1351		TAD RESTOR	
6731	3350		DCA SGNTST	
6732	1045		TAD HORDER	
6733	7700		SMA CLA	
6734	5337		JMP .+3	
6735	4746		JMS I FACNEG	
6736	2350		ISZ SGNTST	
6737	1041		TAD HIGH1	
6740	7700	SGNSWT,	SMA CLA	/OR SPA CLA
6741	5727		JMP I SIGNCL	
6742	4747		JMS I OPNEGS	
6743	2350		ISZ SGNTST	
6744	7000		NOP	
6745	5727		JMP I SIGNCL	

6746	6261	FACNEG,	ACNEG
6747	6306	OPNEGS,	OPNEG
6750	0000	SGNTST,	0
6751	7776	RESTOR,	-2

ACNEG	6261	MPSCON	6364
ADDRS	5662	MP1	6361
ALIGN	6031	MP2	6575
AMOUNT	6104	MP2CON	6362
DEXIT	6700	MP3	6363
DIVCNT	6724	MP5	6576
DMULT	6574	MULTIP	6333
DONE	6072	MUL1	6565
DTEM1	6722	MUL2	6566
DTEM2	6723	MUL3	6567
DVER	6703	MUL4	6570
DVX	6633	MUL5	6571
DV2	6655	NOGO	6074
DV3	6622	NOREXT	6251
EXIT	5714	NORMAL	6105
EXIT6	5770	NORMIT	6717
EXP	0044	OPMINS	6106
EX1	0040	OPNEG	6306
FACNEG	6746	OPNEGS	6747
FLAD	6000	OVER1	0051
FLAG	0061	OVER2	0050
FLDV	6600	PAGENO	5665
FLGT	5702	POINT	6103
FLMY	6367	QUOH	6721
FLPT	5733	QUOL	6720
FLSU	6026	RESTOR	6751
FMULT	6400	SAVE	5663
FMULT1	6366	SCALE	6112
FNORM	6200	SGNSW	6572
FPNT	5600	SGNSWT	6740
GO2	5661	SGNTST	6750
GO6	6224	SHFTAC	6116
HIGH1	0041	SHFTOP	6141
HORDER	0045	SHIFT	6227
INDRCT	5666	SIGNCL	6727
JUMP	5657	SIGNP	6573
JUMP2	5660	SIGNI	6377
LOOP01	5627	SMACLA	6577
LORDER	0047	SPACLA	6726
LOW1	0043	TABLE	5671
MASK3	5664	TABLE6	5750
MASK5	5667	TCON1	6110
MASK7	5670	TCON2	6111
MIDDL	0046	TEST1	6107
MID1	0042	THIR	6365
MIF	6725		

/4/17/65-HB-DEC
/4 WORD
/FLOATING POINT I/O ROUTINES
/REQUIRES FLOATING POINT INTERPRETER
/ENTRY AT 0007

*7			
0007	5600	FPNT,	5600
*44			
0044	0000	EXPONT,	0
0045	0000	HORDER,	0
0046	0000	MIDDL,	0
0047	0000	LORDER,	0
*52			
0052	0000	FPAC1,	0
0053	0000		0
0054	0000		0
0055	0000		0
0056	7777	SWIT1,	7777
0057	7777	SWIT2,	7777
0060	0000	CHAR,	0
0061	0000	DSWIT,	0

/IF = 0, NO CR-LF AFTER OUTPUT
/IF = 0, NO LF AFTER CR IN INPUT
/CONTAINS LAST CHARACTER READ
/= 0 IF NO CONVERSION TOOK PLACE

*6767			
6767	0000	PRCHAR,	0
6770	1057		TAD SWIT2
6771	7650		SNA CLA
6772	5767		JMP I PRCHAR
6773	1377		TAD LFED
6774	4776		JMS I OPUT
6775	5767		JMP I PRCHAR
6776	7345	OPUT,	OUT
6777	0212	LFED,	0212

/DOUBLE PRECISION DECIMAL-BINARY
/INPUT AND CONVERSION
*7000

7000	0000	DECONV,	0	
7001	7200		CLA	/INITIALIZE MANTISSA
7002	3045		DCA HORDER	
7003	3046		DCA MIDDL	
7004	3047		DCA LORDER	
7005	3266		DCA SIGN	
7006	3267		DCA DNUMBR	
7007	4350		JMS INPUT	

7010	1340		TAD PLUS	/TEST FOR SIGN
7011	7450		SNA	
7012	5220		JMP DECON	
7013	1337		TAD MINUS	
7014	7440		SZA	
7015	5221		JMP .+4	
7016	7240		CLA CMA	
7017	3266		DCA SIGN	/IF-, SET SWITCH
7020	4350	DECON,	JMS INPUT	
7021	7200		CLA	
7022	1060		TAD CHAR	/IS IT A DIGIT
7023	1341		TAD MIN9	
7024	7500		SMA	
7025	5600		JMP I DECONV	/NO
7026	1342		TAD PLUS12	
7027	7510		SPA	
7030	5600		JMP I DECONV	/NO
7031	3265		DCA DIGIT	/YES
7032	1045		TAD HORDER	
7033	0343		AND MASK	/OVERFLOW?
7034	7440		SZA	
7035	5220		JMP DECON	/YES-IGNORE
7036	2061		ISZ DSWIT	
7037	2267		ISZ DNUMBR	/INDEX NUMBER OF DIGITS
7040	4242		JMS MULT10	
7041	5220		JMP DECON	/CONTINUE
7042	0000	MULT10,	0	/ROUTINE TO MULTIPLY
7043	1047		TAD LORDER	/DOUBLE PRECISION WORD
7044	3043		DCA 43	/BY TEN (DECIMAL)
7045	1046		TAD MIDL	
7046	3042		DCA 42	
7047	1045		TAD HORDER	/REMAIN=REMAINDER
7050	3041		DCA 41	
7051	3040		DCA 40	
7052	4270		JMS MULT2	/CALL SUBROUTINE TO
7053	4270		JMS MULT2	/MULTIPLY BY TWO
7054	4307		JMS DUBLAD	/CALL DOUBLE ADD
7055	4270		JMS MULT2	
7056	1265		TAD DIGIT	/ADD LAST DIGIT RECEIVED
7057	3043		DCA 43	
7060	3042		DCA 42	
7061	3041		DCA 41	
7062	4307		JMS DUBLAD	
7063	1040		TAD 40	/EXIT WITH REMAINDER
7064	5642		JMP I MULT10	/IN AC
7065	0000	DIGIT,	0	/STORAGE FOR DIGIT
7066	0000	SIGN,	0	/=0 IF PLUS: =7777 IF MINUS
7067	0000	DNUMBR,	0	/=NUMBER OF DIGITS
7070	0000	MULT2,	0	/MULTIPLY LORDER, HORDER BY 2

7071	7300		CLA CLL	
7072	1047		TAD LORDER	
7073	7004		RAL	
7074	3047		DCA LORDER	
7075	1046		TAD MIDDLE	
7076	7004		RAL	
7077	3046		DCA MIDDLE	
7100	1045		TAD HORDER	
7101	7004		RAL	
7102	3045		DCA HORDER	
7103	1040		TAD 40	
7104	7004		RAL	
7105	3040		DCA 40	
7106	5670		JMP I MULT2	
7107	0000	DUBLAD,	0	/DOUBLE PRECISION ADDITION
7110	7300		CLA CLL	
7111	1047		TAD LORDER	
7112	1043		TAD 43	
7113	3047		DCA LORDER	
7114	7004		RAL	
7115	1046		TAD MIDDLE	
7116	1042		TAD 42	
7117	3046		DCA MIDDLE	
7120	7004		RAL	
7121	1045		TAD HORDER	
7122	1041		TAD 41	
7123	3045		DCA HORDER	
7124	7004		RAL	
7125	1040		TAD 40	
7126	3040		DCA 40	
7127	5707		JMP I DUBLAD	
7130	0000	MSIGN,	0	/ROUTINE TO FORM
7131	7300		CLA CLL	/2'S COMPLEMENT
7132	2266		ISZ SIGN	/IF C(SIGN)=7777
7133	5730		JMP I MSIGN	
7134	4736		JMS I .+2	
7135	5730		JMP I MSIGN	
7136	6261		6261	/"ACNEG" IN INTERPRETER
7137	7776	MINUS,	253-255	/TEST FOR SIGN
7140	7525	PLUS,	-253	
7141	7506	MIN9,	-272	/TEST FOR DIGIT
7142	0012	PLUS12,	272-260	
7143	7600	MASK,	7600	/TEST FOR OVERFLOW
7144	7775	C.10,	7775	
7145	3146		3146	
			7146 3146	3146
7147	3147		3147	

/INPUT A CHARACTER, IF CR, TEST
 /INPUT SWITCH TO SEE IF LF SHOULD
 /BE TYPED. IF RUBOUT, RESTART INPUT

7150	0000	INPUT,	0	/INPUT A CHARACTER
7151	7200		CLA	
7152	6031		KSF	
7153	5352		JMP .-1	
7154	6036		KRB	
7155	3060		DCA CHAR	
7156	1060		TAD CHAR	
7157	4774		JMS I OUTPUT	
7160	1060		TAD CHAR	
7161	7450		SNA	
7162	5351		JMP INPUT+1	/IGNORE BLANKS
7163	1376		TAD MRBOUT	
7164	7450		SNA	
7165	5775		JMP I RESTRT	/RUBOUT-RESTART INPUT
7166	1377		TAD MINCR	
7167	7650		SNA CLA	
7170	4773		JMS I PRINT	/CR - SEE IF TO BE FOLLOWED
7171	1060		TAD CHAR	/BY LF
7172	5750		JMP I INPUT	/EXIT ROUTINE

7173	6767	PRINT,	PRCHAR
7174	7345	OUTPUT,	OUT
7175	7401	RESTRT,	FLINTP+1
7176	7401	MRBOUT,	-377
7177	0162	MINCR,	377-215

/FLOATING OUTPUT "E" FORMAT

/USES: TSF
 / JMP .-1
 / TLS

*7200

7200	0000	FLOUTP,	0	
7201	4217		JMS FOUTCN	/CONVERT MANTISSA AND OUTPUT
7202	1324		TAD BEXP	
7203	3044		DCA EXPONT	
7204	1343		TAD CHE	
7205	4345		JMS OUT	
7206	4737		JMS I FEXPPT	/CONVERT EXPONENT AND OUTPUT
7207	1056		TAD SWITI	/PRINT CR-LF?
7210	7650		SNA CLA	
7211	5600		JMP I FLOUTP	/NO-EXIT
7212	1341		TAD CARRTN	/YES
7213	4345		JMS OUT	
7214	1342		TAD LNFEED	
7215	4345		JMS OUT	
7216	5600		JMP I FLOUTP	/EXIT

/THIS WHOLE SUBROUTINE MAY BE ALTERED TO BUFFER
 /THE OUTPUT DIGITS : CHANGE JMS OUTDG TO DCA I 10, ETC.
 FOUTCN, 0

7217	0000		CLA CLL	
7220	7300		TAD HORDER	/NUMBER>0??
7221	1045		SPA CLA	
7222	7710		CLA CML	/NO SET LINK
7223	7220		TAD SPLUS	/YES
7224	1327		SZL	
7225	7430		TAD SMINUS	/NO
7226	1330		JMS OUT	
7227	4345		JMS OUTDG	/OUTPUT "0"
7230	4353		TAD PERIOD	
7231	1331		JMS OUT	/OUTPUT "."
7232	4345		CLA CLL	
7233	7300		TAD HORDER	
7234	1045		SMA CLA	
7235	7700		JMP FG01	
7236	5242		CMA	/NUMBER IS NEGATIVE
7237	7040		DCA I SNPT	/NEGATE
7240	3733		JMS I MSNPT	
7241	4732		CLA CMA	/SUBTRACT 1 FROM BINARY EXPON
7242	7240	FG01,	TAD EXPONT	/COMPENSATE AT FG04
7243	1044		DCA EXPONT	
7244	3044		DCA BEXP	/INITIALIZE DECIMAL EXPONENT
7245	3324		TAD EXPONT	/IS -4<EXPONENT<-1
7246	1044	FG02,	SMA	
7247	7500		JMP FG03	/TOO LARGE: MULTIPLY BY 1/10
7250	5263		TAD FOUR	
7251	1326		SMA CLA	
7252	7700		JMP FG04	
7253	5270		JMS I FPNT	/TOO SMALL-TIMES TEN
7254	4407		FMPY I TENPT	/TEN
7255	3740		FEXT	
7256	0000		CLA CMA	
7257	7240		TAD BEXP	
7260	1324		DCA BEXP	
7261	3324		JMP FG02	
7262	5246		JMS I FPNT	
7263	4407	FG03,	FMPY I PRC.10	/ONE TENTH
7264	3744		FEXT	
7265	0000		ISZ BEXP	
7266	2324		JMP FG02	
7267	5246			

7270	3734	FG04,	DCA I DPT	/MULTIPLY BY TWO
7271	4736		JMS I M2PT	/IE.SHIFT LEFT
7272	4735		JMS I M10PT	/MULTIPLY BY TEN
7273	7410		SKP	
7274	4360	FG05A,	JMS DIVTWO	/COMPENSATE FOR
7275	2044		ISZ EXPONT	/BINARY EXPONENT
7276	5274		JMP FG05A	
7277	7450		SNA	/IS FIRST DIGIT A ZERO
7300	5311		JMP FG07	/YES, IGNORE
7301	4353	FG06,	JMS OUTDG	/MULTIPLICATIONS YIELD
7302	1325		TAD MINUS7	/DECIMAL DIGITS AS HIGH
7303	3044		DCA EXPONT	/ORDER REMAINDERS
7304	4735	FG06A,	JMS I M10PT	/IE. .672X10=6+.72.. ETC
7305	4353		JMS OUTDG	
7306	2044		ISZ EXPONT	/7 DIGITS OUTPUT??
7307	5304		JMP FG06A	/NO: CONTINUE
7310	5617		JMP I FOUTCN	/YES:EXIT
7311	7240	FG07,	CLA CMA	/IGNORE FIRST DIGIT
7312	1324		TAD BEXP	/SUBTRACT 1 FROM
7313	3324		DCA BEXP	/DECIMAL EXPONENT
7314	1045		TAD HORDER	
7315	7640		SZA CLA	
7316	5322		JMP .+4	/IS MANTISSA ZERO?
7317	1047		TAD LORDER	
7320	7650		SNA CLA	
7321	3324		DCA BEXP	/YES:EXP=0
7322	7240		CLA CMA	
7323	5302		JMP FG06+1	
7324	0000	BEXP,	0	/CONTAINS DECIMAL EXPONENT
7325	7767	MINUS7,	-11	/NUMBER OF DIGITS OUTPUT
7326	0004	FOUR,	0004	
7327	0253	SPLUS,	253	
7330	0002	SMINUS,	255-253	
7331	0256	PERIOD,	256	
7332	7130	MSNPT,	MSIGN	
7333	7066	SNPT,	SIGN	/POINTERS
7334	7065	DPT,	DIGIT	
7335	7042	M10PT,	MULT10	
7336	7070	M2PT,	MULT2	
7337	7523	FEXPPT,	FEXC	
7340	7504	TENPT,	TEN	
7341	0215	CARRTN,	0215	
7342	0212	LNFEED,	0212	
7343	0305	CHE,	305	
7344	7144	PRC.10,	C.10	

7345	0000	OUT,	0	/OUTPUT ONE ASCII CHARACTER
7346	6041		TSF	
7347	5346		JMP .-1	
7350	6046		TLS	
7351	7200		CLA	
7352	5745		JMP I OUT	

7353	0000	OUTDG,	0	/OUTPUT ONE DIGIT
7354	1357		TAD C260	
7355	4345		JMS OUT	
7356	5753		JMP I OUTDG	

7357	0260	C260,	0260
------	------	-------	------

7360	0000	DIVTWO,	0	/DIVIDE BY TWO IE.
7361	7110		CLL RAR	/ROTATE RIGHT
7362	3345		DCA OUT	/TEMPORARY STORAGE
7363	1045		TAD HORDER	
7364	7010		RAR	
7365	3045		DCA HORDER	
7366	1046		TAD MIDDLE	
7367	7010		RAR	
7370	3046		DCA MIDDLE	
7371	1047		TAD LORDER	
7372	7010		RAR	
7373	3047		DCA LORDER	
7374	1345		TAD OUT	
7375	5760		JMP I DIVTWO	

/FLOATING POINT INPUT
*7400

7400	0000	FLINTP,	0	/INITIALIZE "PERIOD SWITCH"
7401	7240		CLA CMA	
7402	3314		DCA PRSW	
7403	3061		DCA DSWIT	
7404	4717		JMS I DPCVPT	/7777 = NO PERIOD
7405	7200		CLA	
7406	1060		TAD CHAR	
7407	1313		TAD PER	
7410	7640		SZA CLA	
7411	5220		JMP FIG01	/PERIOD FOUND
7412	1314		TAD PRSW	/SECOND PERIOD
7413	7650		SNA CLA	/YES, TERMINATE
7414	5222		JMP FIG02	/NO - SET NUMBER OF DIGITS TO
7415	3722		DCA I DPN	/SET PERIOD SWITCH TO 0
7416	3314		DCA PRSW	/CONVERT REST OF STRING
7417	5720		JMP I DPCSPT	

7420	1314	FIG01,	TAD PRSW	/PERIOD READ IN PREVIOUSLY?
7421	7650		SNA CLA	
7422	1722	FIG02,	TAD I DPN	/YES:-NUMBER OF DIGITS IN SER
7423	7041		CMA IAC	/NO
7424	3315		DCA SEXP	
7425	4721		JMS I MSGNPT	/TEST SIGN
7426	1312	FIG03,	TAD C43	
7427	3044		DCA EXPONT	
7430	4407		JMS I FPNT	/NORMALIZE F.P. NUMBER
7431	7000		FNOR	
7432	6052		FPUT FPACI	/SAVE NUMBER
7433	0000		FEXT	
7434	1060		TAD CHAR	
7435	1311		TAD MINUSE	
7436	7640		SZA CLA	/"E" READ IN?
7437	5252		JMP ENDFI	/NO
7440	4717		JMS I DPCVPT	/YES - CONVERT DECIMAL EXPONEN
7441	4721		JMS I MSGNPT	/TEST SIGN
7442	1045		TAD HORDER	/EXPONENT TOO LARGE??
7443	7510		SPA	
7444	7001		IAC	
7445	7640		SZA CLA	
7446	5277		JMP EXCESS	/YES
7447	1047		TAD LORDER	/NO:DECIMAL POINT IS
7450	1315		TAD SEXP	/C(SEXP)PLACES TO RIGHT
7451	3315		DCA SEXP	/OF LAST DIGIT

/END OF FLOATING POINT INPUT
/COMPENSATE FOR DECIMAL EXPONENTS

7452	4407	ENDFI,	JMS I FPNT	/RESTORE MANTISSA
7453	5052		FGET FPACI	
7454	0000		FEXT	
7455	1315		TAD SEXP	
7456	7450		SNA	
7457	5600		JMP I FLINTP	
7460	7700		SMA CLA	
7461	5270		JMP FIG04	
7462	4407		JMS I FPNT	/. IS TO THE LEFT:
7463	3710		FMPY I PC.10	/TIMES .1000
7464	0000		FEXT	
7465	2315		ISZ SEXP	
7466	5255		JMP ENDFI+3	
7467	5600		JMP I FLINTP	

7470	4407	FIG04,	JMS I FPNT	/. IS TO THE RIGHT: /MULTIPLY BY 10
7471	3304		FMPY TEN	
7472	0000		FEXT	
7473	7240		CLA CMA	
7474	1315		TAD SEXP	
7475	3315		DCA SEXP	
7476	5255		JMP ENDFI+3	
7477	1316	EXCESS,	TAD C3777	
7500	3044		DCA EXPONT	
7501	1316		TAD C3777	
7502	3045		DCA HORDER	
7503	5600		JMP I FLINTP	
7504	0004	TEN,	0004	
7505	2400		2400	
7506	0000		0000	
7507	0000		0000	
7510	7144	PC.10,	C.10	/ .10
7511	7473	MINUSE,	-305	
7512	0043	C43,	0043	
7513	7522	PER,	-256	
7514	0000	PRSW,	0	
7515	0000	SEXP,	0	/CONTAINS DECIMAL EXPONENT
7516	3777	C3777,	3777	
7517	7000	DPCVPT,	DECONV	
7520	7020	DPCSPT,	DECON	
7521	7130	MSGNPT,	MSIGN	
7522	7067	DPN,	DNUMBR	
/OUTPUT THE EXPONENT				
7523	0000	FEXC,	0	
7524	7300		CLA CLL	
7525	1044		TAD EXPONT	
7526	7510		SPA	
7527	7061		CMA IAC CML	
7530	3044		DCA EXPONT	
7531	1367		TAD C253	
7532	7430		SZL	
7533	1370		TAD C255	
7534	4775		JMS I DGPT	
7535	3045		DCA HORDER	
7536	1044		TAD EXPONT	
7537	2045		ISZ HORDER	
7540	1371		TAD M144	
7541	7500		SMA	
7542	5337		JMP .-3	

7543	1372	TAD C144
7544	3044	DCA EXPONT
7545	7040	CMA
7546	1045	TAD HORDER
7547	7440	SZA
7550	4775	JMS I DGPT
7551	3045	DCA HORDER
7552	1044	TAD EXPONT
7553	2045	ISZ HORDER
7554	1373	TAD M12
7555	7500	SMA
7556	5353	JMP .-3
7557	1374	TAD C12
7560	3047	DCA LORDER
7561	7240	CLA CMA
7562	1045	TAD HORDER
7563	4775	JMS I DGPT
7564	1047	TAD LORDER
7565	4775	JMS I DGPT
7566	5723	JMP I FEXC

7567	7773	C253,	0253-260
7570	0002	C255,	255-253
7571	7634	M144,	7634
7572	0144	C144,	0144
7573	7766	M12,	7766
7574	0012	C12,	0012
7575	7353	DGPT,	OUTDG

BEXP	7324
CARRTN	7341
CHAR	0060
CHE	7343
C.10	7144
C12	7574
C144	7572
C253	7567
C255	7570
C260	7357
C3777	7516
C43	7512
DECON	7020
DECONV	7000
DGPT	7575
DIGIT	7065
DIVTWO	7360
DNUMBR	7067
DPCSPT	7520

DPCVPT	7517	MINUSE	7511
DPN	7522	MINUS7	7325
DPT	7334	MIN9	7141
DSWIT	0061	MRBOUT	7176
DUBLAD	7107	MSGNPT	7521
ENDFI	7452	MSIGN	7130
EXCESS	7477	MSNPT	7332
EXPONT	0044	MULT10	7042
FEXC	7523	MULT2	7070
FEXPPT	7337	M10PT	7335
FG01	7242	M12	7573
FG02	7246	M144	7571
FG03	7263	M2PT	7336
FG04	7270	OPUT	6776
FG05A	7274	OUT	7345
FG06	7301	OUTDG	7353
FG06A	7304	OUTPUT	7174
FG07	7311	PC.10	7510
FIG01	7420	PER	7513
FIG02	7422	PERIOD	7331
FIG03	7426	PLUS	7140
FIG04	7470	PLUS12	7142
FLINTP	7400	PRCHAR	6767
FLOUTP	7200	PRC.10	7344
FOUR	7326	PRINT	7173
FOUTCN	7217	PRSW	7514
FPAC1	0052	RESTRT	7175
FPNT	0007	SEXP	7515
HORDER	0045	SIGN	7066
INPUT	7150	SMINUS	7330
LFED	6777	SNPT	7333
LNFEED	7342	SPLUS	7327
LORDER	0047	SWIT1	0056
MASK	7143	SWIT2	0057
MIDDL	0046	TEN	7504
MINCR	7177	TENPT	7340
MINUS	7137		

11. DIAGRAMS (Not Applicable)

12 REFERENCES

See Digital-8-5-S.