

EINSTEIN REFERENCE CARD

(79-0871-7)

LIST OF RESERVED WORDS

NOTATION

The following notations have been used.

1. **J** indicates a numeric expression which must evaluate to a number in the range 0 to 255. If the result of the expression is not an integer, then it is rounded down to the nearest integer value.
2. **I** as above (J) but range is increased to 0 to 65535, or -32768 to +32767.
3. **N** a numeric expression which evaluates to a real number i.e. not necessarily an integer.
4. **V** a numeric variable name.
5. **SV** a string variable name.
6. **L** a line number.
7. **File** a valid file name including (optional) drive number, filename, and (optional) file descriptor.

ABS (N)

ADC (J)

AND see IF statement

APPEND <File>,SV

ASC (<string expression>)

ATN (N)

AUTO L1,L2

BCOL J

BEEP J

BIN\$ (I,J)

BTN (J)

CALL I

CALL (<numeric or string expression>)

CHAIN <File>

CHR\$ (J)

CLEAR I1,I2

CLOSE SV1,SV2,...,SVn

CLS

CLS 40

CLS 32

CONT

COS (N)

CREATE <File>,SV,I

DATA data1,data2,...,datan

DEEK (<address>)

DEF FN V1(V2)=<expression>

DEG (N)

DEL L1,L2

**DIM <array name>(I1 , I2 ,... , In),<array name>(...),
...,<array name>(...)**

DIR

DIR <filetypes>

DOKE <address>,I1,I2,...,In

DOS

DRAW x1,y1,z1 TO x2,y2,Z2 TO TO xn,yn,zn

DRIVE J

ELLIPSE x,y,R,T,z

ELSE (see IF statement)

END

EOF

ERA <File>

ERL

ERR (see ON ERR)

ERR\$

EVAL (<string expression>)

EXP (N)

FILL x,y,J

FMT J1,J2

FN (see DEF FN)

FOR V=N1 TO N2 STEP N3

GOOL J1,J2

GOSUB L

GOTO L

HEX\$ (I,J)

HOLD L1,L2

IF <condition> THEN <statement(s)>ELSE <statement(s)>

IF <condition> AND <condition> THEN <statement>

INCH

INCH\$

INCH\$ (J)

INP (J)

INPUT " prompt ";V1,V2,...,Vn

INPUT# J
 INT (N)
 IOM J1,J2
 KBD
 KBD\$
 KEY N, <string expression>
 LEFT\$ (<string expression>,J)
 LEN (<string expression>)
 LET V=<expression>
 LIST L1,L2,L3
 LISTP L1,L2,L3
 LN (N)
 LOAD <File>
 LOCK <File>
 LOG (N)
 MAG J
 MGE
 MID\$ (<string expression>,J1,J2)
 MOD
 MOS
 MUL\$ (<string expression>,J)
 MUSIC string expression 1, string expression 2,
 string expression 3
 NEW
 NEXT V1,V2,...,Vn
 NOT
 NULL J
 OFF
 ON J GOTO L1,L2,...,Ln
 ON J GOSUB L1,L2,...,Ln
 ON ERR <statement>
 OPEN <File>,SV,I
 OR
 ORIGIN x,y
 OUT J1,J2
 PEEK (I)
 PI
 PLOT x,y

POINT (x,y)
 POKE <address>,J1,J2,...,Jn
 POLY N,x,y,R,T,z
 POP
 POS (J)
 PRINT <expression>
 PRINT@ x,y,<expression>
 PRINT# J
 PSG <register number>,J
 PSW <password>
 PTR J,I
 RAD (N)
 READ V1,V2,...,Vn
 REM <statement>
 REN <old File> TO <new File>
 RENUM L1,L2
 RESTORE L
 RETURN
 RIGHT\$ (<string expression>,J)
 RND (I)
 RUN
 RUN L
 RUN <File>
 SAVE <File>
 SCRNS\$ (J)
 SEP J
 SGN (N)
 SHAPE N,<string expression>
 SIN (N)
 SIZE
 SPC (J)
 SPEED J
 SPRITE <sprite number>,x,y,<colour number>,N
 SPRITE OFF <sprite number>
 SQR (N)
 STEP N
 STOP

STR\$ (N)

SWAP V1,V2

TAB (J1,J2)

TAN (N)

TCOL N1,N2

TEMPO N

THEN <statement>

TI\$ = "HHMMSS"

TO N

UNLOCK <File>

UNPLOT x,y

VAL (<string expression>)

VDEEK (I)

VDOKE <address>,I1,I2,...,In

VERIFY <File>

VOICE <voice generator>,<noise period>,<envelope
shape>,<envelope period>

VPEEK (I)

VPOKE <address>,J1,J2,...,Jn

WAIT J1,J2,J3

WIDTH J

XOR

ZONE J1,J2

ARITHMETIC OPERATORS : +,-,*,/,↑,MOD,()

RELATIONAL OPERATORS : >,<=,<,>=,<>

LOGICAL OPERATORS : NOT, AND, OR, XOR

SUFFIX INDICATOR FOR STRING VARIABLE NAMES : \$

PREFIX INDICATOR FOR HEXADECIMAL NUMBERS : &

COLOUR CODES

VALUE	COLOUR	VALUE	COLOUR
0	Transparent	8	Medium Red
1	Black	9	Light Red
2	Medium Green	10	Dark Yellow
3	Light Green	11	Light Yellow
4	Dark Blue	12	Dark Green
5	Light Blue	13	Magenta
6	Dark Red	14	Grey
7	Cyan	15	White

MOS COMMANDS

A	xxxx yyyy	- Arithmetic
B	Xy wwzz	- Baud Rate
C	xxxx yyyy zzzz	- Copy
D	xxxx	- HEX to DECIMAL conversions
E	xxxx	- Execute
F	xxxx yyyy zz	- Fill
G	xxxx yyyy	- Go to
H	dddd	- DECIMAL to HEX conversion
M	xxxx	- Modify
R	xxxx yyyy sstt d	- Read
T	xxxx yyyy zz	- Tabulate
W	xxxx yyyy sstt d	- Write
X		- Cold Start vector into language
Y		- Warm Start vector into language
Zx		- Display registers according to x
Z0		- Display normal registers
Z1		- Display alternate registers
Z2		- Display special registers

DOS COMMANDS

DIR	<filetypes>	- Display directory of specified files
DISP	<File>	- Display text file on screen
DRIVE	<drive number>	- Changes default drive
ERA	<File>	- Erases files
GO		- Jump to location 0100H
LOAD	<File>	- Load files
LOCK	<File>	- Lock files to prevent writing
MOS		- Return to MOS
PSW	<eight character name>	- Password Protection
REN	<old File> TO <new File>	- Renames files
SAVE N	<File>	- Saves memory from 0100H to files
UNLOCK	<File>	- Unlocks files for writing

40 COLUMN DISPLAY - 24 rows, 40 characters per row

32 COLUMN DISPLAY - 24 rows, 32 characters per row

CHARACTER CELLS - 8x8 pixels in 32 column display
8x6 pixels in 40 column display

PIXEL GRID - 192 vertical x 256 horizontal in 32 column display
192 vertical x 240 horizontal in 40 column display

DECIMAL/BINARY/HEXADECIMAL EQUIVALENTS

DEC	BINARY	HEX	DEC	BINARY	HEX	DEC	BINARY	HEX	DEC	BINARY	HEX
0	00000000	0	64	01000000	40	128	10000000	80	192	11000000	C0
1	00000001	1	65	01000001	41	129	10000001	81	193	11000001	C1
2	00000010	2	66	01000010	42	130	10000010	82	194	11000010	C2
3	00000011	3	67	01000011	43	131	10000011	83	195	11000011	C3
4	00000100	4	68	01000100	44	132	10000100	84	196	11000100	C4
5	00000101	5	69	01000101	45	133	10000101	85	197	11000101	C5
6	00000110	6	70	01000110	46	134	10000110	86	198	11000110	C6
7	00000111	7	71	01000111	47	135	10000111	87	199	11000111	C7
8	00001000	8	72	01001000	48	136	10001000	88	200	11001000	C8
9	00001001	9	73	01001001	49	137	10001001	89	201	11001001	C9
10	00001010	A	74	01001010	4A	138	10001010	8A	202	11001010	CA
11	00001011	B	75	01001011	4B	139	10001011	8B	203	11001011	CB
12	00001100	C	76	01001100	4C	140	10001100	8C	204	11001100	CC
13	00001101	D	77	01001101	4D	141	10001101	8D	205	11001101	CD
14	00001110	E	78	01001110	4E	142	10001110	8E	206	11001110	CE
15	00001111	F	79	01001111	4F	143	10001111	8F	207	11001111	CF
16	00010000	10	80	01010000	50	144	10010000	90	208	11010000	D0
17	00010001	11	81	01010001	51	145	10010001	91	209	11010001	D1
18	00010010	12	82	01010010	52	146	10010010	92	210	11010010	D2
19	00010011	13	83	01010011	53	147	10010011	93	211	11010011	D3
20	00010100	14	84	01010100	54	148	10010100	94	212	11010100	D4
21	00010101	15	85	01010101	55	149	10010101	95	213	11010101	D5
22	00010110	16	86	01010110	56	150	10010110	96	214	11010110	D6
23	00010111	17	87	01010111	57	151	10010111	97	215	11010111	D7
24	00011000	18	88	01011000	58	152	10011000	98	216	11011000	D8
25	00011001	19	89	01011001	59	153	10011001	99	217	11011001	D9
26	00011010	1A	90	01011010	5A	154	10011010	9A	218	11011010	DA
27	00011011	1B	91	01011011	5B	155	10011011	9B	219	11011011	DB
28	00011100	1C	92	01011100	5C	156	10011100	9C	220	11011100	DC
29	00011101	1D	93	01011101	5D	157	10011101	9D	221	11011101	DD
30	00011110	1E	94	01011110	5E	158	10011110	9E	222	11011110	DE
31	00011111	1F	95	01011111	5F	159	10011111	9F	223	11011111	DF
32	00100000	20	96	01100000	60	160	10100000	A0	224	11100000	E0
33	00100001	21	97	01100001	61	161	10100001	A1	225	11100001	E1
34	00100010	22	98	01100010	62	162	10100010	A2	226	11100010	E2
35	00100011	23	99	01100011	63	163	10100011	A3	227	11100011	E3
36	00100100	24	100	01100100	64	164	10100100	A4	228	11100100	E4
37	00100101	25	101	01100101	65	165	10100101	A5	229	11100101	E5
38	00100110	26	102	01100110	66	166	10100110	A6	230	11100110	E6
39	00100111	27	103	01100111	67	167	10100111	A7	231	11100111	E7
40	00101000	28	104	01101000	68	168	10101000	A8	232	11101000	E8
41	00101001	29	105	01101001	69	169	10101001	A9	233	11101001	E9
42	00101010	2A	106	01101010	6A	170	10101010	AA	234	11101010	EA
43	00101011	2B	107	01101011	6B	171	10101011	AB	235	11101011	EB
44	00101100	2C	108	01101100	6C	172	10101100	AC	236	11101100	EC
45	00101101	2D	109	01101101	6D	173	10101101	AD	237	11101101	ED
46	00101110	2E	110	01101110	6E	174	10101110	AE	238	11101110	EE
47	00101111	2F	111	01101111	6F	175	10101111	AF	239	11101111	EF
48	00110000	30	112	01110000	70	176	10110000	B0	240	11110000	F0
49	00110001	31	113	01110001	71	177	10110001	B1	241	11110001	F1
50	00110010	32	114	01110010	72	178	10110010	B2	242	11110010	F2
51	00110011	33	115	01110011	73	179	10110011	B3	243	11110011	F3
52	00110100	34	116	01110100	74	180	10110100	B4	244	11110100	F4
53	00110101	35	117	01110101	75	181	10110101	B5	245	11110101	F5
54	00110110	36	118	01110110	76	182	10110110	B6	246	11110110	F6
55	00110111	37	119	01110111	77	183	10110111	B7	247	11110111	F7
56	00111000	38	120	01111000	78	184	10111000	B8	248	11111000	F8
57	00111001	39	121	01111001	79	185	10111001	B9	249	11111001	F9
58	00111010	3A	122	01111010	7A	186	10111010	BA	250	11111010	FA
59	00111011	3B	123	01111011	7B	187	10111011	BB	251	11111011	FB
60	00111100	3C	124	01111100	7C	188	10111100	BC	252	11111100	FC
61	00111101	3D	125	01111101	7D	189	10111101	BD	253	11111101	FD
62	00111110	3E	126	01111110	7E	190	10111110	BE	254	11111110	FE
63	00111111	3F	127	01111111	7F	191	10111111	BF	255	11111111	FF

SCREEN CONTROL CODES

CTRL - A Dump Screen to Printer
 CTRL - D Move Cursor Right
 CTRL - F Delete at Cursor
 CTRL - G "BEEP" sound tone
 CTRL - H Move Cursor Left
 CTRL - I Tab Cursor (in BASIC only)
 CTRL - J Move Cursor Down
 CTRL - K Move Cursor Up
 CTRL - L Clear Screen
 CTRL - M Carriage Return
 CTRL - N Clear to 40 column
 CTRL - O Clear to 32 column
 CTRL - Q Cursor ON
 CTRL - R Printer Echo
 CTRL - S Printer OFF
 CTRL - T Cursor OFF
 CTRL - U Clear to end of line
 CTRL - V Clear to end of screen
 CTRL - X Clear whole line
 CTRL - Y Delete to left of Cursor
 CTRL - Z Insert at Cursor
 CTRL - ↑ Cursor Home

ASCII CODE	CHAR.	HEX	ASCII CODE	CHAR.	HEX	ASCII CODE	CHAR.	HEX
32	SP	20	65	A	41	98	b	62
33	!	21	66	B	42	99	c	63
34	"	22	67	C	43	100	d	64
35	#	23	68	D	44	101	e	65
36	\$	24	69	E	45	102	f	66
37	%	25	70	F	46	103	g	67
38	&	26	71	G	47	104	h	68
39	'	27	72	H	48	105	i	69
40	(28	73	I	49	106	j	6A
41)	29	74	J	4A	107	k	6B
42	*	2A	75	K	4B	108	l	6C
43	+	2B	76	L	4C	109	m	6D
44	,	2C	77	M	4D	110	n	6E
45	-	2D	78	N	4E	111	o	6F
46	.	2E	79	O	4F	112	p	70
47	/	2F	80	P	50	113	q	71
48	0	30	81	Q	51	114	r	72
49	1	31	82	R	52	115	s	73
50	2	32	83	S	53	116	t	74
51	3	33	84	T	54	117	u	75
52	4	34	85	U	55	118	v	76
53	5	35	86	V	56	119	w	77
54	6	36	87	W	57	120	x	78
55	7	37	88	X	58	121	y	79
56	8	38	89	Y	59	122	z	7A
57	9	39	90	Z	5A	123	¼	7B
58	:	3A	91	←	5B	124	½	7C
59	;	3B	92	½	5C	125	¾	7D
60	<	3C	93	→	5D	126	÷	7E
61	=	3D	94	↑	5E			
62	>	3E	95	—	5F			
63	?	3F	96	£	60			
64	@	40	97	a	61			

ERROR MESSAGES

	CODE	HEX	DECIMAL	
Break	00	0		Interruption from keyboard!
Next	01	1		NEXT statement found without corresponding FOR
Syntax	02	2		Typing error in line
Return	03	3		RETURN or POP found without corresponding GOSUB
Data	04	4		No more DATA statements from READ
Qty	05	5		Number specified outside allowable range
Ovfl	06	6		Number too large
Mem full	07	7		No more memory left
Branch	08	8		Attempt to refer to non-existent line
Range	09	9		Outside dimensions specified for array
Dimension	0A	10		DIM encountered for already dimensioned array
Division	0B	11		Divide by Zero!
Stack Full	0C	12		No more stack for FOR, GOSUB or expressions
Type	0D	13		String given when number expected or vice versa
Cmd	0E	14		Auxiliary Reserved Word not defined in system
Str Ovfl	0F	15		String expression too long
Str Complex	10	16		String expression too complex -- split it up!
Cont	11	17		Cannot continue after error or program mod.
Fn Defn	12	18		FN user function not defined by a previous DEF
Operand	13	19		Operand expected in expression
Bad Data	14	20		Disc checksum error
End of Text	15	21		End of File encountered
File	16	22		FDESC not defined (or used by another file)
Drive Select	17	23		Drive selected not available in system
File Type	18	24		File of incorrect type
No File	19	25		File not found
File Exists	1A	26		File already present (in RENAME)
File Locked	1B	27		File has been locked ('Read Only')
Disc Locked	1C	28		Disc is in 'Read Only' mode
Disc Seek	1D	29		Attempt to seek beyond end of disc
Disc Full	1E	30		No space for file contents
Dir Full	1F	31		Too many files in Directory