

```

J001      * BASIC-16 INITIALIZATION ROUTINE
0002      EXT SIT ADDRESS OF LAST AVAILABLE MEMORY LOCATION
0003      EXT .PTB ADDRESS OF LOWEST WORD IN TABLE
0004      EXT TYPE OUTPUTS MESSAGE ON ASR
0005      EXT IPUT INPUTS A LINE FROM THE ASR
0006      EXT GNBC GETS NEXT NON-BLANK CHARACTER FROM THE
0007      * INPUT BUFFER
0008      EXT LODF FLAGS PAPER TAPE INPUT FROM ASR
0009      EXT LFCR OUTPUTS CARRIAGE RETURN, LINE FEED ON ASR
0010      EXT JOB JOB COMMAND PROCESSOR
0011      EXT FINT INTEGER TO FLOATING POINT CONVERSION
0012      EXT DELT DAC TO ROUTINE TO FLAG ERROR DF
0013      EXT ATND DAC TO ARCIANGENT FUNCTION
0014      EXT TAND DAC TO TANGENT FUNCTION
0015      EXT SIND DAC TO SINE FUNCTION
0016      EXT COSD DAC TO COSINE FUNCTION
0017      EXT SQRD DAC TO SQUARE ROOT FUNCTION
0018      EXT SCVL STORES A NUMBER INTO THE FLOATING POINT
0019      * ACCUMULATOR
0020      EXT PCVL PRINTS THE NUMBER IN THE FLOATING POINT
0021      * ACCUMULATOR
0022      EXT C1
0023      EXT C10
0024      EXT C215
0025      EXT C240
0026      EXT C241
0027      EXT C260
0028      *
0029      ENT INIT
0030      EJCT
    
```



0031			REL		
0032		*			
0033		*			
0034	00000	0 000000	INITA DAC	*	ADDRESS OF FIRST WORD ABOVE MATH PACKAGE
0035	00001	0 000000	INIT DAC	**	ENTRY
0036	00002	0 02 00000	LDA INTA		INITIALIZE LOW POINTER TO FIRST WORD
0037	00003	0 04 00000	SIA PTB		ABOVE MATH PACKAGE
0038	00004	0 04 00000	SIA LDFB		MAKE SURE THERE IS NO TAPE INPUT
0039	00005	0 10 00000	JST LFCR		C/R, LINE FEED
0040	00006	0 10 00000	JST TYPE		OUTPUT ID MESSAGE
0041	00007	0 000000	XAC IDMS		
0042	00010	0 10 00000	JST LFCR		A COUPLE OF LINE FEEDS
0043	00011	0 10 00000	JST LFCR		
0044	00012	0 02 00174	LDA ATOD		LOAD ADDRESS OF ATAN MESSAGE
0045	00013	0 10 00143	JST DFUA		DELETE ARCTANGENT FUCTION
0046	00014	0 02 00000	LDA DELT		YES- REPLACE POINTER TO ATAN WITH A
0047	00015	0 13 00000	IMA ATND		POINTER TO ROUTINE TO FLAG ERROR DF
0048	00016	0 04 00000	SIA PTB		AND SET LOW POINTER TO FIRST WORD OF ATAN
0049	00017	0 02 00175	LDA SCOD		LOAD ADDRESS OF SIN, COS, TAN MESSAGE
0050	00020	0 10 00143	JST DFUA		DELETE SIN, COS, TAN FUNCTIONS
0051	00021	0 02 00000	LDA DELT		YES- REPLACE POINTER TO THOSE FUNCTIONS
0052	00022	0 04 00000	SIA TAND		WITH POINTER TO FLAG ERROR DF
0053	00023	0 04 00000	SIA SIND		X
0054	00024	0 13 00000	IMA COSD		X
0055	00025	0 04 00000	SIA PTB		SET LOW POINTER TO FIRST WORD OF COS
0056	00026	0 02 00176	LDA SQOD		LOAD ADDRESS OF SQUARE ROOT MESSAGE
0057	00027	0 10 00143	JST DFUA		DELETE SQUARE ROOT FUNCTION
0058	00030	0 02 00000	LDA DLLT		YES-REPLACE POINTER TO SQUARE ROOT FUNCTION
0059	00031	0 13 00000	IMA SGRO		WITH POINTER TO ROUTINE TO FLAG ERROR DF
0060	00032	0 04 00000	SIA PTB		AND SET LOW POINTER TO FIRST WORD OF SQR
0061			* NOW		SET HIGH POINTER, SIT
0062	00033	0 02 00171	IN01 LDA	C16K	LOAD HIGHEST POSSIBLE ADDRESS (16K)
0063	00034	0 04 00000	IN10 SIA	SIT	SURE ADDRESS IN HIGH POINTER
0064	00035	-0 13 00000	IMA*	SIT	CHECK IF ADDRESS EXISTS BY STORING,
0065	00036	-0 13 00000	IMA*	SIT	THEN LOADING (BUT DON'T DESTROY CONTENTS OF
0066	00037	0 05 00000	ERA	SIT	ADDRESS), AND COMPARING
0067	00040	101040	SNZ		TEST IF THE SAME
0068	00041	0 01 00044	JMP	*+3	YES-HIGH MEMORY ADDRESS IS IN SIT
0069	00042	0 07 00170	SUB	C4K	NO-SUBTRACT 4K
0070	00043	0 01 00034	JMP	IN10	AND TRY AGAIN
0071	00044	0 10 00000	JST	TYPE	ASK USER IF HE WOULD LIKE TO USE ALL OF
0072	00045	0 000000	XAC	HMAM	CODE
0073	00046	0 10 00000	IN03 JST	TYPE	REQUEST A YES OR HIGH OCTAL ADDRESS
0074	00047	0 000000	XAC	AYOH	X
0075	00050	0 10 00000	JST	LFCR	C/R, LINE FEED
0076	00051	0 02 00000	LDA	C241	OUTPUT EXCLAMATION POINT
0077	00052	0 10 00000	JST	IPUT	THEN INPUT LINE FROM ASK
0078	00053	0 000000	XAC	SBUF	INPUT BUFFER
0079	00054	0 10 00000	JST	GNBC	GET THE FIRST NON-BLANK CHARACTER
0080	00055	0 11 00172	CAS	C331	IS IT A 'Y'

0081	00036	0 01 00046	JMP	IN03	NOT Y OR NUMBER-REQUEST YES OR HIGH
0082			*		OCTAL ADDRESS
0083	00057	0 01 00114	JMP	IN04	YES-ASSUME YES ANSWER-JUMP TO CALCULATE
0084			*		USER SPACE
0085	00060	000201	IAB		SAVE THE CHARACTER
0086	00061	140040	CRA	CKA	
0087	00062	0 04 00167	STA	HOA	INITIALIZE HIGH OCTAL ADDRESS TO ZERO
0088	00063	000201	IAB		AND CLEAR B REGISTER
0089	00064	0 07 00000	IN05	SUB	C260
0090	00065	100400	SPL		SUBTRACT *260 TO GET PURE NUMBER
0091	00066	0 01 00046	JMP	IN03	MAKE SURE IT'S NOT LESS THAN ZERO
0092	00067	0 11 00000	CAS	C10	IF LESS, REQUEST YES OR HIGH ADDRESS
0093	00070	101000	NOP		OR GREATER THAN 7
0094	00071	0 01 00046	JMP	IN03	X
0095			*		IF GREATER, THEN REQUEST YES OR HIGH OCTAL
0096	00072	0 13 00167	IMA	HOA	ADDRESS
0097	00073	0412 75	LLR	3	ADD THE DIGIT TO THE HIGH ADDRESS
0098	00074	0 05 00167	ERA	HOA	BY SHIFTING ADDRESS THREE PLACES LEFT
0099	00075	000201	IAB		ADD THE DIGIT
0100	00076	100040	SZE		MAKE SURE THERE IS NO OVERFLOW BY TESTING
0101	00077	0 01 00046	JMP	IN03	THE B REGISTER
0102	00100	000201	IAB		OVERFLOW-REQUEST YES OR HIGH OCTAL ADDRESS
0103	00101	0 04 00167	STA	HOA	REPOSITION
0104	00102	0 10 00000	JST	GNBC	SAVE ADDRESS
0105	00103	0 11 00000	CAS	C215	GET NEXT NON-BLANK CHARACTER
0106	00104	0 01 00064	JMP	IN05	CHECK FOR C/R
0107	00105	100000	SKP		ADD DIGIT TO HIGH ADDRESS
0108	00106	000000	OCT	0	C/R-SKIP OUT OF LOOP
0109	00107	0 02 00167	LDA	HOA	SHOULDN'T GET HERE
0110	00110	0 11 00000	CAS	SIT	LOAD USER INPUT HIGH ADDRESS
0111	00111	0 01 00046	JMP	IN03	COMPARE WITH HIGH CORE ADDRESS
0112	00112	101000	NOP		TOO LARGE-TRY AGAIN
0113	00113	0 04 00000	SIA	SIT	X
0114			*		SAVE AS HIGH POINTER
0115	00114	0 02 00000	IN04	LDA	SIT
0116	00115	0 07 00000	SUB	PTB	LOAD HIGH POINTER
0117	00116	0 06 00000	ADD	C1	SUBTRACT LOW POINTER
0118	00117	0 11 00000	CAS	C10	ADD ONE
0119	00120	0 01 00126	JMP	**6	MAKE SURE IT ISN'T TOO SMALL
0120	00121	101000	NOP		OK-JUMP TO PRINT USER SPACE MESSAGE
0121	00122	0 10 00000	JST	TYPE	X
0122	00123	0 000000	XAC	ISSM	FLAG INSUFFICIENT USER SPACE
0123	00124	0 10 00000	JST	LFCR	X
0124	00125	0 01 00033	JMP	IN01	C/R, LINE FEED
0125	00126	0 10 00000	JST	FINT	LOOP TO SET HIGH POINTER
0126	00127	0 10 00000	JST	SCVL	FLOAT THE NUMBER
0127	00130	0 02 00000	LDA	C240	STORE IT INTO FLOATING POINT ACCUMULATOR
0128	00131	0 10 00000	JST	PCVL	SUPPRESS BLANKS
0129	00132	0 10 00000	JST	TYPE	PRINT THE NUMBER
0130	00133	0 000000	XAC	USPM	USER SPACE MESSAGE
					X

0131 00134 0 10 00000
 0132 00135 0 02 00001
 0133 00136 0 07 00000
 0134 00137 0 04 00001
 0135 00140 0 02 00061
 0136 00141 -0 04 00001
 0137 00142 0 01 00000
 0138

JST LFCR
 LDA INIT
 SUB CI
 SIA INIT
 LDA CRA
 SIA* INIT
 JMP JOB
 EJCT

C/R, LINE FEED
 CHANGE THE CALL TO THE INITIALIZATION
 ROUTINE TO A CRA

EXIT THROUGH JOB COMMAND PROCESSOR

0139	00143	0 000000	DFUA	DAC	**	ENTRY
0140	00144	0 04 00150		STA	DFID	STORE FUNCTION NAME MESSAGE
0141	00145	0 10 00000		JST	TYPE	OUTPUT QUESTION PREFIX
0142	00146	0 000000		XAC	DFD	X
0143	00147	0 10 00000		JST	TYPE	PRINT FUNCTION NAME
0144	00150	0 000000	DFID	DAC	**	X
0145	00151	0 10 00000		JST	LFCR	OUTPUT C/R, LINE FEED
0146	00152	0 02 00000	DFD2	LDA	C241	OUTPUT AN EXCAMATION POINT
0147	00153	0 10 00000		JST	IPUT	INPUT FROM THE ASK UNTIL A C/R
0148	00154	0 000000		XAC	SBUF	INPUT BUFFER
0149	00155	0 10 00000		JST	GNBC	GET THE FIRST NON-BLANK CHARACTER
0150	00156	0 11 00172		CAS	C331	IS IT A 'Y'
0151	00157	0 01 00164		JMP	DF01	NO-REQUEST A YES OR A NO
0152	00160	-0 01 00143		JMP*	DFUA	Y-ASSUME YES ANSWER-EXIT TO RESET LOW POINTER
0153			*			
0154	00161	0 11 00173		CAS	C316	IS IT A 'N'
0155	00162	100000		SKP		NO-REQUEST A YES OR A NO ANSWER
0156	00163	0 01 00033		JMP	IN01	N-ASSUME NO ANSWER-JUMP TO SET HIGH POINTER
0157	00164	0 10 00000	DF01	JST	TYPE	REQUEST A YES OR A NO ANSWER
0158	00165	0 000000		XAC	AYON	X
0159	00166	0 01 00152		JMP	DF02	LOOP TO INPUT ANSWER
0160				EJCT		

0161	00167	000000	HOA	BSZ	1
C162	00170	010000	C4K	OCT	10000
0163	00171	037777	C16KI	OCT	37777
0164	00172	000331	C331	OCT	331
0165	00173	000316	C316	OCT	316
0166	00174	0 000000	ATQD	XAC	ATNQ
0167	00175	0 000000	SCQD	XAC	SCTQ
0168	00176	0 000000	SQWD	XAC	SQRQ
0169					END

ATND	000000E	ATQD	000174	C1	000000E	C10	000000E
C16K	000171	C215	000000E	C240	000000E	C241	000000E
C260	000000E	C316	000173	C331	000172	C4K	000170
CO5D	000000E	CRA	000061	DELT	000000E	DF01	000164
DF02	000152	DFID	000150	DFUA	000143	FINI	000000E
GNBC	000000E	HOA	000167	IN01	000033	IN03	000046
IN04	000114	IN05	000064	IN10	000034	IN11	000001
INTA	000000	IPUT	000000E	JOB	000000E	LFCK	000000E
LD0F	000000E	PCVL	000000E	PTB	000000E	SCQD	000175
SCVL	000000E	SIND	000000E	SIT	000000E	SQWD	000176
SQRD	000000E	TAND	000000E	TYPE	000000E		

0000 WARNING ON ERROR FLAGS
DAP-16 MOD 2 REV. A 03-16-70

BASIC-16 INITIALIZATION ROUTINE

U		ATND	13	47					
		ATNQ	166						
	166	ATQD	44						
		AYOH	74						
		AYON	158						
U		CI	22	117	133				
U		C10	23	92	118				
	163	C16K	62						
U		C215	24	105					
U		C240	25	127					
U		C241	26	76	146				
U		C260	27	89					
	165	C316	154						
	164	C331	80	150					
	162	C4K	69						
U		COSD	16	54					
	86	CRA	135						
U		DELT	12	46	51	58			
	157	DF01	151						
	146	DF02	159						
	144	DFID	140						
		DFU	142						
	139	DFUA	45	50	57	152			
U		FINT	11	125					
U		GNBC	6	79	104	149			
		HMAN	72						
	161	HOA	87	96	98	103	109		
		IDMS	41						
	62	IN01	124	156					
	73	IN03	81	91	94	101	111		
	115	IN04	83						
	89	IN05	106						
	63	IN10	70						
	35	INIT	29	132	134	136			
	34	INTA	36						
U		IPUT	5	77	147				
		ISSM	122						
U		JOB	10	137					
U		LFCR	9	39	42	43	75	123	131 145
U		LODF	8	38					
U		PCVL	20	128					
U		PTB	3	37	48	55	60	116	
		SBUF	78	148					
	167	SCOD	49						
		SCTQ	167						
U		SCVL	18	126					
U		SIND	15	53					
U		SIT	2	63	64	65	66	110	113 15
	168	SQOD	56						
U		SQRD	17	59					

BASIC-16 INITIALIZATION ROUTINE

U	SORD	168									
U	TAND	14	22								
	TYPE	4	40	71	73	121	129	141	143		
		157									
	USPM	130									

34 SYMBOLS
125 REFERS
169 RECORDS
23 U FLAGS

016-XREF 24 OCT 69