HP38G Entry Reference

Complete listing sorted by functionality
Edition 2.11, 30 May 2005

Carsten Dominik, Thomas Rast & Eduardo M. Kalinowski

Table of Contents

1 Introduction

This is a list of SystemRPL, User RPL and ML entries. The list groups the entries by task in many different chapters and sections. If you are looking for a particular entry go directly to the Index. There is also an address-sorted list, if you want to look up a particular address.

1.1 Disclaimer and Acknowledgments

The information provided in this document was compiled from a large variety of sources. The transformation of all the different formats to a single database was largely done with special purpose programs to reverse-engineer the different documents. This has worked very well in many cases, and less well in some other cases. If some of the information looks oddly formatted, the reason is probably the automatic extraction.

Many of the authors of the original documents will find literal bits and pieces of their text in this document. Thanks to all of them for their generosity in allowing me to use their documents and files freely.

Neither we nor the authors of the different sources assume any warranty. This document is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

If you find any errors, let us know so that the database can up updated and fixed. Sent bug reports and other comments to Carsten Dominik. Reports about the ML chapter should be sent directly to Thomas Rast, but with a CC to Carsten Dominik.

Here is a list of sources which have been used.

Programming in System RPL by Eduardo Kalinowski
: This book has been a major source for the database. The entire book has been reverse-engineered using pdftotext and then a variety of Emacs and Perl programs to extract and format the reference part of the book.

CAS Documentation Draft by Bernard Parisse
: Bernard Parisse has kindly sent me a file with draft documentation about most CAS entries which is the basis of the CAS chapter. This covers both code derived from Erable (written by Bernard) and from ALG48 (by Mika Heiskanen and Claude-Nicolas Fiechter). The documentation is not complete, and not entirely up-to-date. However, the information given should be accurate.

entries.srt by Mika Heiskanen
: Mika's really useful collection of entry description has been used to double-check the information derived from Eduardo's book.

ML entry descriptions by Peter Geelhoed
: Peter Geelhoed created the initial version of the ML section for this document.

HP48/49 entry cross-reference by Joe Horn
: This document has been used to make a list of entries for the HP49 in the first place, and to add and double-check addresses for both calculators.

Various posts on `comp.sys.hp48`
: A number of post on `comp.sys.hp48` have documented a set of entry points, for example the Graphical Toolbox (Cyrille de Brebisson), the Editor related entries (myself) and other stuff.

Supported entry lists from HP
: HP has published lists of supported entries for all calculators in the database. The lists generally only contain names and addresses, no further description.

Further contributions
: Denis Martinez, Alberto Zamora Oyarce, Wolfgang Rautenberg, Michael de Coninck, Christoph Giesselink, Martin Lang, Piotr Kowalewski, Lilian Pigallio and in particular Jean-Yves Avenard have also contributed information about various entry points and/or have replied to my questions about different aspects related to entries.

1.2 Terminology

1.2.1 Abbreviations used in Stack Diagrams

Here is a list of the codes use to denote different objects in the stack diagrams.

`ob`	any object
`1...n`	n objects
`#`	binary integer (BINT)
`HXS`	hex string (User binary integer)
`CHR`	character
`$`	character string
`T`	TRUE
`F`	FALSE
`flag`	TRUE or FALSE
`%`	real number
`%%`	extended real number
`%C`	complex number
`%%C`	extended complex number
`z, Z ,ZINT`	infinite precision integer
`N`	positive infinite precision integer
`s, symb`	symbolic
`u, unit`	unit object
`{}`	list
`A, []`	Array
`V, []`	Vector
`M, [[]]`	Matrix
`P`	Polynom, a list of Qs
`Q`	ZINT or P
`meta, ob1..obn #n`	meta object
`grob`	graphical object
`menu`	list or program returning a list

UserRPL stack diagrams use some additional abbreviations

`x,y`	real, list, generic UserRPL object
`c, (,)`	complex number
`#`	hex string (User binary integer)
`greek theta`	angle (a real number)
`m,n`	integer (ZINT or real)
`date`	DD.MMYYYY or MM.DDYYYY
`name`	global name
`prog,prg`	program
`f,func`	function
`F`	integral of f

1.2.2 Unsupported Entry Points

A large number of entries in this database are not officially supported (i.e. their address is not guarantied by HP to be stable). However, many of these entries can still be used, provided that the entry address is (or has been) *stable* in all ROM versions.

On the HP49G, two address intervals have been pointed out by Jean-Yves Avenard to be stable, so entries found in these intervals will be added to this database.

On the HP48G, no new ROM versions are to be expected, and all entries can be considered *stable*.

The names of unsupported but stable entries will be *enclosed in single parenthesis*, like `(CURSOR@)`.

1.2.3 More Information

This database has been used to create the entries reference in the second edtion of *Programming in System RPL* by Eduardo M. Kalinowski and C. Dominik. In this book, the entry list is embedded into a lot more information about SystemRPL and the HP49G, so if you need additional information, check the book. The main reasons to make also the entry database available is that it is a more compact listing, contains information about ML entries as well and lists the addresses of the entry on many different calculators.

2 HP Objects

2.1 Binary Integers

2.1.1 Built-in BINTS 0-127

0403F	BINT0	0d 0h aka: ZERO, any
04049	BINT1	1d 1h aka: ONE, real, MEMERR
04053	BINT2	2d 2h aka: TWO, cmp
0405D	BINT3	3d 3h aka: THREE, str
04067	BINT4	4d 4h aka: FOUR, arry
04071	BINT5	5d 5h aka: FIVE, list
0407B	BINT6	6d 6h aka: SIX, id, idnt
04085	BINT7	7d 7h aka: SEVEN, lam
0408F	BINT8	8d 8h aka: EIGHT, seco
04099	BINT9	9d 9h aka: NINE, symb
040A3	BINT10	10d Ah aka: TEN, sym
040AD	BINT11	11d Bh aka: ELEVEN, hxs
040B7	BINT12	12d Ch aka: TWELVE, grob
040C1	BINT13	13d Dh aka: TAGGED, THIRTEEN
040CB	BINT14	14d Eh aka: EXT, FOURTEEN, unitob
040D5	BINT15	15d Fh aka: FIFTEEN, rompointer

040DF	BINT16	16d 10h aka: REALOB, SIXTEEN
040E9	BINT17	17d 11h aka: SEVENTEEN, 2REAL, REALREAL
040F3	BINT18	18d 12h aka: EIGHTEEN
040FD	BINT19	19d 13h aka: NINETEEN
04107	BINT20	20d 14h aka: TWENTY
04111	BINT21	21d 15h aka: TWENTYONE
0411B	BINT22	22d 16h aka: TWENTYTWO
04125	BINT23	23d 17h aka: TWENTYTHREE
0412F	BINT24	24d 18h aka: TWENTYFOUR
04139	BINT25	25d 19h aka: TWENTYFIVE
04143	BINT26	26d 1Ah aka: REALSYM, TWENTYSIX
0414D	BINT27	27d 1Bh aka: TWENTYSEVEN
04157	BINT28	28d 1Ch aka: TWENTYEIGHT
04161	BINT29	29d 1Dh aka: TWENTYNINE
0416B	BINT30	30d 1Eh aka: REALEXT, THIRTY
04175	BINT31	31d 1Fh aka: THIRTYONE
0417F	BINT32	32d 20h aka: THIRTYTWO
04189	BINT33	33d 21h aka: THIRTYTHREE
04193	BINT34	34d 22h aka: THIRTYFOUR
0419D	BINT35	35d 23h aka: THIRTYFIVE

041B1	BINT37	37d 25h aka: THIRTYSEVEN
041BB	BINT38	38d 26h aka: THIRTYEIGHT
041C5	BINT39	39d 27h aka: THIRTYNINE
041CF	BINT40	40d 28h aka: FORTY, FOURTY
041D9	BINT41	41d 29h aka: FORTYONE
041E3	BINT42	42d 2Ah aka: FORTYTWO
041ED	BINT43	43d 2Bh aka: FORTYTHREE
80000	BINT44	44d 2Ch aka: FORTYFOUR
8000A	BINT45	45d 2Dh aka: FORTYFIVE
80014	BINT46	46d 2Eh aka: FORTYSIX
8001E	BINT47	47d 2Fh aka: FORTYSEVEN
80028	BINT48	48d 30h aka: FORTYEIGHT
80032	BINT49	49d 31h aka: FORTYNINE
8003C	BINT50	50d 32h aka: FIFTY
80046	BINT51	51d 33h aka: FIFTYONE
80050	BINT52	52d 34h aka: FIFTYTWO
8005A	BINT53	53d 35h aka: FIFTYTHREE, STRLIST, THREEFIVE
80064	BINT54	54d 36h aka: FIFTYFOUR
8006E	BINT55	55d 37h aka: FIFTYFIVE
8006E	#THREESEVEN	55d 37h

80078	BINT56	56d 38h aka: FIFTYSIX
80082	BINT57	57d 39h aka: FIFTYSEVEN
8008C	BINT58	58d 3Ah aka: FIFTYEIGHT
80096	BINT59	59d 3Bh aka: FIFTYNINE
800A0	BINT60	60d 3Ch aka: SIXTY
800AA	BINT61	61d 3Dh aka: SIXTYONE
800B4	BINT62	62d 3Eh aka: SIXTYTWO
800BE	BINT63	63d 3Fh aka: SIXTYTHREE
800BE	BINT3Fh	64d 3Fh
800C8	BINT64	64d 40h aka: BINT40h, SIXTYFOUR, YHI
800D2	BINT65	65d 41h aka: ARRYREAL
800D2	BINT_65d	65d 41h
800D2	SIXTYFIVE	65d 41h
800DC	BINT66	66d 42h aka: FOURTWO
800DC	SIXTYSIX	66d 42h
800E6	BINT67	67d 43h aka: FOURTHREE
800F0	BINT68	68d 44h aka: SIXTYEIGHT
800F0	2ARRY	68d 44h
800FA	BINT69	69d 45h aka: FOURFIVE
80104	BINT70	70d 46h aka: SEVENTY
80104	ARRYID	70d 46h
8010E	SEVENTYONE	71d 47h
80118	SEVENTYTWO	72h 48d
80122	SEVENTYTHREE	73d 49h

8012C	BINT74	74d 4Ah aka: SEVENTYFOUR
80136	SEVENTYFIVE	75d 4Bh
80140	SEVENTYSIX	76d 4Ch
8014A	SEVENTYSEVEN	77d 4Dh
80154	SEVENTYEIGHT	78d 4Eh
8015E	BINT79	79d 4Fh aka: SEVENTYNINE
80168	BINT80	80d 50h aka: EIGHTY
80168	LISTOB	80d 50h
80172	BINT81	81d 51h aka: EIGHTYONE, LISTREAL
8017C	BINT82	82d 52h aka: LISTCMP
80186	BINT83	83d 53h aka: FIVETHREE
80190	BINT84	84d 54h aka: FIVEFOUR
8019A	BINT85	85d 55h aka: 2LIST
801A4	BINT86	86d 56h aka: FIVESIX
801A4	LISTID	86d 56h
801AE	BINT87	87d 57h aka: LISTLAM
801B8	EIGHTYEIGHT	88d 58h
801C2	EIGHTYNINE	89d 59d
801CC	NINETY	90d 5Ah
801D6	BINT91	91d 5Bh aka: BINT_91d
801E0	NINETYTWO	92d 5Ch
801EA	NINETYTHREE	93d 5Dh
801F4	NINETYFOUR	94d 5Eh
801FE	NINETYFIVE	95d 5Fh
80208	BINT96	96d 60h aka: BINT_96d
80212	BINT97	97d 61h aka: IDREAL
8021C	IDCMP	98d 62h

8021C	NINETYEIGHT	98d 62h
80226	NINETYNINE	99d 63h
80230	BINT100	100d 64h aka: ONEHUNDRED
80230	IDARRY	100d 64h
8023A	IDLIST	101d 65h
8023A	SIXFIVE	101d 65h
80244	BINT111	111d 6Fh aka: char
80258	LAMREAL	113d 71h
80262	BINT_114	114d 72hd
8026C	BINT115	115d 73h aka: BINT_115d
80276	BINT116	116d 74h aka: BINT_116d
80280	LAMLIST	117d 75h
80280	BINT_117d	117d 75h
8028A	BINT122	122d 7Ah aka: BINT_122d

2.1.2 Built-in BINTS 127-255

80294	BINT128	128d 80h aka: BINT80h
80294	BINT_128d	128d 80h
8029E	BINT130	130d 82h aka: BINT130d, BINT_130d, XHI-1
802A8	BINT131	131d 83h aka: BINT_131d, BINT131d, XHI
802BC	SYMBREAL	145d 91h
802C6	SYMBCMP	146d 92h
802D0	SYMBSYM	154d 9Ah
802DA	SYMBUNIT	158d 9Eh
802E4	backup	159d 9Fh
802EE	SYMOB	160d A0h
802F8	SYMREAL	161d A1h
80302	SYMCMP	162d A2h
80316	SYMARRY	164d A4h
80320	SYMLIST	165d A5h
8032A	SYMID	166d A6h

80334	SYMLAM	167d A7h
8033E	SYMSYMB	169d A9h
80348	SYMSYM	170d AAh
80352	SYMEXT	174d AEh
8035C	HXSREAL	177d B1h
80366	2HXS	187d BBh
80370	BINTC0h	192d C0h
8037A	2GROB	204d CCh
80384	TAGGEDANY	208d D0h
8038E	EXTREAL	225d E1h
80398	EXTSYM	234d EAh
80398	UNITSYM	234d EAh
803A2	2EXT	238d EEh
803AC	ROMPANY	240d F0h
803B6	BINT253	253d FDh
803C0	BINT255d	255d FFh

2.1.3 Built-in BINTS 256-

803CA	REALOBOB	256d 100h
803D4	#_102	258d 102h
803D4	#_258_d	258d 102h
803DE	#SyntaxErr	262d 106h
803E8	BINT_263d	263d 107h
803F2	REALREALOB	272d 110h
803FC	3REAL	273d 111h
80406	Err#Kill	291d 123h
80406	#_291_d	291d 123h
80406	#_123	291d 123h
80410	#_124	292d 124h
80410	Err#NoLstStk	292d 124h
80410	#_292_d	292d 124h
8041A	#NoRoomForSt	305d 131h
8041A	BINT_305d	305d 131h
80424	BINT_306d	306d 132h
8042E	REALSTRSTR	307d 133h
8042E	BINT_307d	307d 133h
80492	Err#Cont	318d 13Eh

8049C	INTEGER337	337d 151h
804A6	CMPOBOB	512d 200h
804B0	Err#NoLstArg	517d 205h
804B0	#_205	517d 205h
804B0	#_517_d	517d 205h
804BA	STRREALREAL	785d 311h
804C4	ARRYREALREAL	1041d 411h
804CE	ARRYREALCMP	1042d 412h
804D8	3ARRY	1092d 444h
804E2	ARRYLISTREAL	1105d 451h
804EC	ARRYLISTCMP	1106d 452h
804F6	LISTREALOB	1296d 510h
80500	LISTREALREAL	1297d 511h
8050A	LISTLISTOB	1360d 550h
80514	IDREALOB	1552d 610h
8051E	IDLISTOB	1616d 650h
80528	FSTMACROROM#	1792d 700h
80528	LAMANYANY	1792d 700h
80532	PROGIDREAL	2145d 861h
8053C	PROGIDCMP	2146d 862h
80546	PROGIDLIST	2149d 865h
80550	PROGIDEXT	2158d 86Eh
8055A	ATTNERR	2563d A03h
80564	SYMREALREAL	2577d A11h
8056E	SYMREALCMP	2578d A12h
80578	SYMREALSYM	2586d A1Ah
80582	SYMCMPREAL	2593d A21h
8058C	SYMCMPCMP	2594d A22h
80596	SYMCMPSYM	2602d A2Ah
805A0	SYMIDREAL	2657d A61h
805AA	SYMIDCMP	2658d A62h
805B4	SYMIDLIST	2661d A65h
805BE	SYMIDEXT	2670d A6Eh
805C8	SYMSYMREAL	2721d AA1h
805D2	SYMSYMCMP	2722d AA2h
805DC	3SYM	2730d AAAh
805E6	XFERFAIL	3078d C06h
805F0	PROTERR	3079d C07h

805FA	InvalServCmd	3080d C08h
80604	Connecting	3082d C0Ah
8060E	Retry	3083d C0Bh
80618	#CAlarmErr	3583d DFFh
80622	EXTOBOB	3584d E00h
03FDB	TYPEREAL	10547d 2933h
0402B	TYPEEREL	10581d 2955h
03FF9	TYPEIDNT	10568d 2948h
03FE5	TYPECMP	10615d 2977h
03FEF	TYPELIST	10868d 2A74h
04017	TYPERRP	10902d 2A96h
0400D	TYPESYMB	10936d 2AB8h
04035	TYPEEXT	10970d 2ADAh
04003	TYPECOL	11677d 2D9Dh
03FF9	TYPEIDNT	10568d 2948h
04021	TYPELAM	11885d 2E6Dh
8062C	#EXITERR	458752d 70000h
8065E	MINUSFIVE	1048571d FFFFBh
80654	MINUSFOUR	1048572d FFFFCh
8064A	MINUSTHREE	1048573d FFFFDh
80640	MINUSTWO	1048574d FFFFEh
80636	MINUSONE	1048575d FFFFFh

2.1.4 Pushing Several BINTs

815D1	ZEROZERO	(→ #0 #0)
844CF	ONEONE	(→ #1 #1) aka: ONEDUP
8159A	DROPZERO	(ob → #0)
8159F	2DROP00	(ob ob → #0 #0)
8156D	DROPONE	(ob → #1)
84493	DUPZERO	(ob → ob ob #0)
844A7	DUPONE	(ob → ob ob #1)
844E3	DUPTWO	(ob → ob ob #2)
844BB	SWAPONE	(ob ob' → ob' ob #1)
838A9	ZEROSWAP	(ob → #0 ob)
83AE8	ZEROOVER	(ob → ob #0 ob)
83F8E	ZEROFALSE	(→ #0 F)

838D6	ONESWAP	(ob → #1 ob)
83FA2	ONEFALSE	(→ #1 F)

2.1.5 Conversion

81CB1	COERCE	(% → #)
83750	COERCEDUP	(% → # #)
838EA	COERCESWAP	(ob % → # ob)
458CA	%ABSCOERCE	(% → #)
81C02	CHR>#	(chr → #)

2.1.6 Arithmetic Functions

817D4	#+	(# #' → #+#')
817E8	#1+	(# → #+1)
817F7	#2+	(# → #+2)
815A4	#3+	(# → #+3)
815A9	#4+	(# → #+4)
815AE	#5+	(# → #+5)
DF0F3	#8+	(# → #+8)
817D9	#-	(# #' → #-#')
817ED	#1-	(# → #-1)
817FC	#2-	(# → #-2)
DF133	#3-	(# → #-3)
817CF	#*	(# #' → #*#')
813B5	#*OVF	(# #' → #*#') $0 \leq$ result $\leq$ FFFFF
817F2	#2*	(# → #*2)
817DE	#/	(# #' → #r #q)
81801	#2/	(# → #/2) Rounded down.
81595	#-#2/	(# #' → (#-#')/2)
8154F	#+DUP	(# #' → #+#' #+#')
8386D	#+SWAP	(ob # #' → #+#' ob)
83AC0	#+OVER	(ob # #' → ob #+#' ob)
83881	#-SWAP	(ob # #' → #-#' ob)
83AD4	#-OVER	(ob # #' → ob #-#' ob)
83895	#1+SWAP	(ob # → #+1 ob)
83A48	#1-ROT	(ob ob' # → ob' #-1 ob)

838BD	#1-1SWAP	(# → 1 #-1) Returns the bint ONE and the result.
81563	DUP#1+	(# → # #+1)
8415A	2DUP#+	(# #' → # #' #+#') aka: DUP3PICK#+
84236	DROP#1-	(# ob → #-1)
81540	SWAP#-	(# #' → #'-#)
81568	SWAP#1+	(meta ob → meta&ob) aka: SWP1+
82E95	'RSWP1+	(# → nob #+1) nob is the next object in the runstream.
84222	SWAP#1-	(# ob → ob #-1)
841FA	SWAPOVER#-	(# #' → #' #-#')
84182	OVER#+	(# #' → # #'+#)
841D2	OVER#-	(# #' → # #'-#)
84A42	OVER#1-	(# #' → # #' #'')
8416E	ROT#+	(# ob #' → ob #'+#)
841BE	ROT#-	(# ob #' → ob #'-#)
8420E	ROT#1+	(# ob ob' → ob ob' #+1)
81D2E	ROT#1+UNROT	(# ob ob' → #+1 ob ob')
8383B	ROT#+SWAP	(# ob #' → #'+# ob) aka: ROT+SWAP
84196	3PICK#+	(# ob #' → # ob #'+#)
841AA	4PICK#+	(# ob1 ob2 #' → # ob1 ob2 #'+#)
83854	4PICK#+SWAP	(# ob1 ob2 #' → # ob1 #'+# ob2) aka: 4PICK+SWAP
813A1	#MIN	(# #' → #'')
8139C	#MAX	(# #' → #'')
817E3	#AND	(# #' → #'') Bitwise AND.

2.1.7 Tests

81806	#=	(# #' → flag)
8180B	#<>	(# #' → flag)
81810	#<	(# #' → flag)
81900	#<=	(# #' → flag)
81815	#>	(# #' → flag)
818FB	#>=	(# #' → flag)
8181F	#0<>	(# → flag)
8181A	#0=	(# → flag)

813BA	#1<>	(# → flag)
813BF	#1=	(# → flag)
8411E	#2<>	(# → flag)
813C4	#2=	(# → flag)
81581	#3=	(# → flag)
8410A	#5=	(# → flag)
840C9	#<3	(# → flag)
84146	#>1	(# → flag) aka: ONE#>
81572	2DUP#<	(# #' → # #' flag)
8155E	2DUP#>	(# #' → # #' flag)
83DF4	ONE_EQ	(# → flag) Uses EQ test.
8151D	OVER#=	(# #' → # flag)
81577	2DUP#=	(# #' → # #' flag)
840A1	OVER#0=	(# #' → # #' flag)
8157C	DUP#0=	(# → # flag)
840B5	OVER#<	(# #' → # flag)
81586	DUP#1=	(# → # flag)
84132	OVER#>	(# #' → # flag)
8158B	DUP#0<>	(# → # flag)
840DD	DUP#<7	(# → # flag) Returns TRUE if the argument is smaller than #7.
84083	2#0=OR	(# # → flag) Returns TRUE if either argument is zero.

2.2 Real Numbers

2.2.1 Built-in Real Numbers

8083B	%-MAXREAL	-9.99E499
807E2	%-9	-9
807CD	%-8	-8
807B8	%-7	-7
807A3	%-6	-6
8078E	%-5	-5
80779	%-4	-4
80764	%-3	-3
8074F	%-2	-2

8073A	%-1	-1
80865	%-MINREAL	-1E-499
80668	%0	0
80850	%MINREAL	1E-499
80979	%.5	.5
8098E	%-.5	-.5
8067D	%1	1
80692	%2	2
80964	%e	e
806A7	%3	3
807F7	%PI	π
806BC	%4	4
806D1	%5	5
806E6	%6	6
806FB	%7	7
80710	%8	8
80725	%9	9
809A3	%10	10
48F1C	%15	15
48FB7	%25	25
809B8	%180	180
809CD	%200	200
809F7	%400	400
809E2	%360	360
80826	%MAXREAL	9.99E499

2.2.2 Built-in Extended Real Numbers

8087A	%%0	0
80916	%%.1	0.1
80930	%%.5	0.5
80894	%%1	1
808AE	%%2	2
808C8	%%3	3
8080C	%%PI	π
808E2	%%4	4
808FC	%%5	5
8094A	%%10	10

2.2.3 Conversion

81257	%>%%	(% → %%)
838FE	%>%%SWAP	(ob % → %% ob)
81252	%%>%	(%% → %)
81CB6	UNCOERCE	(# → %)
845A1	UNCOERCE%%	(# → %%)
81BFD	C%>%	(C% → %re %im)

2.2.4 Real Functions

8128E	%+	(% %' → %+%')
81293	%-	(% %' → %-%')
81289	%*	(% %' → %*%')
81298	%/	(% %' → %/%')
8129D	%^	(% %' → %^%')
812A2	%ABS	(% → %')
812D9	%CHS	(% → -%)
8133D	%SGN	(% → -1/0/1)
8134C	%SQRT	(% → $\sqrt{a}$%)
812ED	%EXP	(% → e^%)
812F2	%EXPM1	(% → e^%-1)
8130B	%LN	(% → LN%)
81310	%LNP1	(% → LN(%+1))
81315	%LOG	(% → LOG%)
812B1	%ALOG	(% → 10^%)
81342	%SIN	(% → SIN%)
812E3	%COS	(% → COS%)
81356	%TAN	(% → TAN%)
812BB	%ASIN	(% → ASIN%)
812A7	%ACOS	(% → ACOS%)
812C5	%ATAN	(% → ATAN%)
81347	%SINH	(% → SINH%)
812E8	%COSH	(% → COSH%)
8135B	%TANH	(% → TANH%)
812C0	%ASINH	(% → ASINH%)
812AC	%ACOSH	(% → ACOSH%)

812CA	%ATANH	(% → ATANH%)
8131A	%MANTISSA	(% → %mant)
812F7	%EXPONENT	(% → %expn)
81301	%FP	(% → %frac)
81306	%IP	(% → %int)
812FC	%FLOOR	(% → %maxint <=%)
812CF	%CEIL	(% → %minint >=%)
8131F	%MOD	(% %' → %rem)
812B6	%ANGLE	(%x %y → %ang)
82FDF	RNDXY	(% %places → %')
82FF3	TRCXY	(% %places → %')
812DE	%COMB	(% %' → COMB(%,%'))
81333	%PERM	(% %' → PERM(%,%'))
81324	%NFACT	(% → %!) Calculates factorial of number.
82F5D	%FACT	(% → gamma(%+1)) Calculates gamma(x+1).
81329	%NROOT	(% %n → %') Calculates the %nth root of the real number. Equivalent to user function XROOT.
82F99	%MIN	(% %' → %lesser)
82F8F	%MAX	(% %' → %greater)
837F0	%MAXorder	(% %' → %max %min)
81338	%RAN	(→ %random) Returns next random number.
82FB7	%RANDOMIZE	(%seed →) System level RDZ: seeds the random number generator.
8132E	%OF	(% %' → %'/% * 100)
81351	%T	(% %' → %pctotal)
812D4	%CH	(% %' → %pcchange)
82F53	%D>R	(%deg → %rad)
82FAD	%R>D	(%rad → %deg)
82FC1	%REC>%POL	(%r %ang → %x %y)
82FA3	%POL>%REC	(%x %y → %r %ang)
82FCB	%SPH>%REC	(%r %ang %ph → %x %y %z)

2.2.5 Extended Real Functions

845C9	SWAP%%/	(%% %%' → %%'')

8458D	%%/>%	(%% %%' → %)

2.2.6 Tests

8127F	%=	(% %' → flag)
8127A	%<>	(% %' → flag)
8126B	%<	(% %' → flag)
81270	%<=	(% %' → flag)
81284	%>	(% %' → flag)
81275	%>=	(% %' → flag)
81266	%0=	(% → flag)
845B5	DUP%0=	(% → flag)
82F3F	%0<>	(% → flag) Can be used to change a user flag into a system flag.
81261	%0<	(% → flag)
8125C	%0>	(% → flag)
82F49	%0>=	(% → flag)

2.3 Complex Numbers

2.3.1 Conversion

81BF8	%>C%	(%re %im → C%)
83D18	SWAP%>C%	(%im %re → C%)

2.4 Character Strings

2.4.1 Built-in Characters

80C80	CHR_00	'\00', CHR 0d 00h The NULL character.
80ED3	CHR_Newline	'\0a', CHR 10d 0Ah
80C87	CHR_...	'...', CHR 31d 1Fh
80EEF	CHR_Space	' ', CHR 32d 20h The space character.
80C95	CHR_DblQuote	'"', CHR 34d 22h
80C9C	CHR_#	'#', CHR 35d 23h

```
80ECC     CHR_LeftPar          '(', CHR 40d 28h
80EE1     CHR_RightPar         ')', CHR 41d 29h
80CA3     CHR_*                '*', CHR 42d 2Ah
80CAA     CHR_+                '+', CHR 43d 2Bh
80CB1     CHR_,                ',', CHR 44d 2Ch
80CB8     CHR_-                '-', CHR 45d 2Dh
80CBF     CHR_.                '.', CHR 46d 2Eh
80CC6     CHR_/                '/', CHR 47d 2Fh
80CCD     CHR_0                '0', CHR 48d 30h
80CD4     CHR_1                '1', CHR 49d 31h
80CDB     CHR_2                '2', CHR 50d 32h
80CE2     CHR_3                '3', CHR 51d 33h
80CE9     CHR_4                '4', CHR 52d 34h
80CF0     CHR_5                '5', CHR 53d 35h
80CF7     CHR_6                '6', CHR 54d 36h
80CFE     CHR_7                '7', CHR 55d 37h
80D05     CHR_8                '8', CHR 56d 38h
80D0C     CHR_9                '9', CHR 57d 39h
80D13     CHR_:                ':', CHR 58d 3Ah
80D1A     CHR_;                ';', CHR 59d 3Bh
80D21     CHR_<                '<', CHR 60d 3Ch
80D28     CHR_=                '=', CHR 61d 3Dh
80D2F     CHR_>                '>', CHR 62d 3Eh
80D36     CHR_A                'A', CHR 65d 41h
80D3D     CHR_B                'B', CHR 66d 42h
80D44     CHR_C                'C', CHR 67d 43h
80D4B     CHR_D                'D', CHR 68d 44h
80D52     CHR_E                'E', CHR 69d 45h
80D59     CHR_F                'F', CHR 70d 46h
80D60     CHR_G                'G', CHR 71d 47h
80D67     CHR_H                'H', CHR 72d 48h
80D6E     CHR_I                'I', CHR 73d 49h
80D75     CHR_J                'J', CHR 74d 4Ah
80D7C     CHR_K                'K', CHR 75d 4Bh
80D83     CHR_L                'L', CHR 76d 4Ch
80D8A     CHR_M                'M', CHR 77d 4Dh
80D91     CHR_N                'N', CHR 78d 4Eh
80D98     CHR_O                'O', CHR 79d 4Fh
```

```
80D9F     CHR_P              'P', CHR 80d 50h
80DA6     CHR_Q              'Q', CHR 81d 51h
80DAD     CHR_R              'R', CHR 82d 52h
80DB4     CHR_S              'S', CHR 83d 53h
80DBB     CHR_T              'T', CHR 84d 54h
80DC2     CHR_U              'U', CHR 85d 55h
80DC9     CHR_V              'V', CHR 86d 56h
80DD0     CHR_W              'W', CHR 87d 57h
80DD7     CHR_X              'X', CHR 88d 58h
80DDE     CHR_Y              'Y', CHR 89d 59h
80DE5     CHR_Z              'Z', CHR 90d 5Ah
80EFD     CHR_[              '[', CHR 91d 5Bh
80F04     CHR_]              ']', CHR 93d 5Dh
80EF6     CHR_UndScore       '_', CHR 95d 5Fh
80DEC     CHR_a              'a', CHR 97d 61h
80DF3     CHR_b              'b', CHR 98d 62h
80DFA     CHR_c              'c', CHR 99d 63h
80E01     CHR_d              'd', CHR 100d 64h
80E08     CHR_e              'e', CHR 101d 65h
80E0F     CHR_f              'f', CHR 102d 66h
80E16     CHR_g              'g', CHR 103d 67h
80E1D     CHR_h              'h', CHR 104d 68h
80E24     CHR_i              'i', CHR 105d 69h
80E2B     CHR_j              'j', CHR 106d 6Ah
80E32     CHR_k              'k', CHR 107d 6Bh
80E39     CHR_l              'l', CHR 108d 6Ch
80E40     CHR_m              'm', CHR 109d 5Dh
80E47     CHR_n              'n', CHR 110d 6Eh
80E4E     CHR_o              'o', CHR 111d 6Fh
80E55     CHR_p              'p', CHR 112d 70h
80E5C     CHR_q              'q', CHR 113d 71h
80E63     CHR_r              'r', CHR 114d 72h
80E6A     CHR_s              's', CHR 115d 73h
80E71     CHR_t              't', CHR 116d 74h
80E78     CHR_u              'u', CHR 117d 75h
80E7F     CHR_v              'v', CHR 118d 76h
80E86     CHR_w              'w', CHR 119d 77h
80E8D     CHR_x              'x', CHR 120d 78h
```

80E94	CHR_y	'y', CHR 121d 79h
80E9B	CHR_z	'z', CHR 122d 7Ah
80F0B	CHR_{	'{', CHR 123d 7Bh
80F12	CHR_}	'{', CHR 125d 7Dh
80EB7	CHR_Angle	'∠', CHR 128d 80h
80EC5	CHR_Integral	'∫', CHR 132d 84h
80EBE	CHR_Deriv	'∂', CHR 136d 88h
80EA2	CHR_->	'→', CHR 141d 8Dh
80EA9	CHR_<<	'≪', CHR 171d ABh
80EB0	CHR_>>	'≫', CHR 187d BBh
80EDA	CHR_Pi	'π', CHR 135d 87h
80EE8	CHR_Sigma	'Σ', CHR 133d 85h
80F19	CHR_<=	'≤', CHR 137d 89h
80F20	CHR_>=	'≥', CHR 138d 8Ah
80F27	CHR_<>	'≠', CHR 139d 8Bh
80C8E	CHR_'	

2.4.2 Built-in Strings

055BF	NULL$	""
		Empty string.
80B10	SPACE$	" "
		aka: tok_
80ACE	14SPACES$	" "
		String of 14 spaces.
80AF4	NEWLINE$	"\0a"
		Newline.
0E909	CRLF$	"\0d\0a"
		Carriage return and line feed.
80B70	toklparen	"("
80B7C	tokrparen	")"
80A26	tok[	"["
80A0C	tok]	"]"
80A32	tok{	"{"
80A3E	tok}	"}"
80A92	tok<<	"≪"
80A86	tok>>	"≫"
80FF1	$_LRParens	"()"
80FB9	$_{}	"{}"
80FAB	$_<<>>	"≪≫"

80FD5	$_’’	"’’"
		Two single quotes.
80FE3	$_::	"::"
80FFF	$_2DQ	""""
		Two double quotes.
80B4C	tok,	","
80B40	tok’	"’"
		One single quote.
80BB8	tok-	"-"
80B58	tok.	"."
80BC4	tok=	"="
80C08	tok0	"0"
80C14	tok1	"1"
80C20	tok2	"2"
80C2C	tok3	"3"
80C38	tok4	"4"
80C44	tok5	"5"
80C50	tok6	"6"
80C5C	tok7	"7"
80B64	tok;	";"
80C68	tok8	"8"
80C74	tok9	"9"
80A7A	tokESC	"\1B"
		Escape character.
80A9E	tokexponent	"E"
80B34	tokquote	"""
		One double quote.
80A4A	toksharp	"#"
80A62	tok$	"$"
80A6E	tok&	"&"
80B94	tok*	"*"
80BAC	tok+	"+"
80BA0	tok/	"/"
80AAA	tokanglesign	"$\angle$"
80BDC	tokDER	"∂"
80B00	$DER	"der"
80AB6	tokSIGMA	"Σ"
80BD0	tokSQRT	"$\sqrt{a}$"
80A56	tokuscore	"_"
80AC2	tokWHERE	"|"

80B88	tok^	"^"
4F7F8	tok:	":"
8104D	$_RAD	"RAD"
8105D	$_GRAD	"GRAD"
80F9B	$_XYZ	"XYZ"
80F8B	$_R<Z	"R∠Z" "R<angle>Z"
80F7B	$_R<<	"R∠∠" "R<angle><angle>"
8101F	$_EXIT	"EXIT"
8100D	$_ECHO	"ECHO"
81031	$_Undefined	"Undefined"
80BE8	tokCTGROB	"GROB"
80BFA	tokCTSTR	"C$"
80B1C	tokUNKNOWN	"UNKNOWN"

2.4.3 Built-in Strings with Stack Manipulation

837C8	NULL$SWAP	(ob → $ ob) NULL$, then SWAP.
819FA	DROPNULL$	(ob → NULL$) DROP then NULL$.
819FF	TWODROPNULL$	(ob ob' → NULL$) 2DROP then NULL$.

2.4.4 Conversion

818C9	#>$	(# → $) Creates string from the bint (decimal).
81C07	ID>$	(id/lam → $) Converts identifier into string.

2.4.5 Management

81BE9	#>CHR	(# → chr) Returns character with the specified ASCII code.
84ABA	CHR>$	(chr → $* Strings) Converts a character into a string.
819DC	LEN$	($ → #length) Returns length in bytes.
8154A	DUPLEN$	($ → $ #) DUP then LEN$.

83C00	NEWLINE$&$	($ → "$\0a") Appends newline character to string. aka: NEWLINE&$
819D7	CAR$	($ → chr) ($ → "") Returns first character of string as a string, or NULL$ for null string.
83179	CDR$	($ → $') Returns string without first character, or NULL$ for null string.
81B3A	POS$	($ $find start# → #pos) ($ $find start# → #0) Search for $find in $search, starting at position #start. Returns position of $find or 0 if not found. Same entry as POSCHR.
81B44	POSCHR	($search chr #start → #pos) ($search chr #start → #0) Same entry as <REF>POS$.
81B3F	POS$REV	($ $find #limit → #pos) ($ $find #limit → #0) Searches backwards from #limit to #1. Same entry as <REF>POSCHRREV.
81B49	POSCHRREV	($seach chr #start → #pos) ($seach chr #start → #0) Same entry as <REF>POS$REV.
819E6	SUB$	($ #start #end → $') Returns substring between specified positions.
83CB4	#1-SUB$	($ #start #end+#1 → $') Does #1- and then SUB$.
83CC8	1_#1-SUB$	($ #end → $') Returns substring with the first #end characters. aka: 1_#1-SUB
83CDC	LAST$	($ #start → $') Returns substring from the specified start position to the end (inclusive).
83CF0	#1+LAST$	($ #start-#1 → $') Returns substring from the specified start position to the end (exclusive).
837DC	SUB$SWAP	(ob $ # #' → $' ob) SUB$ then SWAP.
81A72	SUB$1#	($ #pos → #') Returns bint with ASCII code of character at the specified position.
81491	EXPAND	(hxs #nibs → hxs') Appends #nibs zero nibbles to the hxs.

819C8	&$	($ $' → $+$') Concatenates two strings.
84975	&$SWAP	(ob $ $' → $+$' ob) &$ then SWAP.
8316F	!append$	($ $' → $+$') Tries &$, if not enough memory does !!append$?.
8399E	!append$SWAP	(ob $ $' → $+$' ob) !append$ then SWAP.
81590	!!append$	($ $' → $+$') Tries appending "in place".
819CD	>H$	($ chr → $') Prepends character to string
819D2	>T$	($ chr → $') Appends character to string.
8361F	APPEND_SPACE	($ → $') Appends space to string.
831B5	SWAP&$	($ $' → $'+$) Concatenates two strings.

2.4.6 Parsing Strings

82E45	!*triand	(T T →) (F T → F T <SEMI>)
81C89	tok8cktrior	($1 $1 → :: $1 <Ob1> ;) ($1 $2 → :: $1 <Ob2> <Rest> ;)
81C8E	tok8trior	(GNT data $1 $1 → :: GNT data GetNextToken ;) (GNT data $1 $2 → :: $1 <Ob1> <Rest> ;)
81C93	nultrior	(NULL$ → :: ;) ($ → :: $ <Ob1> <Rest> ;)
82E27	GetNextToken	(hxs-mask $ #start → hxs-mask $ #next $token)

2.4.7 Decompilation

83197	EDITDECOMP$	(ob → $) Calls setStdEditWid and the decompiles for editing like <REF>editdecomp$w.
83183	DECOMP$	(ob → $) Calls <REF>setStdWid and decompiles entire object (UserRPL components only). Breaks the string into lines using DcompWidth as width.
8318D	DO>STR	($ → $) (ob → $) Internal version of →STR.

831BF Decomp%Short (% #width → $)
Decompiles a real number into a string of the given #width. It will drop less significant digits or add zeros as needed, but will also exceed #width when necessary. E.g. "-1.e-33" cannot be written with less than 7 characters, so even if #width is less, 7 chars will be used. %0 is always decompiled as "0".

2.4.8 String Tests

819E1 NULL$? (ob → flag)
83C78 DUPNULL$? (ob → ob flag)

2.5 HEX Strings

2.5.1 Conversion

81BEE #>HXS (# → hxs)
Length will be five.

2.5.2 General Functions

055B5 NULLHXS HXS 0
Puts a null hxs in the stack.
81491 EXPAND (hxs #nibs → hxs')
Appends #nibs zero nibbles to the hxs.

2.6 Arrays

2.6.1 General Functions

81BD5 ARSIZE ([] → #)
Returns number of elements as a bint.
81BDA DIMLIMITS ([] → {#n #m})
Returns list of array dimensions.
81BDF GETATELN (# [] → ob T)
(# [] → F)
Gets one element from array.
81E64 MATCON ([%] % → [%]')
([C%] C% → [C%]')
Replace all elements of [F%] by F%.

81AA4	MDIMS	([1D] → #m F) ([2D] → #m #n T) If it is a vector, returns number of elements and FALSE. If it is an array (including arrays with only one line), returns dimensions and TRUE.
83A0C	MDIMSDROP	([2D] → #m #n) MDIMS followed by DROP.
83BB0	OVERARSIZE	([] ob → [] ob #elts) Does OVER then <REF>ARSIZE.
81AA9	PUTEL	([%] % # → [%]') ([C%] C% # → [C%]') Puts element at specified position. Converts to "short" before. Warning: no copy to tempob first.
81E6E	MATREDIM	([F%] {#n #m} → [F%]')
81E78	MATTRN	([F%] → [F%]') Transposes matrix.

2.6.2 Conversion

81BE4	MAKEARRY	({#n #m} ob → []) Makes array with all elements initialized to ob.

2.7 Unit Objects

2.7.1 Creating Units

81F0E	EXTN	(ob1..obn #n → u) Builds a unit object.

2.8 Composites

2.8.1 General Operations

81F18	&COMP	(comp comp' → comp'') Concatenates two composites.
81E96	>TCOMP	(comp ob → comp+ob) Adds ob to tail (end) of composite.
81E8C	>HCOMP	(comp ob → ob+comp) Adds ob to head (beginning) of composite.
819A0	CARCOMP	(comp → ob_head) (comp_null → comp_null) Returns first object of the composite, or a null composite if the argument is a null composite.

83BEC	?CARCOMP	(comp T → ob) (comp F → comp) If the flag is TRUE, does CARCOMP.
81EA0	CDRCOMP	(comp → comp-ob_head) (comp_null → comp_null) Returns the composite minus its first object, or a null composite if the argument is a null composite.
819AA	LENCOMP	(comp → #n) Returns length of composite (number of objects).
83CA0	DUPLENCOMP	(comp → comp #n) Does DUP then <REF>LENCOMP.
819B4	NULLCOMP?	(comp → flag) If the composite is empty, returns TRUE.
83C8C	DUPNULLCOMP?	(comp → comp flag) Does DUP then <REF>NULLCOMP?.
819AF	NTHELCOMP	(comp #i → ob T) (comp #i → F) Returns specified element of composite and TRUE, or just FALSE if it could not be found.
8360B	NTHCOMPDROP	(comp #i → ob) Does <REF>NTHELCOMP then DROP.
8378C	NTHCOMDDUP	(comp #i → ob ob) Does <REF>NTHCOMPDROP then DUP.
81EDC	POSCOMP	(comp ob pred → #i) (comp ob pred → #0) (eg: pred = ' %<) Evaluates pred for all elements of composite and ob, and returns index of first object for which the pred is TRUE. If no one returned TRUE, returns #0. For example, the program below returns #4: :: { %1 %2 %3 %-4 %-5 %6 %7 } %0 ' %< POSCOMP ;
84A6F	EQUALPOSCOMP	(comp ob → #pos) (comp ob → #0) POSCOMP with EQUAL as test.
81EBE	EQUALPOSCMP	(comp ob → #pos) (comp ob → #0)
84A88	NTHOF	(ob comp → #i) (ob comp → #0) Does SWAP then <REF>EQUALPOSCOMP.
84A56	#=POSCOMP	(comp # → #i) (comp # → #0) POSCOMP with #= as test.
819B9	SUBCOMP	(comp #m #n → comp') Returns a sub-composite. Makes all index checks first.

81392	Embedded?	(ob1 ob2 → flag) Returns TRUE if ob2 is embedded in, or is the same as, ob1. Otherwise returns FALSE.
81EC8	Lookup	(ob test comp → nextob T) (ob test comp → ob F) Tests every odd element (1,3,...) in the composite. If a test returns TRUE, the object after the tested one is returned, along with TRUE. If no object tests TRUE, FALSE is returned. For example, the program below returns %6 and TRUE. :: %0 ' %< { %1 %2 %3 %-4 %-5 %6 } Lookup ;
81ED2	Lookup.1	(ob test → nextob T) (ob test → ob F) Return Stack: (comp →) Lookup with the composite already pushed (with >R) onto the runstream. Called by Lookup.
84A9C	EQLookup	(ob comp → nextob T) (ob comp → ob F) Lookup with EQ as test.
81B35	NEXTCOMPOB	(comp #ofs → comp #ofs' ob T) (comp #ofs → comp F) Returns object at specified nibble offset from start. If the object is SEMI (i.e., the end of the composite has been reached) returns FALSE. To get the first element, use FIVE as offset value (to skip the prolog). ZERO works as well.

2.8.2 Building

81EFA	{}N	(obn..ob1 #n → { obn..ob1 })
81EF0	::N	(ob1..obn #n → :: ob1..obn ;)
81F04	SYMBN	(ob1..obn #n → sym)
8490C	top&Cr	(meta1 meta2 → symb) Does top& then <REF>SYMBN: .
81F0E	EXTN	(ob1..obn #n → u) Builds a unit object.
82E77	P::N	(ob1..obn #n → seco) Build seco with possible garbage collection.

2.8.3 Exploding

819A5	INNERCOMP	(comp → obn..ob1 #n)
83C50	DUPINCOMP	(comp → comp obn..ob1 #n)
83C64	SWAPINCOMP	(comp obj → obj obn..ob1 #n)
835F7	INCOMPDROP	(comp → obn..ob1)
836B0	INNERDUP	(comp → obn..ob1 #n #n)
840F6	INNER#1=	(comp → obn..ob1 flag)

2.8.4 Lists

055C9	NULL{}	(→ {}) Pushes a null list to the stack.
8447A	DUPNULL{}?	({} → {} flag)
83C28	TWO{}N	(ob1 ob2 → { ob1 ob2 })
83C3C	THREE{}N	(ob1 ob2 ob3 → { ob1 ob2 ob3 })
83C14	#1-{}N	(ob1..obn #n+1 → {})
81EE6	PUTLIST	(ob #i {} → {}') Replaces object at specified position. Assumes valid #i.

2.8.5 Secondaries

055DD	NULL::	(→ :: ;) Returns null secondary.
849F2	Ob>Seco	(ob → :: ob ;) Does ONE then <REF>::N.
849D9	?Ob>Seco	(ob → :: ob ;) If the object is not a secondary, does Ob>Seco.
84A06	2Ob>Seco	(ob1 ob2 → :: ob1 ob2 ;) Does TWO then <REF>::N.
83D40	::NEVAL	(ob1..obn #n → ?) Does <REF>::N then <REF>EVAL.

2.9 Meta Objects

2.9.1 Stack Functions

81D5B	NDROP	(1..n #n →)
849B1	DROPNDROP	(1..n #n ob →)
839E4	#1+NDROP	(ob 1..n #n →) aka: N+1DROP

81D01	psh	(meta1 meta2 → meta2 meta1) Should be called swap.
81D06	roll2ND	(meta1 meta2 meta3 → meta2 meta3 meta1) Should be called rot.
81D10	unroll2ND	(meta1 meta2 meta3 → meta3 meta1 meta2) Should be called unrot.
8431C	SWAPUnNDROP	(meta1 meta2 → meta2) Should be called swapdrop.
84308	SWAPUnDROP	(meta1 meta2 → meta2 ob1..obn) Swaps two metas and drops the count. Should be called swapDROP.
84925	metaROTDUP	(meta1 meta2 meta3 → meta2 meta3 meta1 meta1) Should be called rotdup.

2.9.2 Combining Functions

81CF2	top&	(meta1 meta2 → meta1&meta2)
8309D	pshtop&	(meta1 meta2 → meta2&meta1)
84939	ROTUntop&	(meta1 meta2 meta3 → meta2 meta3&meta1)
8494D	roll2top&	(meta1 meta2 meta3 → meta3 meta1&meta2) aka: rolltwotop&
83061	psh&	(meta1 meta2 meta3 → meta1&meta3 meta2)

2.9.3 Meta and Object Operations

81568	SWAP#1+	(meta ob → meta&ob) aka: SWP1+
815B8	get1	(ob meta → meta ob)
83075	psh1top&	(meta ob → ob&meta)
830B1	pull	(meta&ob → meta ob) aka: #1-SWAP
830BB	pullrev	(ob&meta → meta ob)
81CF7	psh1&	(meta1 meta2 ob → ob&meta1 meta2)
81CFC	psh1&rev	(meta1 meta2 ob → ob&meta1 meta2)
830C5	pullpsh1&	(meta1 meta2&ob → ob&meta1 meta2)
830A7	pshzer	(meta → #0 meta)

2.10 Symbolics

2.10.1 General Operations

81F04	SYMBN	(ob1..obn #n → sym)
055D3	NULLSYMB	(→ sym) Puts a null algebraic in the stack.
8320F	symcomp	(ob → ob') If ob is symbolic, does nothing, otherwise ONE SYMBN.
82E8B	SWAPcompSWAP	(ob ob' → ob'' ob') Does SWAP symcomp SWAP.

2.10.2 Derivatives

2.10.3 Meta Symbolics Functions

8499D	pZpargSWAPUn	(meta → M_rest M_last) <REF>pshzerpsharg then <REF>psh .
84961	plDRPpZparg	(meta&ob → M_last M_rest) Drops ob then calls <REF>pshzerpsharg .

2.11 Library and Backup Objects

2.11.1 Rompointers

81C0C	#>ROMPTR	(#lib #cmd → ROMPTR) Creates rompointer.
81C66	ROMPTR>#	(ROMPTR → #lib #cmd) Splits rompointer.
81C6B	ROMPTR@	(ROMPTR → ob T) (ROMPTR → F) Recalls contents of rompointer.
83688	DUPROMPTR@	(ROMPTR → ROMPTR ob T) (ROMPTR → ROMPTR F) Does DUP then ROMPTR@.
82FFD	?>ROMPTR	(ob → ob') If ROM-WORD? and TYPECOL? then RPL@.
83007	?ROMPTR>	(ob → ob') If <REF>TYPEROMP? and content exists <REF>INHARDROM? then return contents.
83011	RESOROMP	(→ ob) Recalls contents of next object in the runstream (which must be a rompointer).

81C20	COMPILEID	(id → id T) (id → ROMPTR T) (id → F) Searches id in current path, if found returns TRUE. Else searches attached libraries. If nothing was found, return FALSE.
81496	ROM-WORD?	(ob → flag)

2.11.2 Libraries

81C84	TOSRRP	(# →) Attaches library to HOME directory. -- <REF>TEXT:Libraries
81C43	OFFSRRP	(# →) Detaches library from HOME directory. -- <REF>TEXT:Libraries
81C48	ONSRRP?	(# → flag) Returns TRUE if library is attached to HOME directory.
82D5F	XEQSETLIB	(% →) Internal ATTACH.

2.11.3 Backup Objects

81720	BAKNAME	(bak → id T) Returns backup's name
81716	BAK>OB	(bak → ob) Gets backup object.

3 General SysRPL Entries

3.1 Stack Operations

81D3D	DUP	(ob → ob ob)
83728	DUPDUP	(ob → ob ob ob)
8304D	NDUPN	(ob #n → ob..ob #n) (ob #0 → #0)
83A20	DUPROT	(1 2 → 2 2 1)
815C7	DUPUNROT	(1 2 → 2 1 2) aka: SWAPOVER
83B60	DUPROLL	(1..n #n → 1 3..n #n 2)
83B4C	DUPPICK	(n..1 #n → n..1 #n n-1)
81D42	2DUP	(1 2 → 1 2 1 2)
8464B	2DUP5ROLL	(1 2 3 → 2 3 2 3 1)
81D47	NDUP	(1..n #n → 1..n 1..n)
81D51	DROP	(1 →)
81545	DROPDUP	(1 2 → 1 1)
849B1	DROPNDROP	(1..n #n ob →)
8152C	DROPSWAP	(1 2 3 → 2 1)
83A34	DROPROT	(1 2 3 4 → 2 3 1)
83A98	DROPOVER	(1 2 3 → 1 2 1)
81D56	2DROP	(1 2 →)
8149B	3DROP	(1 2 3 →) aka: XYZ>
814AA	4DROP	(1..4 →) aka: XYZW>
814BE	5DROP	(1..5 →)
814D2	6DROP	(1..6 →)
81450	7DROP	(1..7 →)
81D5B	NDROP	(1..n #n →)
839E4	#1+NDROP	(ob 1..n #n →) aka: N+1DROP
81D38	DEPTH	(1..n → 1..n #n)
830CF	reversym	(1..n #n → n..1 #n)
81D4C	SWAP	(1 2 → 2 1)
81536	SWAPDUP	(1 2 → 2 1 1)
8425E	SWAP2DUP	(1 2 → 2 1 2 1)
81455	SWAPDROP	(1 2 → 2) aka: XY>Y

81554	SWAPDROPDUP	(1 2 → 2 2)
8144B	SWAPROT	(1 2 3 → 3 2 1) aka: UNROTSWAP, XYZ>ZYX
84637	SWAP4ROLL	(1 2 3 4 → 2 4 3 1) aka: XYZW>YWZX
8465F	SWAP3PICK	(1 2 3 → 1 3 2 1)
81374	2SWAP	(1 2 3 4 → 3 4 1 2)
81D60	ROT	(1 2 3 → 2 3 1)
8153B	ROTDUP	(1 2 3 → 2 3 1 1)
836EC	ROT2DUP	(1 2 3 → 2 3 1 3 1)
81446	ROTDROP	(1 2 3 → 2 3) aka: XYZ>YZ
81531	ROT2DROP	(1 2 3 → 2) aka: DROPSWAPDROP, XYZ>Y
81441	ROTDROPSWAP	(1 2 3 → 3 2) aka: XYZ>ZY
8143C	ROTSWAP	(1 2 3 → 2 1 3) aka: XYZ>YXZ
8145F	ROTROT2DROP	(1 2 3 → 3) aka: UNROT2DROP, XYZ>Z
83714	ROTOVER	(1 2 3 → 2 3 1 3)
814B4	4ROLL	(1 2 3 4 → 2 3 4 1) aka: FOURROLL, XYZW>YZWX
8393A	4ROLLSWAP	(1 2 3 4 → 2 3 1 4)
83A70	4ROLLROT	(1 2 3 4 → 2 4 1 3) aka: FOURROLLROT
83B10	4ROLLOVER	(1 2 3 4 → 2 3 4 1 4)
814C8	5ROLL	(1 2 3 4 5 → 2 3 4 5 1) aka: FIVEROLL
815BD	6ROLL	(1..6 → 2..6 1) aka: SIXROLL
81379	7ROLL	(1..7 → 2..7 1) aka: SEVENROLL
8137E	8ROLL	(1..8 → 2..8 1) aka: EIGHTROLL
8145A	9ROLL	(1..9 → 2..9 1)
81D6F	ROLL	(1..n #n → 2..n 1)
839F8	ROLLDROP	(1..n #n → 2..n)
837B4	ROLLSWAP	(1..n #n → 2..n-1 1 n)
814FA	#1+ROLL	(ob 1..n #n → 1..n ob)
814FF	#2+ROLL	(a b 1..n #n → b 1..n a)
814F5	#+ROLL	(1..n+m #n #m → 2..n+m 1)
814F0	#-ROLL	(1..n-m #n #m → 2..n-m 1)

815CC	UNROT	(1 2 3 → 3 1 2)
		aka: 3UNROLL, XYZ>ZXY
83764	UNROTDUP	(1 2 3 → 3 1 2 1)
81559	UNROTDROP	(1 2 3 → 3 1)
		aka: SWAPDROPSWAP, XYZ>ZX
83AFC	UNROTOVER	(1 2 3 → 3 1 2 1)
814A5	3UNROLL	(1 2 3 → 3 1 2)
		aka: UNROT, XYZ>ZXY
814B9	4UNROLL	(1 2 3 4 → 4 1 2 3)
		aka: FOURUNROLL, XYZW>WXYZ
83778	4UNROLLDUP	(1 2 3 4 → 4 1 2 3 3)
81464	4UNROLL3DROP	(1 2 3 4 → 4)
		aka: XYZW>W
83A84	4UNROLLROT	(1 2 3 4 → 4 3 2 1)
814CD	5UNROLL	(1 2 3 4 5 → 5 1 2 3 4)
		aka: FIVEUNROLL
815C2	6UNROLL	(1..6 → 6 1..5)
		aka: SIXUNROLL
83633	7UNROLL	(1..7 → 7 1..6)
83B88	8UNROLL	(1..8 → 8 1..7)
83B9C	10UNROLL	(1..10 → 10 1..9)
81D74	UNROLL	(1..n #n → n 1..n-1)
8150E	#1+UNROLL	(ob 1..n #n → n ob 1..n-1)
81513	#2+UNROLL	(a b 1..n #n → n a b 1..n-1)
81509	#+UNROLL	(1..n+m #n #m → n+m 1..n+m-1)
81504	#-UNROLL	(1..n-m #n #m → n-m 1..n+m-1)
81D65	OVER	(1 2 → 1 2 1)
8373C	OVERDUP	(1 2 → 1 2 1 1)
837A0	OVERSWAP	(1 2 → 1 1 2)
		aka: OVERUNROT
837A0	OVERUNROT	(1 2 → 1 1 2)
		aka: OVERSWAP
8469B	OVER5PICK	(1 2 3 4 → 1 2 3 4 3 1)
849C5	2OVER	(1 2 3 4 → 1 2 3 4 1 2)
814A0	3PICK	(1 2 3 → 1 2 3 1)
8394E	3PICKSWAP	(1 2 3 → 1 2 1 3)
83B24	3PICKOVER	(1 2 3 → 1 2 3 1 3)
84673	3PICK3PICK	(1 2 3 → 1 2 3 1 2)
DDCFE	4PICK	(1 2 3 4 → 1 2 3 4 1)
83962	4PICKSWAP	(1 2 3 4 → 1 2 3 1 4)
84687	SWAP4PICK	(1 2 3 4 → 1 2 4 3 1)
83B38	4PICKOVER	(1 2 3 4 → 1 2 3 4 1 4)

814C3	5PICK	(1 2 3 4 5 → 1 2 3 4 5 1)
814D7	6PICK	(1..6 → 1..6 1)
814DC	7PICK	(1..7 → 1..7 1)
814E1	8PICK	(1..8 → 1..8 1)
814E6	9PICK	(1..9 → 1..9 1)
814EB	10PICK	(1..10 → 1..10 1)
81D6A	PICK	(1..n #n → 1..n 1)
81478	#1+PICK	(1..n #n-1 → 1..n 1)
8147D	#2+PICK	(1..n #n-2 → 1..n 1)
81482	#3+PICK	(1..n #n-3 → 1..n 1)
81487	#4+PICK	(1..n #n-4 → 1..n 1)
81473	#+PICK	(1..n+m #n #m → 1..n+m 1)
8146E	#-PICK	(1..n-m #n #m → 1..n-m 1)

3.2 Temporary Environments

3.2.1 Built-in IDs and LAMs

82D4B	NULLID	(→ id) Null (empty) identifier.
84AD8	NULLLAM	(→ lam) Puts NULLLAM in the stack.

3.2.2 Conversion

81BF3	$>ID	($ → ID)
83D04	DUP$>ID	($ → $ ID)

3.2.3 Temporary Environments Words

831C9	BIND	(obn..ob1 {lamn..lam1} →) Binds n objects to n differently named lams.
817C0	DOBIND	(obn..ob1 lamn..lam1 #n →) Binds n objects to n differently named lams.
83F3E	1LAMBIND	(ob →) Binds one object to a null named lam.
83F39	DUP1LAMBIND	(ob → ob) Does DUP then <REF>1LAMBIND.
84434	dvarlsBIND	(ob →) Binds ob to LAM 'dvar.

817BB	ABND	(→) Abandons topmost temporary environment.
816B2	CACHE	(obn..ob1 #n lam →) Binds all objects under the same name. 1LAM has the count.
81383	DUMP	(NULLLAM → ob1..obn #n) Inverse of CACHE. Always does garbage collection.
81C16	@LAM	(lam → ob T) (lam → F) Tries recalling object from lam. If successful, returns object and TRUE, otherwise returns just FALSE.
81C75	STOLAM	(ob lam →) Tries storing object in lam. Generates "Undefined Local Name" error if lam is not found.
817C5	GETLAM	(#n → ob) Gets contents of nth topmost lam.
815D6	1GETLAM	(→ ob)
815E0	2GETLAM	(→ ob)
815EA	3GETLAM	(→ ob)
815F4	4GETLAM	(→ ob)
815FE	5GETLAM	(→ ob)
81608	6GETLAM	(→ ob)
81612	7GETLAM	(→ ob)
8161C	8GETLAM	(→ ob)
81626	9GETLAM	(→ ob)
81630	10GETLAM	(→ ob)
8163A	11GETLAM	(→ ob)
81644	12GETLAM	(→ ob)
8164E	13GETLAM	(→ ob)
81658	14GETLAM	(→ ob)
81662	15GETLAM	(→ ob)
8166C	16GETLAM	(→ ob)
81676	17GETLAM	(→ ob)
81680	18GETLAM	(→ ob)
8168A	19GETLAM	(→ ob)
81694	20GETLAM	(→ ob)
8169E	21GETLAM	(→ ob)
816A8	22GETLAM	(→ ob)
817CA	PUTLAM	(ob #n →) Stores new contents to nth topmost lam.
815DB	1PUTLAM	(ob →)

815E5	2PUTLAM	(ob →)
815EF	3PUTLAM	(ob →)
815F9	4PUTLAM	(ob →)
81603	5PUTLAM	(ob →)
8160D	6PUTLAM	(ob →)
81617	7PUTLAM	(ob →)
81621	8PUTLAM	(ob →)
8162B	9PUTLAM	(ob →)
81635	10PUTLAM	(ob →)
8163F	11PUTLAM	(ob →)
81649	12PUTLAM	(ob →)
81653	13PUTLAM	(ob →)
8165D	14PUTLAM	(ob →)
81667	15PUTLAM	(ob →)
81671	16PUTLAM	(ob →)
8167B	17PUTLAM	(ob →)
81685	18PUTLAM	(ob →)
8168F	19PUTLAM	(ob →)
81699	20PUTLAM	(ob →)
816A3	21PUTLAM	(ob →)
816AD	22PUTLAM	(ob →)
83F25	1GETABND	(→ 1lamob) Does <REF>1GETLAM then <REF>ABND .
83822	1ABNDSWAP	(ob → 1lamob ob) Does <REF>1GETABND then SWAP.
83976	1GETSWAP	(ob → 1lamob ob) Does <REF>1GETLAM then SWAP.
83D54	2GETEVAL	(→ ?) Does <REF>2GETLAM then <REF>EVAL .
81397	GETLAMPAIR	(#n → #n ob lam F) (#n → #n T) Gets lam contents and name (10 = 1lam, 20 = 2lam, etc.)
84AD3	1NULLLAM{}	(→ {}) Puts a list with one NULLLAM in the stack.

3.3 Error Handling

3.3.1 General Words

81B17 ERRBEEP
(→)
Beeps.

819EB ERROR@
(→ #)
Returns current error number.

819F0 ERRORSTO
(# →)
Stores new error number.

8424A ERROROUT
(# →)
Stores new error number and calls ERRJMP.

819F5 ERRORCLR
(→)
Stores zero as new error number.

81A18 ERRJMP
(→)
Invokes error handling sub-system.

81A04 GETEXITMSG
(→ $)
Gets EXITMSG (user defined error message).

81A09 EXITMSGSTO
($ →)
Stores $ as EXITMSG.

8212F DO#EXIT
(# →)
Stores new error number, does <REF>AtUserStack and then <REF>ERRJMP.

82139 DO$EXIT
($ →)
Stores string as EXITMSG, #70000 as error number, does <REF>AtUserStack and then <REF>ERRJMP .

81F9A ABORT
(→)
Does <REF>ERRORCLR and <REF>ERRJMP .

81A0E ERRSET
(→)
Sets new error trap.

81A13 ERRTRAP
(→)
Error trap marker. If no error happens, still removes all temporary environments created since ERRSET.

831E7 JstGetTHEMESG
(# → $)
Fetches message from message table. To get a message from a library, use the formula:
libnum*#100+msgnum.
--
<REF>TEXT:Libraries aka: JstGETTHEMSG

831DD GETTHEMESG
(# → $)
If #70000 then does <REF>GETEXITMSG, else does <REF>JstGetTHEMESG .
--
<REF>TEXT:Libraries

829EF ?GetMsg
(# → $msg)
(ob → ob)
If the argument is a bint, does JstGETTHEMSG to fetch a message. Other arguments are returned unchanged.
--
<REF>TEXT:Libraries

3.3.2 Error Generating Words

81A22	SETMEMERR	Error 001h Generates "Insufficient Memory" error.
81A27	SETROMPERR	Error 004h Generates "Undefined XLIB Name" error.
81A1D	SETLBERR	Error 006h Generates "Power Lost" error.
81CA7	SETSTACKERR	Error 201h Generates "Too Few Arguments" error.
81CA2	SETTYPEERR	Error 202h Generates "Bad Argument Type" error.
81C9D	SETSIZEERR	Error 203h Generates "Bad Argument Value" error.
81C98	SETNONEXTERR	Error 204h Generates "Undefined Name" error.

3.4 Conditionals

3.4.1 Boolean Flags

82F3F	%0<>	(% → flag) Can be used to change a user flag into a system flag.
03A81	TRUE	(→ T)
0BDC1	TrueTrue	(→ T T)
83F66	TrueFalse	(→ T F) aka: TRUEFALSE
03AC0	FALSE	(→ F)
83F7A	FalseTrue	(→ F T) aka: FALSETRUE
81FDB	FalseFalse	(→ F F)
82ED1	failed	(→ F T)
81522	DROPTRUE	(ob → T)
81527	DROPFALSE	(ob → F)
835E3	2DROPFALSE	(ob1 ob2 → F)
83926	XYZ>ZTRUE	(ob1 ob2 ob3 → ob3 T)
8182E	NOT	(flag → flag') Returns FALSE if the input is TRUE, and vice-versa.
81824	AND	(flag1 flag2 → flag) Returns TRUE if both flags are TRUE.
81833	OR	(flag1 flag2 → flag) Returns TRUE if either flag is TRUE.

8183D	XOR	(flag1 flag2 → flag) Returns TRUE if flags are different.
8401F	ORNOT	(flag1 flag2 → flag) Returns FALSE if either flag is TRUE.
836C4	NOTAND	(flag1 flag2 → flag) Returns TRUE if flag1 is TRUE and flag2 is FALSE.
83700	ROTAND	(flag1 ob flag2 → ob flag) Returns TRUE if either flag is TRUE.

3.4.2 General Tests

81842	EQ	(ob1 ob2 → flag) Returns TRUE if both objects are the same, i.e., they occupy the same physical space in memory. Only the addresses of the objects are tested.
84047	2DUPEQ	(ob1 ob2 → ob1 ob2 flag) Does 2DUP then EQ.
8405B	EQOR	(flag ob1 ob2 → flag') Does EQ then OR.
83AAC	EQOVER	(ob3 ob1 ob2 → ob3 flag ob3) Does EQ then OVER.
81847	EQUAL	(ob1 ob2 → flag) Returns TRUE if the objects are equal (but not necessarily the same), i.e., their prologs and contents are the same.
84033	EQUALNOT	(ob1 ob2 → flag) Returns TRUE if the objects are different.
8406F	EQUALOR	(flag ob1 ob2 → flag') Does EQUAL then OR.

3.4.3 True/False Tests

813D3	?SEMI	(T → :: ;) (F → :: <ob1> <rest> ;)
81414	NOT?SEMI	(T → :: <ob1> <rest> ;) (F → :: ;)
842EF	?SEMIDROP	(ob T → :: ob ;) (ob F → :: <ob1> <rest> ;)
8398A	?SWAP	(ob1 ob2 T → :: ob2 ob1 <ob1> <rest> ;) (ob1 ob2 F → :: ob1 ob2 <ob1> <rest> ;)
8380E	?SKIPSWAP	(ob1 ob2 T → :: ob1 ob2 <ob1> <rest> ;) (ob1 ob2 F → :: ob2 ob1 <ob1> <rest> ;)
839CB	?SWAPDROP	(ob1 ob2 T → :: ob1 <ob1> <rest> ;) (ob1 ob2 F → :: ob2 <ob1> <rest> ;)

839B2	NOT?SWAPDROP	(ob1 ob2 T → :: ob2 <ob1> <rest> ;) (ob1 ob2 F → :: ob1 <ob1> <rest> ;)
817A2	RPIT	(T ob → :: ob <ob1> <rest> ;) (F ob → :: <ob1> <rest> ;) ob is actually executed, and not pushed in the stack.
817A7	RPITE	(T ob1 ob2 → :: ob1 <ob1> <rest> ;) (F ob1 ob2 → ob2 <ob1> <rest> ;) ob1 or ob2 is actually executed, and not pushed in the stack.
813FB	COLARPITE	(T ob1 ob2 → :: ob1 ;) (F ob1 ob2 → :: ob2 ;) ob1 or ob2 is actually executed, and not pushed in the stack.
DE622	2'RCOLARPITE	Return to composite and ITE there.
81405	IT	(T → :: <ob1> <rest> ;) (F → :: <ob2> <rest> ;)
8173E	?SKIP	(T → :: <ob2> <rest> ;) (F → :: <ob1> <rest> ;) aka: NOT_IT
8140A	ITE	(T → :: <ob1> <ob3> <rest> ;) (F → :: <ob2> <rest> ;)
8140F	ITE_DROP	(ob T → :: <ob2> <rest> ;) (ob F → :: ob <ob1> <rest> ;)
8486C	ANDITE	(f1 f2 → :: <ob1> <ob3> <rest> ;) (f1 f2 → :: <ob2> <rest> ;)
813D8	case	(T → :: <ob1> ;) (F → :: <ob2> <rest> ;)
81419	NOTcase	(T → :: <ob2> <rest> ;) (F → :: <ob1> ;)
846F5	ANDcase	(f1 f2 → :: <ob1> ;) (f1 f2 → :: <ob2> <rest> ;)
847EA	ANDNOTcase	(f1 f2 → :: <ob1> ;) (f1 f2 → :: <ob2> <rest> ;)
81FD1	ORcase	(f1 f2 → :: <ob1> ;) (f1 f2 → :: <ob2> <rest> ;)
813EC	casedrop	(ob T → :: <ob1> ;) (ob F → :: ob <ob2> <rest> ;)
8142D	NOTcasedrop	(ob T → :: ob <ob2> <rest> ;) (ob F → :: <ob1> ;)
813E2	case2drop	(ob1 ob2 T → :: <ob1> ;) (ob1 ob2 F → :: ob1 ob2 <ob2> <rest> ;)
81423	NOTcase2drop	(ob1 ob2 T → :: ob1 ob2 <ob2> <rest> ;) (ob1 ob2 F → :: <ob1> ;)
813E7	caseDROP	(ob T → :: ;) (ob F → :: ob <ob1> <rest> ;)

81428	NOTcaseDROP	(ob T → :: ob <ob1> <rest> ;) (ob F → :: ;)
842A4	casedrptru	(ob T → T) (ob F → :: ob <ob1> <rest> ;) Note: should be called caseDRPTRU.
83FD9	casedrpfls	(ob T → F) (ob F → :: ob <ob1> <rest> ;) Note: should be called caseDRPFLS.
844F7	NOTcsdrpfls	(ob T → :: ob <ob1> <rest> ;) (ob F → F) Note: should be called NOTcaseDRPFLS.
813DD	case2DROP	(ob1 ob2 T → :: ;) (ob1 ob2 F → :: ob1 ob2 <ob1> <rest> ;)
8141E	NOTcase2DROP	(ob1 ob2 T → :: ob1 ob2 <ob1> <rest> ;) (ob1 ob2 F → :: ;)
83FF2	case2drpfls	(ob1 ob2 T → F) (ob1 ob2 F → :: ob1 ob2 <ob1> <rest> ;) Note: should be called case2DRPFLS.
83F52	caseTRUE	(T → T) (F → :: <ob1> <rest> ;)
842BD	NOTcaseTRUE	(T → :: <ob1> <rest> ;) (F → T)
8400B	caseFALSE	(T → F) (F → :: <ob1> <rest> ;)
842D6	NOTcaseFALSE	(T → :: <ob1> <rest> ;) (F → F)
81400	COLAcase	(T → :: <ob1> ;) (F → :: <ob2> <rest> ;) Drops the rest of current stream and executes case in the stream above.

3.4.4 Binary Integer Tests

83DDB	#=?SKIP	(#m #n → :: <ob2> <rest> ;) (#m #n → :: <ob1> <rest> ;)
83E08	#>?SKIP	(#m #n → :: <ob1> <rest> ;) (#m #n → :: <ob2> <rest> ;)
8369C	#=ITE	(#m #n → :: <ob1> <ob3> <rest> ;) (#m #n → :: <ob2> <rest> ;)
848A8	#<ITE	(#m #n → :: <ob1> <ob3> <rest> ;) (#m #n → :: <ob2> <rest> ;)
848BC	#>ITE	(#m #n → :: <ob2> <rest> ;) (#m #n → :: <ob1> <ob3> <rest> ;)
8136A	#=case	(#m #n → :: <ob1> ;) (#m #n → :: <ob2> <rest> ;)
81365	OVER#=case	(#m #n → :: #m <ob1> ;) (#m #n → :: #m <ob2> <rest> ;)

8136F	#=casedrop	(#m #n → :: <ob1> ;) (#m #n → :: #m <ob2> <rest> ;) Note: should be called OVER#=casedrop.
83FB6	#=casedrpfls	(#m #n → F) (#m #n → :: #m <ob1> <rest> ;) Note: should be called OVER#=caseDRPFLS.
84745	#<>case	(#m #n → :: <ob2> <rest> ;) (#m #n → :: <ob1> ;)
8471D	#<case	(#m #n → :: <ob1> ;) (#m #n → :: <ob2> <rest> ;)
84772	#>case	(#m #n → :: <ob2> <rest> ;) (#m #n → :: <ob1> ;)
815B3	#0=?SEMI	(#0 → :: ;) (# → :: <ob1> <rest> ;)
83DA9	#0=?SKIP	(#0 → :: <ob2> <rest> ;) (# → :: <ob1> <rest> ;)
84894	#0=ITE	(#0 → :: <ob1> <ob3> <rest> ;) (# → :: <ob2> <rest>)
84853	DUP#0=IT	(#0 → :: #0 <ob1> <rest> ;) (# → :: # <ob2> <rest> ;)
848D0	DUP#0=ITE	(#0 → :: #0 <ob1> <ob3> <rest> ;) (# → :: # <ob2> <rest> ;)
81360	#0=case	(#0 → :: <ob1> ;) (# → :: <ob2> <rest> ;)
81FAE	DUP#0=case	(#0 → :: #0 <ob1> ;) (# → :: # <ob2> <rest> ;)
81388	DUP#0=csedrp	(#0 → :: <ob1> ;) (# → :: # <ob2> <rest> ;)
846C8	DUP#0=csDROP	(#0 → :: ;) (# → :: # <ob1> <rest> ;)
84731	#1=case	(#1 → :: <ob1> ;) (# → :: <ob2> <rest> ;)
83DC2	#1=?SKIP	(#1 → :: <ob2> <rest> ;) (# → :: <ob1> <rest> ;)
84759	#>2case	(#0/#1/#2 → :: <ob2> <rest> ;) (# → :: <ob1> ;)
82BCF	?CaseKeyDef	(# #' → :: ' ob1 T ;) (# #' → :: <ob2> <rest> ;) Compares two bints. If equal, quotes the next object from the runsream and returns it along with TRUE.
82BED	?CaseRomptr@	(# #' → ob T) (# #' → F) (# #' → :: <ob2> <rest> ;) Compares two bints. If equal, tries to resolve the rompointer which must be the next object in the runstream. The ROMPTR@ pushes TRUE when successful, so this entry can be used directly for key handlers.

3.4.5 Real and Complex Number Tests

84786	j%0=case	(%0 → :: <ob1> ;) (ob → :: <ob2> <rest> ;)

3.4.6 General Object Tests

8483A	EQIT	(ob1 ob1 → :: <ob1> <rest> ;) (ob1 ob2 → :: <ob2> <rest> ;)
84880	EQITE	(ob1 ob1 → :: <ob1> <ob3> <rest> ;) (ob1 ob2 → :: <ob2> <rest> ;)
846E1	jEQcase	(ob1 ob1 → :: <ob1> ;) (ob1 ob2 → :: <ob2> <rest> ;)
DE46C	EQcase	(ob1 ob1 → :: ob1 <ob1> ;) (ob1 ob2 → :: ob1 <ob2> <rest> ;) Note: Should be called OVEREQcase.
8138D	EQcasedrop	(ob1 ob1 → :: <ob1> ;) (ob1 ob2 → :: ob1 <ob2> <rest> ;) Note: should be called OVEREQcasedrop.
84709	EQUALcase	(ob1 ob1 → :: <ob1> ;) (ob1 ob2 → :: <ob2> <rest> ;)
847FE	EQUALNOTcase	(ob1 ob1 → :: <ob2> <rest> ;) (ob1 ob2 → :: <ob1> ;)
846AF	EQUALcasedrp	(ob ob1 ob2 → :: <ob1> ;) (ob ob1 ob2 → :: ob <ob2> <rest> ;)
84812	dIDNTNcase	(id → :: id <ob2> <rest> ;) (ob → :: ob <ob1> ;)
8479A	REALcase	(% → :: <ob1> ;) (ob → :: <ob2> <rest> ;)
84826	dREALNcase	(% → :: % <ob2> <rest> ;) (ob → :: ob <ob1> ;)
847AE	dARRYcase	([] → :: [] <ob1> ;) (ob → :: ob <ob2> <rest> ;)
847C2	dLISTcase	({} → :: {} ob1 ;) (ob → :: ob <ob2> <rest> ;)

3.4.7 Miscellaneous

848E4	UserITE	(#set → :: <ob1> <ob3> <rest> ;) (#clr → :: <ob2> <rest> ;)
848F8	SysITE	(#set → :: <ob1> <ob3> <rest> ;) (#clr → :: <ob2> <rest> ;)

845F6	caseDoBadKey	(T → :: DoBadKey ;) (F → :: <ob1> <rest> ;) aka: caseDEADKEY
845DD	caseDrpBadKy	(ob T → :: DoBadKey ;) (ob F → :: ob <ob1> <rest> ;)
83BD8	caseERRJMP	(T → :: ERRJMP ;) (F → :: <ob> <rest> ;)
84510	caseSIZEERR	(T → :: SIZEERR ;) (F → :: <ob> <rest> ;)
84524	NcaseSIZEERR	(T → :: <ob> <rest> ;) (F → :: SIZEERR ;)
84551	NcaseTYPEERR	(T → :: <ob1> <rest> ;) (F → :: TYPEERR ;)
847D6	EditExstCase	(→ :: <ob1> <rest> ;) (→ :: <rest> ;) Tests if there is an edit line active.

3.5 Runstream Control

81789	NOP	(→) Does nothing.
8172A	'R	(→ ob) Pushes next object in return stack (i.e., the first object in the composite above this one) to the stack (skipping it). If top return stack is empty (contains SEMI), a null secondary is pushed and the pointer is not advanced.
8172F	'REVAL	(→ ?) Does <REF>'R then <REF>EVAL.
843E9	'R'R	(→ ob1 ob2) Does <REF>'R twice.
8148C	ticR	(→ ob T) (→ F) Pushes next object in return stack to stack and TRUE, of just FALSE if the top return stack body is empty. In this case, it is dropped.
84407	'RRDROP	(→ ob) Does <REF>'R , then <REF>RDROP.
81734	>R	(:: →) Pushes :: to top of return stack (skips prolog, i.e., the composite will be executed automatically).
8178E	R>	(→ ::) Creates and pops a secondary from top return stack body to stack.

81793	R@	(→ ::) Like <REF>R>, but the return stack is not popped.
81757	EVAL	(ob →) Evaluates object.
81CBB	COMPEVAL	(comp →) EVAL just pushes a list back, this one executes it.
81B21	RDUP	(→) Duplicates top return stack level.
81798	RDROP	(→) Pops the return stack.
813CE	2RDROP	(→) Pops two return stack levels.
81469	3RDROP	(→) Pops three return stack levels.
83D68	DROPRDROP	(ob →) Does DROP then <REF>RDROP .
81432	RDROPCOLA	(→) Does <REF>RDROP then <REF>COLA .
81437	RSWAP	(→) Swap in the return stack.
81B26	RROLL	(#n →) Rolls nth return stack level to top of return stack.
84272	RSKIP	(→) Skips first object in the return stack (i.e., the first object in the composite above this one).
0312B	SEMI	(→) DROP the rest of the current stream.

3.5.1 Quoting Objects

81725	’	(→ nob (nextob)) Pushes next object in the stream to the stack (skipping it).
84330	DUP’	(ob → ob nob) Does DUP then ’.
84358	DROP’	(ob → nob) Does DROP then ’.
84344	SWAP’	(ob1 ob2 → ob2 ob1 nob) Does SWAP then ’.
8436C	OVER’	(ob1 ob2 → ob1 ob2 ob1 nob) Does OVER then ’.
84380	STO’	(ob id/lam → nob) Does STO then ’.
84394	TRUE’	(→ T nob) Pushes TRUE and the next object to the stack.

843C1	FALSE’	(→ F nob) Pushes FALSE and the next object to the stack.
843A8	ONEFALSE’	(→ #1 F nob) Pushes ONE, FALSE and the next object to the stack.
843D5	#1+’	(# → #+1 nob) Does #1+ then ’.
83D2C	’NOP	(→ NOP) Pushes NOP to the stack.
83BC4	’ERRJMP	(→ ERRJMP) Pushes ERRJMP to the stack.
81851	’DROPFALSE	(→ DROPFALSE) Pushes DROPFALSE to the stack.
82BBB	’DoBadKey	(→ DoBadKey) Pushes DoBadKey to the stack.
3C77E	’DoBadKeyT	(→ DoBadKey T) Pushes <REF>DoBadKey and TRUE to the stack.
84579	’xDER	(→ xDER) Pushes xDER (User word ∂) to the stack.
82E4F	’Rapndit	(meta ob1...ob4 → meta&ob ob1...ob4) Takes ob from runstream and appends it to the meta starting in level 5.
84461	’xDEREQ	(ob → flag) Is ob eq to user command xDER?

3.5.2 Skipping Objects

8174D	COLA	Evals next obj and drops rest of this stream.
84420	ONECOLA	Does ONE, then COLA.
83D81	SWAPCOLA	Does SWAP, then COLA.
83D95	XYZ>ZCOLA	Does UNROT2DROP, then COLA.
813F6	COLA_EVAL	Returns and evals first obj in previous stream.
813F1	COLACOLA	Drops rest of current stream does COLA in the above one.
817AC	SKIP	Skips 1 obj in the runstream.
83E21	COLASKIP	Drops rest of current stream and skips one obj in above stream.

3.6 Loops

3.6.1 Indefinite Loops

81748 BEGIN (→)
Pushes interpreter pointer into the return stack.

81743 AGAIN (→)
Sets the interpreter pointer to the topmost value in the return stack, without popping it.

8179D REPEAT (→)
Sets the interpreter pointer to the topmost value in the return stack, without popping it.

817B1 UNTIL (flag →)
If FALSE then <REF>AGAIN, otherwise <REF>RDROP .

83E35 NOT_UNTIL (flag →)
NOT then <REF>UNTIL .

83647 #0=UNTIL (# → #)
Actually, should be called DUP#0=UNTIL.

817B6 WHILE (flag →)
If TRUE does nothing, otherwise <REF>RDROP then <REF>2SKIP .

83E4E NOT_WHILE (flag →)
NOT then <REF>WHILE .

83E67 DUP#0<>WHILE (# →)
Try to guess what it does.

3.6.2 Definite Loops

81752 DO (#stop #start →)

0744F ZERO_DO (#stop →)

83EEE DUP#0_DO (#stop → #stop)

0745A ONE_DO (#stop →)

07467 #1+_ONE_DO (#stop →)

83F07 toLEN_DO ({} → {})
From ONE to #elements.

81784 LOOP (→)

81739 +LOOP (# →)
Increments index by specified number.

83ED5 DROPLOOP (ob →)

83EBC SWAPLOOP (ob1 ob2 → ob2 ob1)

8175C INDEX@ (→ #)
Recalls topmost loop counter value.

83E80 DUPINDEX@ (ob → ob #)

83E94 SWAPINDEX@ (ob1 ob2 → ob2 ob1 #)

83EA8 OVERINDEX@ (ob1 ob2 → ob1 ob2 ob1 #)

841E6 INDEX@#- (# → #’)

81761	INDEXSTO	(# →) Stores new topmost loop counter value.
81766	ISTOP@	(→ #) Recalls topmost loop stop value.
8176B	ISTOPSTO	(# →) Stores new topmost loop stop value.
81770	JINDEX@	(→ #) Recalls second topmost loop counter value.
81775	JINDEXSTO	(# →) Stores new second topmost loop counter value.
8177A	JSTOP@	(→ #) Recalls second topmost loop stop value.
8177F	JSTOPSTO	(# →) Stores new second topmost loop stop value.
84A1A	ExitAtLOOP	(→) Does not exit loop immediately. Just stores zero as the stop value, so all objects until the next LOOP will be evaluated. aka: ZEROISTOPSTO

3.7 Memory Operations

3.7.1 Recalling, Storing and Purging

81C11	@	(id/lam → ob T) (id/lam → F) Basic recalling function.
83674	DUP@	(id/lam → id/lam ob T) (id/lam → id/lam F) Does DUP then <REF>@.
81C70	STO	(ob id/lam →) For ids this assumes ob is not pco. If replacing some object, that object is copied to TEMPOB and pointers are updated. For lams: Errors if lam is unbound.
81C61	REPLACE	(newob oldob → newob) Replaces oldob (in memory) with newob.
81C4D	PURGE	(id →) Purges variable. Does no type check first.
81C2F	CREATE	(ob id →) Creates a variable in the current directory. Errors if id is or contains current directory. Assumes id is not a pco.
84448	'LAMLNAMESTO	(ob →) STO to LAM LAMLNAME.

3.7.2 Directories

81C3E	MAKERRP	(#libnum → rrp) Creates an empty directory.
82D2D	CREATEDIR	(id →) Creates an empty directory. Calls <REF>?PURGE_HERE first to delete the original.
81C57	LASTRAM-WORD	(rrp → ob T) (rrp → F) Recalls first object in directory.
81C5C	PREVRAM-WORD	(ob → ob' T) (ob → F) Recalls next object in directory.
08452	RAM-WORDNAME	(ob → id) Recalls name of object in current directory.
81C7F	SYSRRP?	(rrp → flag) Is rrp HOME?
81C2A	CONTEXT@	(→ rrp) Recalls current directory.
81C25	CONTEXT!	(rrp →) Sets new current directory.
81C7F	SYSRRP?	(rrp → flag) Is rrp HOME?
81C34	HOMEDIR	(→) Sets HOME as current directory. aka: SYSCONTEXT

3.7.3 Temporary Memory

81A4A	TOTEMPOB	(ob → ob') Copies object to TEMPOB and returns pointer to the new copy.
836D8	TOTEMPSWAP	(ob1 ob2 → ob2' ob1) Does TOTEMPOB then SWAP.
82D23	CKREF	(ob → ob') If object is in TEMPOB, is not embedded in a composite and not referenced, does nothing. Else copies it to TEMPOB and returns the copy.
84989	SWAPCKREF	(ob1 ob2 → ob2 ob1') Does SWAP then <REF>CKREF.
81A4F	INTEMNOTREF?	(ob → ob flag) If the object is in TEMPOB area, is not embedded in a composite and is not referenced, returns the object and TRUE, otherwise returns the object and FALSE.

0220E8 ~INTEMPOB? (ob → ob flag)

3.8 Time and Alarms

81A95 SLOW (→)
15 millisecond delay.

81A9A VERYSLOW (→)
300 millisecond delay.

81A9F VERYVERYSLOW (→)
3 second delay.

82C0B dowait (%secs →)
Waits specified number of seconds.

82F85 %>HMS (% → %hms)
Converts from decimal to H.MMSS format.

82F7B %HMS> (%hms → %)
Converts from H.MMSS format to decimal.

82F67 %HMS+ (%hms1 %hms2 → %hms)
Adds time in hms format.

82F71 %HMS- (%hms1 %hms2 → %hms)
Subtracts time in hms format.

831D3 CLKTICKS (→ hxs)
Returns tick count. aka: SysTime

3.9 System Functions

3.9.1 User and System Flags

81A5E SetSysFlag (# →)
Sets the system flag with number #. <REF>TEXT:Flags

81A54 ClrSysFlag (# →)
Clears the system flag with number #. <REF>TEXT:Flags

81A68 TestSysFlag (# → flag)
Returns TRUE if system flag is set. <REF>TEXT:Flags

81A63 SetUserFlag (# →)
Set the user flag with number #. <REF>TEXT:Flags

81A59 ClrUserFlag (# →)
Clear the user flag with number #. <REF>TEXT:Flags

81A6D TestUserFlag (# → flag)
Returns TRUE if user flag is set. <REF>TEXT:Flags

3.9.2 General Functions

81B1C	setbeep	(#ms #Hz →) Also beeps.
82D05	TurnOff	(→) Internal OFF.
81A40	GARBAGE	(→) Forces garbage collection.
81A45	MEM	(→ #) Returns amount of free memory in nibbles. Does not do garbage collection. (The user word does.)
819C3	OSIZE	(ob → #) Returns object size in nibbles. Forces garbage collection.
819BE	OCRC	(ob → #nib hxs) Returns size in nibbles and checksum as hxs.
81BA8	INHARDROM?	(ob → ob flag) Is object address < #80000h?
82D19	CHANGETYPE	(ob #prolog → ob') Changes prolog of object, does TOTEMPOB.

3.10 Kermit

82B57	DOFINISH	(→) Internal FINISH.
10C72	CLOSEUART	(→) Internal CLOSEIO.
82B9D	OpenIO	
82ADF	docr	
81D88	OpenUart?Clr	
82BB1	PRINT	
81A81	SetEcma94	
117CD	AllowPrlcdCl	
81D83	UARTBUFLEN	
82BA7	PUTSERIAL	
81D7E	GETSERIAL	

4 Input and Output

4.1 Checking for Arguments

4.1.1 Number and Type of Arguments

45611 CK0 (→)
Saves current command to LASTCKCMD. Marks stack below level 1 to STACKMARK.

81CD9 CK&DISPATCH0 (→)
Dispatches on stack argument. Does not convert ZINTs to REAL.
--
<REF>CK&DISPATCH1 <REF>CK&DISPATCH2 <REF>TEXT:Dispatch_Types

81CDE CK&DISPATCH1 (→)
Dispatches on stack arguments, stripping tags and converting ZINTS to REALS (HP49 only) if necessary.
--
<REF>CK&DISPATCH0 <REF>CK&DISPATCH2 <REF>TEXT:Dispatch_Types

81CE3 CK&DISPATCH2 (→)
Equivalent to <REF>CK&DISPATCH1.
--
<REF>CK&DISPATCH0
<REF>TEXT:Dispatch_Types

45AC1 CK1&Dispatch (→)
Combines <REF>CK1 with <REF>CK&DISPATCH1.
--
<REF>TEXT:Dispatch_Types

45AD2 CK2&Dispatch (→)
Combines <REF>CK2 with <REF>CK&DISPATCH1.
--
<REF>TEXT:Dispatch_Types

45AE3 CK3&Dispatch (→)
Combines <REF>CK3 with <REF>CK&DISPATCH1.
--
<REF>TEXT:Dispatch_Types

45AF4	CK4&Dispatch	(→) Combines <REF>CK4 with <REF>CK&DISPATCH1. -- <REF>TEXT:Dispatch_Types
45B05	CK5&Dispatch	(→) Combines <REF>CK5 with <REF>CK&DISPATCH1. -- <REF>TEXT:Dispatch_Types
81B12	0LASTOWDOB!	(→) Clears command save by last CK<n> command. <REF>CK0 aka: 0LASTOWDOB!, 0LastRomWrd!

4.1.2 Type Checking

84538	CKREAL	(% → %) (Z → %) Checks for real. If a ZINT, convert to real. Else SETTYPEERR.
8184C	TYPE	(ob → #prolog) Returns address of prolog of object.
816FD	TYPEREAL?	(ob → flag)
83593	DUPTYPEREAL?	(ob → ob flag) aka: DTYPEREAL?
816CB	TYPECMP?	(ob → flag)
83539	DUPTYPECMP?	(ob → ob flag)
816D5	TYPECSTR?	(ob → flag)
8354D	DUPTYPECSTR?	(ob → ob flag) aka: DTYPECSTR?
8351B	DUPTYPEARRY?	(ob → ob flag) aka: DTYPEARRY?
816B7	TYPEARRY?	(ob → flag ???)
816F8	TYPERARRY?	(ob → flag)
816C1	TYPECARRY?	(ob → flag)
816F3	TYPELIST?	(ob → flag)
83589	DUPTYPELIST?	(ob → ob flag) aka: DTYPELIST?
816E9	TYPEIDNT?	(ob → flag)
83575	DUPTYPEIDNT?	(ob → ob flag)
816EE	TYPELAM?	(ob → flag)
8357F	DUPTYPELAM?	(ob → ob flag)
8170C	TYPESYMB?	(ob → flag)

835B1	DUPTYPESYMB?	(ob → ob flag) Is ob a unit object?
816E4	TYPEHSTR?	(ob → flag)
8356B	DUPTYPEHSTR?	(ob → ob flag)
816DF	TYPEGROB?	(ob → flag)
83561	DUPTYPEGROB?	(ob → ob flag)
835BB	DUPTYPETAG?	(ob → ob flag)
816DA	TYPEEXT?	(ob → flag)
83557	DUPTYPEEXT?	(ob → ob flag)
81702	TYPEROMP?	(ob → flag)
8359D	DUPTYPEROMP?	(ob → ob flag)
816BC	TYPEBINT?	(ob → flag)
83525	DUPTYPEBINT?	(ob → ob flag)
81707	TYPERRP?	(ob → flag)
835A7	DUPTYPERRP?	(ob → ob flag)
816C6	TYPECHAR?	(ob → flag)
8352F	DUPTYPECHAR?	(ob → ob flag)
816D0	TYPECOL?	(ob → flag) Is on a secondary?
83543	DUPTYPECOL?	(ob → ob flag) Is ob a secondary? aka: DTYPECOL?

4.2 Keyboard Control

4.2.1 Converting Keycodes

82BD9	Ck&DecKeyLoc	(%rc.p → #kc #p) Converts from user key representation format to system. Does handle shift-hold keys.
82BE3	CodePl>%rc.p	(#kc #p → %rc.p) Converts from system key representation format to user. Does handle shift-hold keys.
82C6F	ModifierKey?	(#kc #pl → flag) Is the key any of the three modifiers right-shift, left-shift, or alpha?

4.2.2 Waiting for Keys

81A86	FLUSHKEYS	(→) Flushes the key buffer. aka: FLUSH

81A31	CHECKKEY	(→ #kc T) (→ F) Returns next key in the key buffer (if there is one), but does not pop it. Does handle shift-hold keys. -- <REF>TEXT:Keycodes
81A36	GETTOUCH	(→ #kc T) (→ F) Pops next key from key buffer (if there is one). Does handle shift-hold keys. -- <REF>TEXT:Keycodes
82C3D	GETKEY	(→ #kc flag) Get a single keypress from the keybuffer, waits if necessary. The key is returned along with TRUE. If an exception happens, returns FALSE. The exception is not handled. Does handle shift-hold keys. -- <REF>TEXT:Keycodes
81A3B	REPKEY?	(#kc → flag) Returns TRUE if the key is being pressed. -- <REF>TEXT:Keycodes
82CDD	REPEATER	(→) Takes two objects from the runstream, a BINT and a program. The BINT must represent a keycode. The program is evaluated at least once, and then again and again as long as the specified key is being pressed. -- <REF>TEXT:Keycodes
82CE7	REPEATERCH	(→) Same as REPEATER, but slower, so more appropriate for scrolling and cursor motions. -- <REF>TEXT:Keycodes
82C5B	KEYINBUFFER?	(→ flag) Returns TRUE if there is at least a key in the key buffer.
82D0F	WaitForKey	(→ #kc #flag) Returns next full key press. Does *not* handle shift-hold keys. -- <REF>TEXT:Keycodes

4.2.3 The ATTN Flag

`81A90` `ATTN?` `( → flag )`
Returns `TRUE` if ⟨CANCEL⟩ has been pressed.

`82BC5` `?ATTNQUIT` `( → )`
If ⟨CANCEL⟩ has been pressed, ABORTs program. aka: `?ATTN_QUIT`

`81A2C` `ATTNFLGCLR` `( → )`
Clears ⟨CANCEL⟩ key counter. Does not affect the key buffer.

4.2.4 Bad Keys

`3C635` `DoBadKey` `( → )`
Beeps.

4.2.5 User Keys

`82C65` `Key>StdKeyOb` `( #kc #pl → ob )`
Recalls the standard assignment of the key. This is the assignment which is active when USER mode is of.

4.3 The Menu

4.3.1 Menu Properties

`82C79` `NoExitAction` `( → )`
Sets `NOP` as ExitAction. Mostly used to avoid that the menu is saved as the previous menu when a new Menu gets installed.

4.3.2 Building Menus

`3CC36` `TakeOver` `( → )`
Override the default menu key executer. If this is the first entry in a program, the program can be used in edit mode. When the first in a program in the label slot of a menu key, the program is evaluated to get the label object (most likely a grob).

82C47 InitMenu (menu →)
menu is {} or :: settings {} ; Settings override the default settings installed by InitMenu.

4.3.3 Menu Display

88527 ?DispMenu (→)
Redisplay the menu now if no key is waiting in the buffer. Even better is this:
:: DA3OK?NOTIT ?DispMenu ;

82C29 DispMenu.1 (→)
Displays the menu immediately.

82C1F DispMenu (→)
:: DispMenu.1 SetDAsValid ;

4.4 InputLine and Inputforms

4.4.1 Inputform

8273D DoInputForm (l1..ln f1..fm #n #m msg $ → ob1..obn T)
(l1..ln f1..fm #n #m msg $ → F)

```
l = $ #x #y
f = msg #x #y #w #h #type legal
dec $hlp ChDat ChDec res init
```

Starts an input form using the old engine.

4.5 The Browser Engines

4.5.1 The HP48 Browser Engine

4.6 The Parametrized Outer Loop (POL)

82CD3 ParOuterLoop (Disp Keys NonAppKeys? DoStdKeys? menu #row suspendOK? ExitCond AppErr →)

82C8D POLSaveUI (Disp Keys NonAppKeys? DoStdKeys? menu #row suspendOK? ExitCond AppErr →)
Saves current UI to LAMSavedUI.

82CB5 POLSetUI <see>ParOuterLoop
Sets new UI, same arguments as to ParOuterLoop.

82C97 POLKeyUI (→)
Displays, reads and evaluates keys according to set UI.

82CAB POLRestoreUI (→)
Restores saved UI from LAMSavedUI.

82CA1 POLResUI&Err (→)
Restores saved UI and executes ERRJMP.

4.7 The Display

4.7.1 Display Organization

81ACC TOADISP (→)
Sets the text display as the active.

81AD1 TOGDISP (→)
Sets the graphic display as the active.

81860 ABUFF (→ textgrob)
Returns the text grob to the stack.

81892 GBUFF (→ graphgrob)
Returns the graphic grob to the stack. The HP49 extable address for ExitAction! is the same, but this must be a bug.

818A1 HARDBUFF (→ dispgrob)
Returns the current grob to the stack.

818A6 HARDBUFF2 (→ menugrob)
Returns the menu grob to the stack.

33905 HARDHEIGHT (→ #height)
Returns the height of HARDBUFF.

4.7.2 Preparing the Display

8230F RECLAIMDISP (→)
Activates the text grob, clears it and sets the default size.

824B3 MENUOFF? (→ flag)
Returns TRUE if the menu grob is off.

8249F TURNMENUOFF (→)
Turns off menu display, enlarges ABUFF to fill screen.

824A9 TURNMENUON (→)
Turns menu grob on.

81AC2 HEIGHTENGROB (grob #rows →)
Heightens graph or text grob.

81AC7	KILLGDISP	(→)
		Clears graph display by setting it to NULLGROB. See DOERASE.
821BB	DOERASE	(→)
		Erases the graphics display grob without changing its size.

4.7.3 Controlling Display Refresh

8208F	ClrDA1OK	(→)
82099	ClrDA2aOK	(→)
820A3	ClrDA2bOK	(→)
820AD	ClrDA2OK	(→)
820B7	ClrDA3OK	(→)
81FEF	DA1OK?	(→ flag)
81FF9	DA2aOK?	(→ flag)
82003	DA2bOK?	(→ flag)
8200D	DA2OK?	(→ flag)
82017	DA3OK?	(→ flag)
82021	DAsOK?	(→ flag)
8202B	DA2aLess1OK?	(→ flag)
26B24	DA3OK?NOTIT	(→)
		Does DA3OK?, NOT then IT.
82067	SetDA1Temp	(→)
82071	SetDA2aTemp	(→)
8207B	SetDA3Temp	(→)
82107	SetDA12Temp	(→)
82111	SetDAsTemp	(→)
82035	SetDA1Valid	(→)
8203F	SetDA2aValid	(→)
82049	SetDA2bValid	(→)
82053	SetDA2Valid	(→)
8205D	SetDA3Valid	(→)
81B9E	SetDA3ValidF	(→)
81B71	SetDA1Bad	(→)
81B4E	ClrDA1Bad	(→)
81B62	DA1Bad?	(→ flag)
81B80	SetDA2aBad	(→)
81B53	ClrDA2aBad	(→)
81B67	DA2aBad?	(→ flag)

81B8A	SetDA2bBad	(→)
81B58	ClrDA2bBad	(→)
81B94	SetDA3Bad	(→)
81B7B	SetDA1NoCh	(→)
81B85	SetDA2aNoCh	(→)
81B8F	SetDA2bNoCh	(→)
81B5D	ClrDA2bNoCh	(→)
81B6C	DA2bNoCh?	(→ flag)
820C1	SetDA2NoCh	(→)
820CB	SetDA12NoCh	(→)
81B99	SetDA3NoCh	(→)
820DF	SetDA13NoCh	(→)
820D5	SetDA23NoCh	(→)
820E9	SetDA12a3NCh	(→) aka: SetDA12a3NoCh
820F3	SetDAsNoCh	(→)
82085	SetDA2aEcho	(→)
81B76	SetDA1IsStat	(→)
81BA3	SetNoRollDA2	(→)

4.7.4 Clearing the Display

82157	BLANKIT	(#startrow #rows →) Clears #rows from HARDBUFF, starting at #startrow.
81A8B	CLCD10	(→) Clears status and stack areas.
821B1	DOCLLCD	(→) Like user word <REF>CLLCD.

4.7.5 Annunciator and Modes Control

82A03	LockAlpha	(→) Sets alpha mode, annunciators, etc.
82A0D	UnLockAlpha	(→) Clears alpha mode, annunciators, etc.

4.7.6 Window Coordinates

81AD6 WINDOWCORNER (→ #y #x)
Gets coordinates of corner of window. Note the order of #x and #y.

81AEF WINDOWXY (#y #x →)
Sets corner coordinates. The name really should be WINDOWYX

4.7.7 Scrolling the Display

81AEA WINDOWUP (→)
Moves display one pixel up.

81ADB WINDOWDOWN (→)
Moves display one pixel down.

81AE0 WINDOWLEFT (→)
Moves display one pixel left.

81AE5 WINDOWRIGHT (→)
Moves display one pixel right.

82341 SCROLLUP (→)
Moves display one pixel up, checks for corresponding key being pressed.

82323 SCROLLDOWN (→)
Moves display one pixel down, checks for corresponding key being pressed.

8232D SCROLLLEFT (→)
Moves display one pixel left, checks for corresponding key being pressed.

82337 SCROLLRIGHT (→)
Moves display one pixel right, checks for corresponding key being pressed.

82297 JUMPTOP (→)
Jumps to top of display.

8228D JUMPBOT (→)
Jumps to bottom of display.

822A1 JUMPLEFT (→)
Jumps to left of display.

822AB JUMPRIGHT (→)
Jumps to right of display.

824E5 WINDOWTOP? (→ flag)
Is window at the top?

824C7 WINDOWBOT? (→ flag)
Is window at the bottom?

824D1 WINDOWLEFT? (→ flag)
Is window at the left?

824DB WINDOWRIGHT? (→ flag)
Is window at the right?

4.7.8 Displaying Text

821CF	DODISP	(ob %row →) Displays any object in specified row.
8186A	DISPROW1	($ →) aka: DISP@01, BIGDISPROW1
8186F	DISPROW2	($ →) aka: DISP@09, BIGDISPROW2
81874	DISPROW3	($ →) aka: DISP@17, BIGDISPROW3
81879	DISPROW4	($ →) aka: DISP@25, BIGDISPROW4
8187E	DISPROW5	($ →)
81883	DISPROW6	($ →)
81888	DISPROW7	($ →)
8188D	DISPROW8	($ →) May not be possible depending on the size of the font and whether the menu is on or off.
81865	DISPN	($ #row →) aka: BIGDISPN

4.7.9 Messages and Boxes

4.8 Graphics

4.8.1 Built-in Grobs

4.8.2 Dimensions

81BAD	GROBDIM	(grob → #height #width)
8460F	GROBDIMw	(grob → #width)
8217F	CKGROBFITS	(g1 g2 #n #m → g1 g2' #n #m) Shrinks g2 if it does not fit in g1.
69E2E	CHECKHEIGHT	(grob #height →) Forces grob (ABUFF/GBUFF) to be at least 64 rows high.

4.8.3 Grob Handling

81897	GROB!	(grob1 grob2 #x #y →) Stores grob1 into grob2. Bang type.

82251	GROB+#	(flag grob1 grob2 #x #y → grob') Inserts grob2 into the specified position of grob1, using OR (if flag is TRUE) or XOR (if flag is FALSE). Does all necessary checks first.
8189C	GROB!ZERO	(grob #x1 #y1 #x2 #y2 → grob') Blanks a rectangular region of the grob. Bang type.
84290	GROB!ZERODRP	(grob #x1 #y1 #x2 #y2 →) Blanks a rectangular region of the grob. Probably only useful if grob is the text or graphics grob (see section on display-organization). Bang type.
818B5	SUBGROB	(grob #x1 #y1 #x2 #y2 → grob') Returns specified portion of grob.
824EF	XYGROBDISP	(#x #y grob →) Stores grob in HARDBUFF with upper left corner at (#x,#y). HARDBUFF is expanded if necessary.
82265	GROB>GDISP	(grob →) Stores new graph grob.
818B0	MAKEGROB	(#height #width → grob) Creates a blank grob.
818AB	INVGROB	(grob → grob') Inverts grob data bits. Bang type.
81AFE	PIXON	(#x #y →) Sets pixel in text grob.
81AF4	PIXOFF	(#x #y →) Clears pixel in text grob.
81B03	PIXON?	(#x #y → flag) Is pixel in text grob on?
81B08	PIXON3	(#x #y →) Sets pixel in graph grob.
81AF9	PIXOFF3	(#x #y →) Clears pixel in graph grob.
81B0D	PIXON?3	(#x #y → flag) Is pixel in graph grob on?
81BBC	LINEON	(#x1 #y1 #x2 #y2 →) Draws a line in text grob.
81BB2	LINEOFF	(#x1 #y1 #x2 #y2 →) Clears a line in text grob.
81BC6	TOGLINE	(#x1 #y1 #x2 #y2 →) Toggles a line in text grob.
81BC1	LINEON3	(#x1 #y1 #x2 #y2 →) Draws a line in graph grob.
81BB7	LINEOFF3	(#x1 #y1 #x2 #y2 →) Clears a line in graph grob.

81BCB TOGLINE3 (#x1 #y1 #x2 #y2 →)
Toggles a line in graph grob.

821D9 DOLCD> (→ grob)
Returns current display.

4.8.4 Creating Menu Label Grobs

822D3 MakeStdLabel ($ → grob)
Makes standard menu label.

822B5 MakeBoxLabel ($ → grob)
Makes label with a box.

822BF MakeDirLabel ($ → grob)
Makes directory label.

822C9 MakeInvLabel ($ → grob)
Makes inverse label.

82C15 Box/StdLabel ($ flag → grob)
If TRUE makes box label, otherwise makes standard label.

8235F Std/BoxLabel ($ flag → grob)
If TRUE makes standard label, otherwise makes box label.

4.8.5 Converting Strings to Grobs

818BA $>BIGGROB ($ → grob)
Makes grob of the string using the large font (5x9).

818BF $>GROB ($ → grob)
Makes grob of the string using the system font. Linefeed does *not* make new line.

818C4 $>grob ($ → grob)
Makes grob of the string using the minifont. Linefeed does *not* make new line.

4.9 Plotting

82161 CHECKPICT (→)
Checks size of GBUFF. If it is smaller than 131x64 sets GBUFF back to its default size (131x64).

82229 GETXMIN (→ %)
Recalls XMIN from the PPAR list if existent. If not, the default PPAR is created in the current directory.
--
<REF>TEXT:Reserved|PPAR

822E7 PUTXMIN
(% →)
Sets a new value for XMIN. PPAR is created if necessary.
--
<REF>TEXT:Reserved|PPAR

8223D GETXMAX
(→ %)
Recalls XMAX from the PPAR list if existent. If not, the default PPAR is created in the current directory.
--
<REF>TEXT:Reserved|PPAR

822FB PUTXMAX
(% →)
Sets a new value for XMAX. PPAR is created if necessary.
--
<REF>TEXT:Reserved|PPAR

82233 GETYMIN
(→ %)
Recalls YMIN from the PPAR list if existent. If not, the default PPAR is created in the current directory.
--
<REF>TEXT:Reserved|PPAR

822F1 PUTYMIN
(% →)
Sets a new value for YMIN. PPAR is created if necessary.
--
<REF>TEXT:Reserved|PPAR

82247 GETYMAX
(→ %)
Recalls YMAX from the PPAR list if existent. If not, the default PPAR is created in the current directory.
--
<REF>TEXT:Reserved|PPAR

82305 PUTYMAX
(% →)
Sets a new value for YMAX. PPAR is created if necessary.
--
<REF>TEXT:Reserved|PPAR

69347 GETYPOS

8221F GDISPCENTER
(→)
Moves to center of graphics display

5 Entries specific to the HP38/39/40

5.1 Topic Variables and the Topic Outer Loop

These entries are used for the implemtation of applets on the HP38G/39G/40G. On the HP49G, they are included for Hp38/39/40 compatibility, probably in order to allow applet devellopment on the HP49G.

8325F	TopOuterLoop	
83269	TOLSaveUI	
83273	TOLSetTopicUI	
8327D	TOLSetViewUI	
83287	TOLKeyUI	
83291	TOLErrorTrap	
8329B	TOLResUI&Err	
832A5	TOLRestoreUI	
832AF	?ExitThisTop	
84AE4	CALCCXT!	(ob →)
84AF4	CALCCXT@	(→ ob)
84B04	PGMCXT!	(ob →)
84B14	PGMCXT@	(→ ob)
84B24	NOTESCXT!	(ob →)
84B34	NOTESCXT@	(→ ob)
84B44	apletPTR!	(ob →)
84B54	apletPTR@	(→ ob)
84B64	funcPTR!	(ob →)
84B74	funcPTR@	(→ ob)
84B84	polarPTR!	(ob →)
84B94	polarPTR@	(→ ob)
84BA4	paramPTR!	(ob →)
84BB4	paramPTR@	(→ ob)
84BC4	seqPTR!	(ob →)
84BD4	seqPTR@	(→ ob)
84BE4	statPTR!	(ob →)
84BF4	statPTR@	(→ ob)
84C04	solvePTR!	(ob →)
84C14	solvePTR@	(→ ob)
84C24	otherPTR!	(ob →)

84C34	otherPTR@	(→ ob)
81996	TopicVarN!	
81991	TopicVarN@	
84C44	TopicVar1!	(ob →)
84C54	TopicVar1@	(→ ob)
84C64	TopicVar2!	(ob →)
84C74	TopicVar2@	(→ ob)
84C84	TopicVar3!	(ob →)
84C94	TopicVar3@	(→ ob)
84CA4	TopicVar4!	(ob →)
84CB4	TopicVar4@	(→ ob)
84CC4	TopicVar5!	(ob →)
84CD4	TopicVar5@	(→ ob)
84CE4	TopicVar6!	(ob →)
84CF4	TopicVar6@	(→ ob)
84D04	TopicVar7!	(ob →)
84D14	TopicVar7@	(→ ob)
84D24	TopicVar8!	(ob →)
84D34	TopicVar8@	(→ ob)
84D44	TopicVar9!	(ob →)
84D54	TopicVar9@	(→ ob)
84D64	TopicVar10!	(ob →)
84D74	TopicVar10@	(→ ob)
84D84	TopicVar11!	(ob →)
84D94	TopicVar11@	(→ ob)
84DA4	TopicVar12!	(ob →)
84DB4	TopicVar12@	(→ ob)
84DC4	TopicVar13!	(ob →)
84DD4	TopicVar13@	(→ ob)
84DE4	TopicVar14!	(ob →)
84DF4	TopicVar14@	(→ ob)
84E04	TopicVar15!	(ob →)
84E14	TopicVar15@	(→ ob)
84E24	TopicVar16!	(ob →)
84E34	TopicVar16@	(→ ob)
84E44	TopicVar17!	(ob →)
84E54	TopicVar17@	(→ ob)
84E64	TopicVar18!	(ob →)

```
84E74     TopicVar18@          ( → ob )
84E84     TopicVar19!          ( ob → )
84E94     TopicVar19@          ( → ob )
84EA4     TopicVar20!          ( ob → )
84EB4     TopicVar20@          ( → ob )
84EC4     TopicVar21!          ( ob → )
84ED4     TopicVar21@          ( → ob )
84EE4     TopicVar22!          ( ob → )
84EF4     TopicVar22@          ( → ob )
84F04     TopicVar23!          ( ob → )
84F14     TopicVar23@          ( → ob )
84F24     TopicVar24!          ( ob → )
84F34     TopicVar24@          ( → ob )
84F44     TopicVar25!          ( ob → )
84F54     TopicVar25@          ( → ob )
84F64     TopicVar26!          ( ob → )
84F74     TopicVar26@          ( → ob )
84F84     TopicVar27!          ( ob → )
84F94     TopicVar27@          ( → ob )
84FA4     TopicVar28!          ( ob → )
84FB4     TopicVar28@          ( → ob )
84FC4     TopicVar29!          ( ob → )
84FD4     TopicVar29@          ( → ob )
84FE4     TopicVar30!          ( ob → )
84FF4     TopicVar30@          ( → ob )
85004     TopicVar31!          ( ob → )
85014     TopicVar31@          ( → ob )
85024     TopicVar32!          ( ob → )
85034     TopicVar32@          ( → ob )
85044     TopicVar33!          ( ob → )
85054     TopicVar33@          ( → ob )
85064     TopicVar34!          ( ob → )
85074     TopicVar34@          ( → ob )
85084     TopicVar35!          ( ob → )
85094     TopicVar35@          ( → ob )
850A4     TopicVar36!          ( ob → )
850B4     TopicVar36@          ( → ob )
850C4     TopicVar37!          ( ob → )
```

```
850D4     TopicVar37@          ( → ob )
850E4     TopicVar38!          ( ob → )
850F4     TopicVar38@          ( → ob )
85104     TopicVar39!          ( ob → )
85114     TopicVar39@          ( → ob )
85124     TopicVar40!          ( ob → )
85134     TopicVar40@          ( → ob )
85144     TopicVar41!          ( ob → )
85154     TopicVar41@          ( → ob )
85164     TopicVar42!          ( ob → )
85174     TopicVar42@          ( → ob )
85184     TopicVar43!          ( ob → )
85194     TopicVar43@          ( → ob )
851A4     TopicVar44!          ( ob → )
851B4     TopicVar44@          ( → ob )
851C4     TopicVar45!          ( ob → )
851D4     TopicVar45@          ( → ob )
851E4     TopicVar46!          ( ob → )
851F4     TopicVar46@          ( → ob )
85204     TopicVar47!          ( ob → )
85214     TopicVar47@          ( → ob )
85224     TopicVar48!          ( ob → )
85234     TopicVar48@          ( → ob )
85244     TopicVar49!          ( ob → )
85254     TopicVar49@          ( → ob )
85264     TopicVar50!          ( ob → )
85274     TopicVar50@          ( → ob )
85284     TopicVar51!          ( ob → )
85294     TopicVar51@          ( → ob )
852A4     TopicVar52@          ( ob → )
852B4     TopicVar52!          ( → ob )
852C4     TopicVar53@          ( ob → )
852E0     TopicVar53!          ( → ob )
85307     TopicVar54@          ( ob → )
85317     TopicVar54!          ( → ob )
85327     TopicVar55@          ( ob → )
85337     TopicVar55!          ( → ob )
85347     TopicVar56@          ( ob → )
```

```
85357     TopicVar56!          ( → ob )
85367     TopicVar57@          ( ob → )
85377     TopicVar57!          ( → ob )
85387     TopicVar58@          ( ob → )
85397     TopicVar58!          ( → ob )
853A7     TopicVar59@          ( ob → )
853B7     TopicVar59!          ( → ob )
853C7     TopicVar60@          ( ob → )
853D7     TopicVar60!          ( → ob )
853E7     TopicVar61@          ( ob → )
853F7     TopicVar61!          ( → ob )
85407     TopicVar62@          ( ob → )
85417     TopicVar62!          ( → ob )
85427     TopicVar63@          ( ob → )
85437     TopicVar63!          ( → ob )
85447     TopicVar64@          ( ob → )
85457     TopicVar64!          ( → ob )
85467     TopicVar65@          ( ob → )
85477     TopicVar65!          ( → ob )
85487     TopicVar66@          ( ob → )
85497     TopicVar66!          ( → ob )
854A7     TopicVar67@          ( ob → )
854B7     TopicVar67!          ( → ob )
854C7     TopicVar68@          ( ob → )
854D7     TopicVar68!          ( → ob )
854E7     TopicVar69@          ( ob → )
854F7     TopicVar69!          ( → ob )
85507     TopicVar70@          ( ob → )
85517     TopicVar70!          ( → ob )
85527     TopicVar71@          ( ob → )
85537     TopicVar71!          ( → ob )
85547     TopicVar72@          ( ob → )
85557     TopicVar72!          ( → ob )
85567     TopicVar73@          ( ob → )
85577     TopicVar73!          ( → ob )
85587     TopicVar74@          ( ob → )
85597     TopicVar74!          ( → ob )
855A7     TopicVar75@          ( ob → )
```

855B7	TopicVar75!	(→ ob)
855C7	TopicVar76@	(ob →)
855D7	TopicVar76!	(→ ob)
855E7	TopicVar77@	(ob →)
855F7	TopicVar77!	(→ ob)
85607	TopicVar78@	(ob →)
85617	TopicVar78!	(→ ob)
85627	TopicVar79@	(ob →)
85637	TopicVar79!	(→ ob)
85647	TopicVar80@	(ob →)
85657	TopicVar80!	(→ ob)
85667	TopicVar81@	(ob →)
85677	TopicVar81!	(→ ob)
85687	TopicVar82@	(ob →)
85697	TopicVar82!	(→ ob)
856A7	TopicVar83@	(ob →)
856B7	TopicVar83!	(→ ob)
856C7	TopicVar84@	(ob →)
856D7	TopicVar84!	(→ ob)
856E7	TopicVar85@	(ob →)
856F7	TopicVar85!	(→ ob)
85707	TopicVar86@	(ob →)
85717	TopicVar86!	(→ ob)
85727	TopicVar87@	(ob →)
85737	TopicVar87!	(→ ob)
85747	TopicVar88@	(ob →)
85757	TopicVar88!	(→ ob)
85767	TopicVar89@	(ob →)
85777	TopicVar89!	(→ ob)
85787	TopicVar90@	(ob →)
85797	TopicVar90!	(→ ob)
857A7	TopicVar91!	(ob →)
857B7	TopicVar91@	(→ ob)
857C7	TOLVar1!	(ob →)
857D7	TOLVar1@	(→ ob)
857E7	TOLVar2!	(ob →)
857F7	TOLVar2@	(→ ob)
85807	TOLVar3!	(ob →)

85817	TOLVar3@	(→ ob)
85827	TOLVar4!	(ob →)
85837	TOLVar4@	(→ ob)
85847	TOLVar5!	(ob →)
85857	TOLVar5@	(→ ob)
85867	TOLVar6!	(ob →)
85877	TOLVar6@	(→ ob)
85887	TOLVar7!	(ob →)
85897	TOLVar7@	(→ ob)
858A7	TOLVar8!	(ob →)
858B7	TOLVar8@	(→ ob)
858C7	TOLVar9!	(ob →)
858D7	TOLVar9@	(→ ob)
858E7	TOLVar10!	(ob →)
858F7	TOLVar10@	(→ ob)
85907	TOLVar11!	(ob →)
85917	TOLVar11@	(→ ob)
85927	TOLVar12!	(ob →)
85937	TOLVar12@	(→ ob)
85947	TOLVar13!	(ob →)
85957	TOLVar13@	(→ ob)
85967	TOLVar14!	(ob →)
85977	TOLVar14@	(→ ob)
85987	TOLVar15!	(ob →)
85997	TOLVar15@	(→ ob)
859A7	TOLVar16!	(ob →)
859B7	TOLVar16@	(→ ob)
859C7	TOLVar17!	(ob →)
859D7	TOLVar17@	(→ ob)
859E7	TOLVar18!	(ob →)
859F7	TOLVar18@	(→ ob)
85A07	TOLVar19!	(ob →)
85A17	TOLVar19@	(→ ob)
85A27	TOLVar20!	(ob →)
85A37	TOLVar20@	(→ ob)
85A47	TOLVar21!	(ob →)
85A57	TOLVar21@	(→ ob)
85A67	TOLVar22!	(ob →)

85A77	TOLVar22@	(→ ob)
85A87	TOLVar23!	(ob →)
85A97	TOLVar23@	(→ ob)
85AA7	TOLVar24!	(ob →)
85AB7	TOLVar24@	(→ ob)
85AC7	TOLVar25!	(ob →)
85AD7	TOLVar25@	(→ ob)
85AE9	TOLVar26!	(ob →)
85AFB	TOLVar26@	(→ ob)
85B0D	TOLVar27!	(ob →)
85B1F	TOLVar27@	(→ ob)
85B31	TOLVar28!	(ob →)
85B43	TOLVar28@	(→ ob)
85B55	TOLVar29!	(ob →)
85B67	TOLVar29@	(→ ob)
85B79	TOLVar30!	(ob →)
85B8B	TOLVar30@	(→ ob)
85B9D	TOLVar31!	(ob →)
85BAF	TOLVar31@	(→ ob)
85BC1	TOLVar32!	(ob →)
85BD3	TOLVar32@	(→ ob)
85BE5	TOLVar33!	(ob →)
85BF7	TOLVar33@	(→ ob)
85C09	TOLVar34!	(ob →)
85C1B	TOLVar34@	(→ ob)
85C2D	TOLVar35!	(ob →)
85C3F	TOLVar35@	(→ ob)
85C51	TOLVar36!	(ob →)
85C63	TOLVar36@	(→ ob)
85C75	TOLVar37!	(ob →)
85C87	TOLVar37@	(→ ob)
85C99	TOLVar38!	(ob →)
85CAB	TOLVar38@	(→ ob)
85CBD	TOLVar39!	(ob →)
85CCF	TOLVar39@	(→ ob)
85CE1	TOLVar40!	(ob →)
85CF3	TOLVar40@	(→ ob)
85D05	TOLVar41!	(ob →)

85D17	TOLVar41@	(→ ob)
85D29	TOLVar42!	(ob →)
85D3B	TOLVar42@	(→ ob)
85D4D	TOLVar43!	(ob →)
85D5F	TOLVar43@	(→ ob)
85D71	TOLVar44!	(ob →)
85D83	TOLVar44@	(→ ob)
85D95	TOLVar45!	(ob →)
85DA7	TOLVar45@	(→ ob)
85DB9	TOLVar46!	(ob →)
85DCB	TOLVar46@	(→ ob)
85DDD	TOLVar47!	(ob →)
85DEF	TOLVar47@	(→ ob)
85E01	TOLVar48!	(ob →)
85E13	TOLVar48@	(→ ob)
85E25	TOLVar49!	(ob →)
85E37	TOLVar49@	(→ ob)
85E49	TOLVar50!	(ob →)
85E5B	TOLVar50@	(→ ob)
85E6D	TOLVar51!	(ob →)
85E7F	TOLVar51@	(→ ob)
85E91	TOLVar52!	(ob →)
85EA3	TOLVar52@	(→ ob)
85EB5	TOLVar53!	(ob →)
85EC7	TOLVar53@	(→ ob)
85ED9	TOLVar54!	(ob →)
85EEB	TOLVar54@	(→ ob)
85EFD	TOLVar55!	(ob →)
85F0F	TOLVar55@	(→ ob)
85F21	TOLVar56!	(ob →)
85F33	TOLVar56@	(→ ob)
85F45	TOLVar57!	(ob →)
85F57	TOLVar57@	(→ ob)
85F69	TOLVar58!	(ob →)
85F7B	TOLVar58@	(→ ob)
85F8D	TOLVar59!	(ob →)
85F9F	TOLVar59@	(→ ob)
85FB1	TOLVar60!	(ob →)

85FC3	TOLVar60@	(→ ob)
85FD5	TOLVar61!	(ob →)
85FE7	TOLVar61@	(→ ob)
85FF9	TOLVar62!	(ob →)
8600B	TOLVar62@	(→ ob)
8601D	TOLVar63!	(ob →)
8602F	TOLVar63@	(→ ob)
86041	TOLVar64!	(ob →)
86053	TOLVar64@	(→ ob)
86065	TOLVar65!	(ob →)
86077	TOLVar65@	(→ ob)
86089	TOLVar66!	(ob →)
8609B	TOLVar66@	(→ ob)
860AD	TOLVar67!	(ob →)
860BF	TOLVar67@	(→ ob)
860D1	TOLVar68!	(ob →)
860E3	TOLVar68@	(→ ob)
860F5	TOLVar69!	(ob →)
86107	TOLVar69@	(→ ob)
86119	TOLVar70!	(ob →)
8612B	TOLVar70@	(→ ob)
8613D	TOLVar71!	(ob →)
8614F	TOLVar71@	(→ ob)
86161	TOLVar72!	(ob →)
86173	TOLVar72@	(→ ob)
86185	TOLVar73!	(ob →)
86197	TOLVar73@	(→ ob)
861A9	TOLVar74!	(ob →)
861BB	TOLVar74@	(→ ob)
861CD	TOLVar75!	(ob →)
861DF	TOLVar75@	(→ ob)
861F1	TOLVar76!	(ob →)
86203	TOLVar76@	(→ ob)
86215	TOLVar77!	(ob →)
86227	TOLVar77@	(→ ob)
86239	TOLVar78!	(ob →)
8624B	TOLVar78@	(→ ob)
8625D	TOLVar79!	(ob →)

8626F	TOLVar79@	(→ ob)
86281	TOLVar80!	(ob →)
86293	TOLVar80@	(→ ob)
862A5	TOLVar81!	(ob →)
862B7	TOLVar81@	(→ ob)
862C9	TOLVar82!	(ob →)
862DB	TOLVar82@	(→ ob)
862ED	TOLVar83!	(ob →)
862FF	TOLVar83@	(→ ob)
86311	TOLVar84!	(ob →)
86323	TOLVar84@	(→ ob)
86335	TOLVar85!	(ob →)
86347	TOLVar85@	(→ ob)
86359	TOLVar86!	(ob →)
8636B	TOLVar86@	(→ ob)
8637D	TOLVar87!	(ob →)
8638F	TOLVar87@	(→ ob)
863A1	TOLVar88!	(ob →)
863B3	TOLVar88@	(→ ob)
863C5	TOLVar89!	(ob →)
863D7	TOLVar89@	(→ ob)
863E9	TOLVar90!	(ob →)
863FB	TOLVar90@	(→ ob)
8640D	TOLVar91!	(ob →)
8641F	TOLVar91@	(→ ob)
86431	TOLVar92!	(ob →)
86443	TOLVar92@	(→ ob)
86455	TOLVar93!	(ob →)
86467	TOLVar93@	(→ ob)
86479	TOLVar94!	(ob →)
8648B	TOLVar94@	(→ ob)
8649D	TOLVar95!	(ob →)
864AF	TOLVar95@	(→ ob)
864C1	TOLVar96!	(ob →)
864D3	TOLVar96@	(→ ob)
864E5	TOLVar97!	(ob →)
864F7	TOLVar97@	(→ ob)
86509	TOLVar98!	(ob →)

8651B	TOLVar98@	(→ ob)
8652D	TOLVar99!	(ob →)
8653F	TOLVar99@	(→ ob)
86551	TOLVar100!	(ob →)
86563	TOLVar100@	(→ ob)
86575	TOLVar101!	(ob →)
86587	TOLVar101@	(→ ob)
86599	TOLVar102!	(ob →)
865AB	TOLVar102@	(→ ob)
865BD	TOLVar103!	(ob →)
865CF	TOLVar103@	(→ ob)
865E1	TOLVar104!	(ob →)
865F3	TOLVar104@	(→ ob)
86605	TOLVar105!	(ob →)
86617	TOLVar105@	(→ ob)
86629	TOLVar106!	(ob →)
8663B	TOLVar106@	(→ ob)
8664D	TOLVar107!	(ob →)
8665F	TOLVar107@	(→ ob)
86671	TOLVar108!	(ob →)
86683	TOLVar108@	(→ ob)
86695	TOLVar109!	(ob →)
866A7	TOLVar109@	(→ ob)
866B9	TOLVar110!	(ob →)
866CB	TOLVar110@	(→ ob)
866DD	TOLVar111!	(ob →)
866EF	TOLVar111@	(→ ob)
86701	TOLVar112!	(ob →)
86713	TOLVar112@	(→ ob)
86725	TOLVar113!	(ob →)
86737	TOLVar113@	(→ ob)
86749	TOLVar114!	(ob →)
8675B	TOLVar114@	(→ ob)
8676D	TOLVar115!	(ob →)
8677F	TOLVar115@	(→ ob)
86791	TOLVar116!	(ob →)
867A3	TOLVar116@	(→ ob)
867B5	TOLVar117!	(ob →)

867C7	TOLVar117@	(→ ob)
867D9	TOLVar118!	(ob →)
867EB	TOLVar118@	(→ ob)
867FD	TOLVar119!	(ob →)
8680F	TOLVar119@	(→ ob)
86821	TOLVar120!	(ob →)
86833	TOLVar120@	(→ ob)
86845	TOLVar121!	(ob →)
86857	TOLVar121@	(→ ob)
86869	TOLVar122!	(ob →)
8687B	TOLVar122@	(→ ob)
8688D	TOLVar123!	(ob →)
8689F	TOLVar123@	(→ ob)
868B1	TOLVar124!	(ob →)
868C3	TOLVar124@	(→ ob)
868D5	TOLVar125!	(ob →)
868E7	TOLVar125@	(→ ob)
868F9	TOLVar126!	(ob →)
8690B	TOLVar126@	(→ ob)
8691D	TOLVar127!	(ob →)
8692F	TOLVar127@	(→ ob)
86941	TOLVar128!	(ob →)
86953	TOLVar128@	(→ ob)
86965	TOLVar129!	(ob →)
86977	TOLVar129@	(→ ob)
86989	TOLVar130!	(ob →)
8699B	TOLVar130@	(→ ob)
869AD	TOLVar131!	(ob →)
869BF	TOLVar131@	(→ ob)
869D1	TOLVar132!	(ob →)
869E3	TOLVar132@	(→ ob)
869F5	TOLVar133!	(ob →)
86A07	TOLVar133@	(→ ob)
86A19	TOLVar134!	(ob →)
86A2B	TOLVar134@	(→ ob)
86A3D	TOLVar135!	(ob →)
86A4F	TOLVar135@	(→ ob)
86A61	TOLVar136!	(ob →)

```
86A73     TOLVar136@          ( → ob )
86A85     TOLVar137!          ( ob → )
86A97     TOLVar137@          ( → ob )
86AA9     TOLVar138!          ( ob → )
86ABB     TOLVar138@          ( → ob )
86ACD     TOLVar139!          ( ob → )
86ADF     TOLVar139@          ( → ob )
86AF1     TOLVar140!          ( ob → )
86B03     TOLVar140@          ( → ob )
86B15     TOLVar141!          ( ob → )
86B27     TOLVar141@          ( → ob )
86B39     TOLVar142!          ( ob → )
86B4B     TOLVar142@          ( → ob )
86B5D     TOLVar143!          ( ob → )
86B6F     TOLVar143@          ( → ob )
86B81     TOLVar144!          ( ob → )
86B93     TOLVar144@          ( → ob )
86BA5     TOLVar145!          ( ob → )
86BB7     TOLVar145@          ( → ob )
86BC9     TOLVar146!          ( ob → )
86BDB     TOLVar146@          ( → ob )
86BED     TOLVar147!          ( ob → )
86BFF     TOLVar147@          ( → ob )
86C11     TOLVar148!          ( ob → )
86C23     TOLVar148@          ( → ob )
86C35     TOLVar149!          ( ob → )
86C47     TOLVar149@          ( → ob )
86C59     TOLVar150!          ( ob → )
86C6B     TOLVar150@          ( → ob )
86C7D     TOLVar151!          ( ob → )
86C8F     TOLVar151@          ( → ob )
86CA1     TOLVar152!          ( ob → )
86CB3     TOLVar152@          ( → ob )
86CC5     TOLVar153!          ( ob → )
86CD7     TOLVar153@          ( → ob )
86CE9     TOLVar154!          ( ob → )
86CFB     TOLVar154@          ( → ob )
86D0D     TOLVar155!          ( ob → )
```

```
86D1F     TOLVar155@          ( → ob )
86D31     TOLVar156!          ( ob → )
86D43     TOLVar156@          ( → ob )
86D55     TOLVar157!          ( ob → )
86D67     TOLVar157@          ( → ob )
86D79     TOLVar158!          ( ob → )
86D8B     TOLVar158@          ( → ob )
86D9D     TOLVar159!          ( ob → )
86DAF     TOLVar159@          ( → ob )
86DC1     TOLVar160!          ( ob → )
86DD3     TOLVar160@          ( → ob )
86DE5     TOLVar161!          ( ob → )
86DF7     TOLVar161@          ( → ob )
86E09     TOLVar162!          ( ob → )
86E1B     TOLVar162@          ( → ob )
86E2D     TOLVar163!          ( ob → )
86E3F     TOLVar163@          ( → ob )
86E51     TOLVar164!          ( ob → )
86E63     TOLVar164@          ( → ob )
86E75     TOLVar165!          ( ob → )
86E87     TOLVar165@          ( → ob )
86E99     TOLVar166!          ( ob → )
86EAB     TOLVar166@          ( → ob )
86EBD     TOLVar167!          ( ob → )
86ECF     TOLVar167@          ( → ob )
86EE1     TOLVar168!          ( ob → )
86EF3     TOLVar168@          ( → ob )
86F05     TOLVar169!          ( ob → )
86F17     TOLVar169@          ( → ob )
86F29     TOLVar170!          ( ob → )
86F3B     TOLVar170@          ( → ob )
86F4D     TOLVar171!          ( ob → )
86F5F     TOLVar171@          ( → ob )
86F71     TOLVar172!          ( ob → )
86F83     TOLVar172@          ( → ob )
86F95     TOLVar173!          ( ob → )
86FA7     TOLVar173@          ( → ob )
86FB9     TOLVar174!          ( ob → )
```

```
86FCB     TOLVar174@          ( → ob )
86FDD     TOLVar175!          ( ob → )
86FEF     TOLVar175@          ( → ob )
87001     TOLVar176!          ( ob → )
87013     TOLVar176@          ( → ob )
87025     TOLVar177!          ( ob → )
87037     TOLVar177@          ( → ob )
87049     TOLVar178!          ( ob → )
8705B     TOLVar178@          ( → ob )
8706D     TOLVar179!          ( ob → )
8707F     TOLVar179@          ( → ob )
87091     TOLVar180!          ( ob → )
870A3     TOLVar180@          ( → ob )
870B5     TOLVar181!          ( ob → )
870C7     TOLVar181@          ( → ob )
870D9     TOLVar182!          ( ob → )
870EB     TOLVar182@          ( → ob )
870FD     TOLVar183!          ( ob → )
8710F     TOLVar183@          ( → ob )
87121     TOLVar184!          ( ob → )
87133     TOLVar184@          ( → ob )
87145     TOLVar185!          ( ob → )
87157     TOLVar185@          ( → ob )
87169     TOLVar186!          ( ob → )
8717B     TOLVar186@          ( → ob )
8718D     TOLVar187!          ( ob → )
8719F     TOLVar187@          ( → ob )
871B1     TOLVar188!          ( ob → )
871C3     TOLVar188@          ( → ob )
871D5     TOLVar189!          ( ob → )
871E7     TOLVar189@          ( → ob )
871F9     TOLVar190!          ( ob → )
8720B     TOLVar190@          ( → ob )
8721D     TOLVar191!          ( ob → )
8722F     TOLVar191@          ( → ob )
87241     TOLVar192!          ( ob → )
87253     TOLVar192@          ( → ob )
87265     TOLVar193!          ( ob → )
```

```
87277     TOLVar193@          ( → ob )
87289     TOLVar194!          ( ob → )
8729B     TOLVar194@          ( → ob )
872AD     TOLVar195!          ( ob → )
872BF     TOLVar195@          ( → ob )
872D1     TOLVar196!          ( ob → )
872E3     TOLVar196@          ( → ob )
872F5     TOLVar197!          ( ob → )
87307     TOLVar197@          ( → ob )
87319     TOLVar198!          ( ob → )
8732B     TOLVar198@          ( → ob )
8733D     TOLVar199!          ( ob → )
8734F     TOLVar199@          ( → ob )
87361     TOLVar200!          ( ob → )
87373     TOLVar200@          ( → ob )
87385     TOLVar201!          ( ob → )
87397     TOLVar201@          ( → ob )
873A9     TOLVar202!          ( ob → )
873BB     TOLVar202@          ( → ob )
873CD     TOLVar203!          ( ob → )
873DF     TOLVar203@          ( → ob )
873F1     TOLVar204!          ( ob → )
87403     TOLVar204@          ( → ob )
87415     TOLVar205!          ( ob → )
87427     TOLVar205@          ( → ob )
87439     TOLVar206!          ( ob → )
8744B     TOLVar206@          ( → ob )
8745D     TOLVar207!          ( ob → )
8746F     TOLVar207@          ( → ob )
87481     TOLVar208!          ( ob → )
87493     TOLVar208@          ( → ob )
874A5     TOLVar209!          ( ob → )
874B7     TOLVar209@          ( → ob )
874C9     TOLVar210!          ( ob → )
874DB     TOLVar210@          ( → ob )
874ED     TOLVar211!          ( ob → )
874FF     TOLVar211@          ( → ob )
87511     TOLVar212!          ( ob → )
```

87523	TOLVar212@	(→ ob)
87535	TOLVar213!	(ob →)
87547	TOLVar213@	(→ ob)
87559	TOLVar214!	(ob →)
8756B	TOLVar214@	(→ ob)
8757D	TOLVar215!	(ob →)
8758F	TOLVar215@	(→ ob)
875A1	TOLVar216!	(ob →)
875B3	TOLVar216@	(→ ob)
875C5	TOLVarN!	(ob →)
875E8	TOLVarN@	(→ ob)
8760B	ClrAllTVars	
87641	ClrAllTOLVs	
8765D	%0AllTopicVs	
87698	%0AllTOLVars	
876D3	TOLVarSet!	
877A0	%0TOLVarSet	
877F0	1getcxt!	
87804	DoInCxt	
8785E	DoInCalcCxt	
87877	DoInAppCxt	
87890	DoInFuncCxt	
878A9	DoInPolarCxt	
878C2	DoInParamCxt	
878DB	DoInSeqCxt	
878F4	DoInStatCxt	
8790D	DoInSolveCxt	
87926	DoInOtherCxt	
879F3	otherNG?	
87A26	GET@tTYPER	

5.2 Special Variables

82D55	StoVar	(ob id →) Stores to variable. ID is one of the ids returned by any of the commands in the following subsections.

5.2.1 Real HOME variables

`0000F2` `~xa` `( → id )`
The id corresponding to the real variable A.

`0010F2` `~xb` `( → id )`
The id corresponding to the real variable B.

`0020F2` `~xc` `( → id )`
The id corresponding to the real variable C.

`0030F2` `~xd` `( → id )`
The id corresponding to the real variable D.

`0050F2` `~xf` `( → id )`
The id corresponding to the real variable F.

`0060F2` `~xg` `( → id )`
The id corresponding to the real variable G.

`0070F2` `~xh` `( → id )`
The id corresponding to the real variable H.

`0090F2` `~xj` `( → id )`
The id corresponding to the real variable J.

`00A0F2` `~xk` `( → id )`
The id corresponding to the real variable K.

`00B0F2` `~xl` `( → id )`
The id corresponding to the real variable L.

`00C0F2` `~xm` `( → id )`
The id corresponding to the real variable M.

`00D0F2` `~xn` `( → id )`
The id corresponding to the real variable N.

`00E0F2` `~xo` `( → id )`
The id corresponding to the real variable O.

`00F0F2` `~xp` `( → id )`
The id corresponding to the real variable P.

`0100F2` `~xq` `( → id )`
The id corresponding to the real variable Q.

`0110F2` `~xr` `( → id )`
The id corresponding to the real variable R.

`0120F2` `~xs` `( → id )`
The id corresponding to the real variable S.

`0130F2` `~xt` `( → id )`
The id corresponding to the real variable T.

`0140F2` `~xu` `( → id )`
The id corresponding to the real variable U.

`0150F2` `~xv` `( → id )`
The id corresponding to the real variable V.

`0160F2` `~xw` `( → id )`
The id corresponding to the real variable W.

`0170F2` `~xx` `( → id )`
The id corresponding to the real variable X.

`0180F2` `~xy` `( → id )`
The id corresponding to the real variable Y.

`0190F2` `~xz` `( → id )`
The id corresponding to the real variable Z.

`01A0F2` `~xtheta` `( → id )`
The id corresponding to the real variable θ.

5.2.2 Complex HOME variableS

`0310F2` `~xz0` `( → id )`
The id corresponding to the complex variable Z0.

`0320F2` `~xz1` `( → id )`
The id corresponding to the complex variable Z1.

`0330F2` `~xz2` `( → id )`
The id corresponding to the complex variable Z2.

`0340F2` `~xz3` `( → id )`
The id corresponding to the complex variable Z3.

`0350F2` `~xz4` `( → id )`
The id corresponding to the complex variable Z4.

`0360F2` `~xz5` `( → id )`
The id corresponding to the complex variable Z5.

`0370F2` `~xz6` `( → id )`
The id corresponding to the complex variable Z6.

`0380F2` `~xz7` `( → id )`
The id corresponding to the complex variable Z7.

`0390F2` `~xz8` `( → id )`
The id corresponding to the complex variable Z8.

`03A0F2` `~xz9` `( → id )`
The id corresponding to the complex variable Z9.

5.2.3 Matrix HOME variables

`03B0F2` `~xm0` `( → id )`
The id corresponding to the matrix variable M0.

`03C0F2` `~xm1` `( → id )`
The id corresponding to the matrix variable M1.

`03D0F2` `~xm2` `( → id )`
The id corresponding to the matrix variable M2.

`03E0F2` `~xm3` `( → id )`
The id corresponding to the matrix variable M3.

`03F0F2` `~xm4` `( → id )`
The id corresponding to the matrix variable M4.

`0400F2` `~xm5` `( → id )`
The id corresponding to the matrix variable M5.

`0410F2` `~xm6` `( → id )`
The id corresponding to the matrix variable M6.

`0420F2` `~xm7` `( → id )`
The id corresponding to the matrix variable M7.

`0430F2` `~xm8` `( → id )`
The id corresponding to the matrix variable M8.

`0440F2` `~xm9` `( → id )`
The id corresponding to the matrix variable M9.

5.2.4 Graphical HOME variables

`04F0F2` `~xg0` `( → id )`
The id corresponding to the graphics variable G0.

`0500F2` `~xg1` `( → id )`
The id corresponding to the graphics variable G1.

`0510F2` `~xg2` `( → id )`
The id corresponding to the graphics variable G2.

`0520F2` `~xg3` `( → id )`
The id corresponding to the graphics variable G3.

`0530F2` `~xg4` `( → id )`
The id corresponding to the graphics variable G4.

`0540F2` `~xg5` `( → id )`
The id corresponding to the graphics variable G5.

`0550F2` `~xg6` `( → id )`
The id corresponding to the graphics variable G6.

`0560F2` `~xg7` `( → id )`
The id corresponding to the graphics variable G7.

`0570F2` `~xg8` `( → id )`
The id corresponding to the graphics variable G8.

`0580F2` `~xg9` `( → id )`
The id corresponding to the graphics variable G9.

5.2.5 List HOME variables

`0450F2` `~xl0` `( → id )`
The id corresponding to the list variable L0.

`0460F2` `~xl1` `( → id )`
The id corresponding to the list variable L1.

`0470F2` `~xl2` `( → id )`
The id corresponding to the list variable L2.

`0480F2` `~xl3` `( → id )`
The id corresponding to the list variable L3.

`0490F2` `~xl4` `( → id )`
The id corresponding to the list variable L4.

`04A0F2` `~xl5` `( → id )`
The id corresponding to the list variable L5.

`04B0F2` `~xl6` `( → id )`
The id corresponding to the list variable L6.

`04C0F2` `~xl7` `( → id )`
The id corresponding to the list variable L7.

`04D0F2` `~xl8` `( → id )`
The id corresponding to the list variable L8.

`04E0F2` `~xl9` `( → id )`
The id corresponding to the list variable L9.

`81F86` `ClearList0` `( → )`
Clears list 0.

`81F2C` `ClearList1` `( → )`
Clears list 1.

`81F36` `ClearList2` `( → )`
Clears list 2.

`81F40` `ClearList3` `( → )`
Clears list 3.

`81F4A` `ClearList4` `( → )`
Clears list 4.

`81F54` `ClearList5` `( → )`
Clears list 5.

`81F5E` `ClearList6` `( → )`
Clears list 6.

`81F68` `ClearList7` `( → )`
Clears list 7.

`81F72` `ClearList8` `( → )`
Clears list 8.

`81F7C` `ClearList9` `( → )`
Clears list 9.

`28381` `(ClearLists)` `( → )`
Clears all lists.

`81F22` `ClrListUtil`

`81D92` `ClrInAplet`

5.2.6 FUNCTION applet

`0D20F2` `~xF0` `( → id )`
The id corresponding to the `FUNCTION` applet variable F0.

`0C00F2` `~xF1` `( → id )`
The id corresponding to the `FUNCTION` applet variable F1.

`0C20F2` `~xF2` `( → id )`
The id corresponding to the `FUNCTION` applet variable F2.

`0C40F2` `~xF3` `( → id )`
The id corresponding to the `FUNCTION` applet variable F3.

`0C60F2` `~xF4` `( → id )`
The id corresponding to the `FUNCTION` applet variable F4.

`0C80F2` `~xF5` `( → id )`
The id corresponding to the `FUNCTION` applet variable F5.

`0CA0F2` `~xF6` `( → id )`
The id corresponding to the `FUNCTION` applet variable F6.

`0CC0F2` `~xF7` `( → id )`
The id corresponding to the `FUNCTION` applet variable F7.

`0CE0F2` `~xF8` `( → id )`
The id corresponding to the `FUNCTION` applet variable F8.

`0D00F2` `~xF9` `( → id )`
The id corresponding to the `FUNCTION` applet variable F9.

`0D30F2` `~idF0` `( → id )`
The id corresponding to the `FUNCTION` applet variable F0.

`0C10F2` `~idF1` `( → id )`
The id corresponding to the `FUNCTION` applet variable F1.

`0C30F2` `~idF2` `( → id )`
The id corresponding to the `FUNCTION` applet variable F2.

`0C50F2` `~idF3` `( → id )`
The id corresponding to the `FUNCTION` applet variable F3.

`0C70F2` `~idF4` `( → id )`
The id corresponding to the `FUNCTION` applet variable F4.

`0C90F2` `~idF5` `( → id )`
The id corresponding to the `FUNCTION` applet variable F5.

`0CB0F2` `~idF6` `( → id )`
The id corresponding to the `FUNCTION` applet variable F6.

`0CD0F2` `~idF7` `( → id )`
The id corresponding to the `FUNCTION` applet variable F7.

`0CF0F2` `~idF8` `( → id )`
The id corresponding to the `FUNCTION` applet variable F8.

`0D10F2` `~idF9` `( → id )`
The id corresponding to the `FUNCTION` applet variable F9.

5.2.7 PARAMETRIC applet

`0F80F2` `~xX0` `( → id )`
The id corresponding to the `PARAMETRIC` applet variable X0.

`0D40F2` `~xX1` `( → id )`
The id corresponding to the `PARAMETRIC` applet variable X1.

`0D80F2` `~xX2` `( → id )`
The id corresponding to the `PARAMETRIC` applet variable X2.

`0DC0F2` `~xX3` `( → id )`
The id corresponding to the `PARAMETRIC` applet variable X3.

`0E00F2` `~xX4` `( → id )`
The id corresponding to the `PARAMETRIC` applet variable X4.

`0E40F2` `~xX5` `( → id )`
The id corresponding to the `PARAMETRIC` applet variable X5.

`0E80F2` `~xX6` `( → id )`
The id corresponding to the `PARAMETRIC` applet variable X6.

`0EC0F2` `~xX7` `( → id )`
The id corresponding to the `PARAMETRIC` applet variable X7.

`0F00F2` `~xX8` `( → id )`
The id corresponding to the `PARAMETRIC` applet variable X8.

`0F40F2` `~xX9` `( → id )`
The id corresponding to the `PARAMETRIC` applet variable X9.

`0FA0F2` `~xY0` `( → id )`
The id corresponding to the `PARAMETRIC` applet variable Y0.

`0D60F2` `~xY1` `( → id )`
The id corresponding to the `PARAMETRIC` applet variable Y1.

`0DA0F2` `~xY2` `( → id )`
The id corresponding to the `PARAMETRIC` applet variable Y2.

`0DE0F2` `~xY3` `( → id )`
The id corresponding to the `PARAMETRIC` applet variable Y3.

`0E20F2` `~xY4` `( → id )`
The id corresponding to the `PARAMETRIC` applet variable Y4.

`0E60F2` `~xY5` `( → id )`
The id corresponding to the `PARAMETRIC` applet variable Y5.

`0EA0F2` `~xY6` `( → id )`
The id corresponding to the `PARAMETRIC` applet variable Y6.

`0EE0F2` `~xY7` `( → id )`
The id corresponding to the `PARAMETRIC` applet variable Y7.

`0F20F2` `~xY8` `( → id )`
The id corresponding to the `PARAMETRIC` applet variable Y8.

`0F60F2` `~xY9` `( → id )`
The id corresponding to the `PARAMETRIC` applet variable Y9.

`0F90F2` `~idX0` `( → id )`
The id corresponding to the `PARAMETRIC` applet variable X0.

`0D50F2` `~idX1` `( → id )`
The id corresponding to the `PARAMETRIC` applet variable X1.

`0D90F2` `~idX2` `( → id )`
The id corresponding to the `PARAMETRIC` applet variable X2.

`0DD0F2` `~idX3` `( → id )`
The id corresponding to the `PARAMETRIC` applet variable X3.

`0E10F2` `~idX4` `( → id )`
The id corresponding to the `PARAMETRIC` applet variable X4.

`0E50F2` `~idX5` `( → id )`
The id corresponding to the `PARAMETRIC` applet variable X5.

`0E90F2` `~idX6` `( → id )`
The id corresponding to the `PARAMETRIC` applet variable X6.

`0ED0F2` `~idX7` `( → id )`
The id corresponding to the `PARAMETRIC` applet variable X7.

`0F10F2` `~idX8` `( → id )`
The id corresponding to the `PARAMETRIC` applet variable X8.

`0F50F2` `~idX9` (→ id)
The id corresponding to the PARAMETRIC applet variable X9.

`0FB0F2` `~idY0` (→ id)
The id corresponding to the PARAMETRIC applet variable Y0.

`0D70F2` `~idY1` (→ id)
The id corresponding to the PARAMETRIC applet variable Y1.

`0DB0F2` `~idY2` (→ id)
The id corresponding to the PARAMETRIC applet variable Y2.

`0DF0F2` `~idY3` (→ id)
The id corresponding to the PARAMETRIC applet variable Y3.

`0E30F2` `~idY4` (→ id)
The id corresponding to the PARAMETRIC applet variable Y4.

`0E70F2` `~idY5` (→ id)
The id corresponding to the PARAMETRIC applet variable Y5.

`0EB0F2` `~idY6` (→ id)
The id corresponding to the PARAMETRIC applet variable Y6.

`0EF0F2` `~idY7` (→ id)
The id corresponding to the PARAMETRIC applet variable Y7.

`0F30F2` `~idY8` (→ id)
The id corresponding to the PARAMETRIC applet variable Y8.

`0F70F2` `~idY9` (→ id)
The id corresponding to the PARAMETRIC applet variable Y9.

5.2.8 POLAR applet

`10E0F2` `~xR0` (→ id)
The id corresponding to the POLAR applet variable R0.

`0FC0F2` `~xR1` (→ id)
The id corresponding to the POLAR applet variable R1.

`0FE0F2` `~xR2` (→ id)
The id corresponding to the POLAR applet variable R2.

`1000F2` `~xR3` (→ id)
The id corresponding to the POLAR applet variable R3.

`1020F2` `~xR4` (→ id)
The id corresponding to the POLAR applet variable R4.

`1040F2` `~xR5` (→ id)
The id corresponding to the POLAR applet variable R5.

`1060F2` `~xR6` (→ id)
The id corresponding to the POLAR applet variable R6.

`1080F2` `~xR7` (→ id)
The id corresponding to the POLAR applet variable R7.

`10A0F2` `~xR8` (→ id)
The id corresponding to the POLAR applet variable R8.

`10C0F2` `~xR9` (→ id)
The id corresponding to the POLAR applet variable R9.

`10F0F2` `~idR0` (→ id)
The id corresponding to the POLAR applet variable R0.

`0FD0F2` `~idR1` (→ id)
The id corresponding to the POLAR applet variable R1.

`0FF0F2` `~idR2` (→ id)
The id corresponding to the POLAR applet variable R2.

`1010F2` `~idR3` (→ id)
The id corresponding to the POLAR applet variable R3.

`1030F2` `~idR4` (→ id)
The id corresponding to the POLAR applet variable R4.

`1050F2` `~idR5` (→ id)
The id corresponding to the POLAR applet variable R5.

`1070F2` `~idR6` (→ id)
The id corresponding to the POLAR applet variable R6.

`1090F2` `~idR7` (→ id)
The id corresponding to the POLAR applet variable R7.

`10B0F2` `~idR8` (→ id)
The id corresponding to the POLAR applet variable R8.

`10D0F2` `~idR9` (→ id)
The id corresponding to the POLAR applet variable R9.

5.2.9 SEQUENCE applet

`0BE0F2` `~xU0` `( → id )`
The id corresponding to the SEQUENCE applet variable U0.

`0AC0F2` `~xU1` `( → id )`
The id corresponding to the SEQUENCE applet variable U1.

`0AE0F2` `~xU2` `( → id )`
The id corresponding to the SEQUENCE applet variable U2.

`0B00F2` `~xU3` `( → id )`
The id corresponding to the SEQUENCE applet variable U3.

`0B20F2` `~xU4` `( → id )`
The id corresponding to the SEQUENCE applet variable U4.

`0B40F2` `~xU5` `( → id )`
The id corresponding to the SEQUENCE applet variable U5.

`0B60F2` `~xU6` `( → id )`
The id corresponding to the SEQUENCE applet variable U6.

`0B80F2` `~xU7` `( → id )`
The id corresponding to the SEQUENCE applet variable U7.

`0BA0F2` `~xU8` `( → id )`
The id corresponding to the SEQUENCE applet variable U8.

`0BC0F2` `~xU9` `( → id )`
The id corresponding to the SEQUENCE applet variable U9.

`0BF0F2` `~idU0` `( → id )`
The id corresponding to the SEQUENCE applet variable U0.

`0AD0F2` `~idU1` `( → id )`
The id corresponding to the SEQUENCE applet variable U1.

`0AF0F2` `~idU2` `( → id )`
The id corresponding to the SEQUENCE applet variable U2.

`0B10F2` `~idU3` `( → id )`
The id corresponding to the SEQUENCE applet variable U3.

`0B30F2` `~idU4` `( → id )`
The id corresponding to the SEQUENCE applet variable U4.

0B50F2 ~idU5 (→ id)
The id corresponding to the SEQUENCE applet variable U5.

0B70F2 ~idU6 (→ id)
The id corresponding to the SEQUENCE applet variable U6.

0B90F2 ~idU7 (→ id)
The id corresponding to the SEQUENCE applet variable U7.

0BB0F2 ~idU8 (→ id)
The id corresponding to the SEQUENCE applet variable U8.

0BD0F2 ~idU9 (→ id)
The id corresponding to the SEQUENCE applet variable U9.

5.2.10 SOLVE applet

08E0F2 ~idE0 (→ id)
The id corresponding to the SOLVE applet variable E0.

0850F2 ~idE1 (→ id)
The id corresponding to the SOLVE applet variable E1.

0860F2 ~idE2 (→ id)
The id corresponding to the SOLVE applet variable E2.

0870F2 ~idE3 (→ id)
The id corresponding to the SOLVE applet variable E3.

0880F2 ~idE4 (→ id)
The id corresponding to the SOLVE applet variable E4.

0890F2 ~idE5 (→ id)
The id corresponding to the SOLVE applet variable E5.

08A0F2 ~idE6 (→ id)
The id corresponding to the SOLVE applet variable E6.

08B0F2 ~idE7 (→ id)
The id corresponding to the SOLVE applet variable E7.

08C0F2 ~idE8 (→ id)
The id corresponding to the SOLVE applet variable E8.

08D0F2 ~idE9 (→ id)
The id corresponding to the SOLVE applet variable E9.

5.2.11 STATISTICS applet

5FFF4	xC1	(→ id) The id corresponding to the STATISTICS applet variable C1.
6001A	xC2	(→ id) The id corresponding to the STATISTICS applet variable C2.
60040	xC3	(→ id) The id corresponding to the STATISTICS applet variable C3.
60066	xC4	(→ id) The id corresponding to the STATISTICS applet variable C4.
6008C	xC5	(→ id) The id corresponding to the STATISTICS applet variable C5.

5.2.12 Unknown

1240F2	~xs1
1250F2	~xs2
1260F2	~xs3
1270F2	~xs4
1280F2	~xs5
1290F2	~xn1
12A0F2	~xn2
12B0F2	~xn3
12C0F2	~xn4
12D0F2	~xn5
1190F2	~xD0
1100F2	~xD1
1110F2	~xD2
1120F2	~xD3
1130F2	~xD4
1140F2	~xD5
1150F2	~xD6
1160F2	~xD7
1170F2	~xD8
1180F2	~xD9
0040F2	~xE

```
11A0F2    ~xH1
11B0F2    ~xH2
11C0F2    ~xH3
11D0F2    ~xH4
11E0F2    ~xH5
0080F2    ~xI
0250F2    ~xQ1
0260F2    ~xQ3
11F0F2    ~xS1
0A70F2    ~xS1fit
0A20F2    ~xS1mark
1200F2    ~xS2
0A80F2    ~xS2fit
0A30F2    ~xS2mark
1210F2    ~xS3
0A90F2    ~xS3fit
0A40F2    ~xS3mark
1220F2    ~xS4
0AA0F2    ~xS4fit
0A50F2    ~xS4mark
1230F2    ~xS5
0AB0F2    ~xS5fit
0A60F2    ~xS5mark
```

5.3 m

```
0680F0    ~m->DEFACOS
06E0F0    ~m->DEFACOSH
0670F0    ~m->DEFASIN
06D0F0    ~m->DEFASINH
0690F0    ~m->DEFATAN
06F0F0    ~m->DEFATANH
0650F0    ~m->DEFCOS
06B0F0    ~m->DEFCOSH
0640F0    ~m->DEFSIN
06A0F0    ~m->DEFSINH
0660F0    ~m->DEFTAN
06C0F0    ~m->DEFTANH
```

```
0630F0    ~m->TRG
05E0F0    ~m->[]<-*/
05D0F0    ~m->[]<-+-
03F0F0    ~m-[]*
0410F0    ~m-[]+
0420F0    ~m-[]-
0400F0    ~m-[]/
03E0F0    ~m-[]L
03A0F0    ~m1/[]*
03B0F0    ~m1/[]/
03D0F0    ~m1/[]E
03C0F0    ~m1/[]^
0210F0    ~m<->*
01F0F0    ~m<->+
0200F0    ~m<->-
0220F0    ~m<->/
0230F0    ~m<-A-+
0240F0    ~m<-A--
0250F0    ~m<-A/*
0260F0    ~m<-A//
0270F0    ~m<-A^*
0500F0    ~m<-M*E
04E0F0    ~m<-M*^
0490F0    ~m<-M+*
0520F0    ~m<-M+L
04A0F0    ~m<-M-*
0530F0    ~m<-M-L
0510F0    ~m<-M/E
04F0F0    ~m<-M/^
0580F0    ~m<-T*/
0570F0    ~m<-T+-
01B0F0    ~m<-T=
05A0F0    ~m<-[*/
0590F0    ~m<-[+-
0560F0    ~m<T>*
0550F0    ~m<T>+
02A0F0    ~mA->*/
0280F0    ~mA->+-
```

```
0290F0    ~mA->--
02B0F0    ~mA->//
02C0F0    ~mA->^^
01C0F0    ~mAFlq
01D0F0    ~mAFqq
01E0F0    ~mAFrq
05F0F0    ~mCONJ[]
0710F0    ~mCOS+
0740F0    ~mCOSH+
02D0F0    ~mD->/+
02E0F0    ~mD->/-
0310F0    ~mD->E+
0320F0    ~mD->E-
0330F0    ~mD->L*
0340F0    ~mD->L/
02F0F0    ~mD->^+
0300F0    ~mD->^-
0450F0    ~mE[]^
0430F0    ~mE^*
0440F0    ~mE^/
0610F0    ~mIM[]
0460F0    ~mL*^
0470F0    ~mL[]*
0480F0    ~mL[]/
0540F0    ~mM->op
0600F0    ~mRE[]
0700F0    ~mSIN+
0730F0    ~mSINH+
01A0F0    ~mT->=
0720F0    ~mTAN+
0750F0    ~mTANH+
0350F0    ~m[]CHS*
0360F0    ~m[]CHS/
0370F0    ~m[]CHSL
0390F0    ~m[]INVE
0380F0    ~m[]INV^
05C0F0    ~m]->*/
05B0F0    ~m]->+-
```

5.4 Keys

The following entries execute the command corresponding to the corresponding key.

`00D0A1` `~NSKey3.2` ???
The unshifted key 3.2.

`00E0A1` `~NSKey3.3` ???
The unshifted key 3.3.

`0120A1` `~NSKey4.1` ???
The unshifted key 4.1.

`0130A1` `~NSKey4.2` ???
The unshifted key 4.2.

`0140A1` `~NSKey4.3` ???
The unshifted key 4.3.

`0150A1` `~NSKey4.4` ???
The unshifted key 4.4.

`0160A1` `~NSKey4.5` ???
The unshifted key 4.5.

`0170A1` `~NSKey4.6` ???
The unshifted key 4.6.

`0190A1` `~NSKey5.2` ???
The unshifted key 5.2.

`01A0A1` `~NSKey5.3` ???
The unshifted key 5.3.

`01B0A1` `~NSKey5.4` ???
The unshifted key 5.4.

`01C0A1` `~NSKey5.5` ???
The unshifted key 5.5.

`01D0A1` `~NSKey6.1` ???
The unshifted key 6.1.

`01E0A1` `~NSKey6.2` ???
The unshifted key 6.2.

`01F0A1` `~NSKey6.3` ???
The unshifted key 6.3.

`0200A1` `~NSKey6.4` ???
The unshifted key 6.4.

`0210A1` `~NSKey6.5` ???
The unshifted key 6.5.

`0220A1` `~NSKey7.1` ???
The unshifted key 7.1.

`0230A1` `~NSKey7.2` ???
The unshifted key 7.2.

`0240A1` `~NSKey7.3` ???
The unshifted key 7.3.

`0250A1` `~NSKey7.4` ???
The unshifted key 7.4.

0260A1	~NSKey7.5	??? The unshifted key 7.5.
0270A1	~NSKey8.1	??? The unshifted key 8.1.
0280A1	~NSKey8.2	??? The unshifted key 8.2.
0290A1	~NSKey8.3	??? The unshifted key 8.3.
02A0A1	~NSKey8.4	??? The unshifted key 8.4.
02B0A1	~NSKey8.5	??? The unshifted key 8.5.
02D0A1	~NSKey9.2	??? The unshifted key 9.2.
02E0A1	~NSKey9.3	??? The unshifted key 9.3.
02F0A1	~NSKey9.4	??? The unshifted key 9.4.
0300A1	~NSKey9.5	??? The unshifted key 9.5.
0000A2	~LSKey1.1	??? The shifted key 1.1.
0010A2	~LSKey1.2	??? The shifted key 1.2.
0020A2	~LSKey1.3	??? The shifted key 1.3.
0030A2	~LSKey1.4	??? The shifted key 1.4.
0040A2	~LSKey1.5	??? The shifted key 1.5.
0050A2	~LSKey1.6	??? The shifted key 1.6.
0060A2	~LSKey2.1	??? The shifted key 2.1.
0070A2	~LSKey2.2	??? The shifted key 2.2.
0080A2	~LSKey2.3	??? The shifted key 2.3.
00C0A2	~LSKey3.1	??? The shifted key 3.1.
00D0A2	~LSKey3.2	??? The shifted key 3.2.
00E0A2	~LSKey3.3	??? The shifted key 3.3.
0120A2	~LSKey4.1	??? The shifted key 4.1.

0130A2	~LSKey4.2	??? The shifted key 4.2.
0140A2	~LSKey4.3	??? The shifted key 4.3.
0150A2	~LSKey4.4	??? The shifted key 4.4.
0160A2	~LSKey4.5	??? The shifted key 4.5.
0170A2	~LSKey4.6	??? The shifted key 4.6.
0180A2	~LSKey5.1	??? The shifted key 5.1.
0190A2	~LSKey5.2	??? The shifted key 5.2.
01B0A2	~LSKey5.4	??? The shifted key 5.4.
01C0A2	~LSKey5.5	??? The shifted key 5.5.
01D0A2	~LSKey6.1	??? The shifted key 6.1.
01E0A2	~LSKey6.2	??? The shifted key 6.2.
01F0A2	~LSKey6.3	??? The shifted key 6.3.
0200A2	~LSKey6.4	??? The shifted key 6.4.
0210A2	~LSKey6.5	??? The shifted key 6.5.
0220A2	~LSKey7.1	??? The shifted key 7.1.
0230A2	~LSKey7.2	??? The shifted key 7.2.
0240A2	~LSKey7.3	??? The shifted key 7.3.
0250A2	~LSKey7.4	??? The shifted key 7.4.
0260A2	~LSKey7.5	??? The shifted key 7.5.
0270A2	~LSKey8.1	??? The shifted key 8.1.
0280A2	~LSKey8.2	??? The shifted key 8.2.
0290A2	~LSKey8.3	??? The shifted key 8.3.
02A0A2	~LSKey8.4	??? The shifted key 8.4.

`02B0A2` `~LSKey8.5` `???`
The shifted key 8.5.

`02C0A2` `~LSKey9.1` `???`
The shifted key 9.1.

`02D0A2` `~LSKey9.2` `???`
The shifted key 9.2.

`02E0A2` `~LSKey9.3` `???`
The shifted key 9.3.

`02F0A2` `~LSKey9.4` `???`
The shifted key 9.4.

`0300A2` `~LSKey9.5` `???`
The shifted key 9.5.

`0000A4` `~ANSKey1.1` `???`
The key 1.1 in alpha mode.

`0010A4` `~ANSKey1.2` `???`
The key 1.2 in alpha mode.

`0020A4` `~ANSKey1.3` `???`
The key 1.3 in alpha mode.

`0030A4` `~ANSKey1.4` `???`
The key 1.4 in alpha mode.

`0040A4` `~ANSKey1.5` `???`
The key 1.5 in alpha mode.

`0050A4` `~ANSKey1.6` `???`
The key 1.6 in alpha mode.

`0060A4` `~ANSKey2.1` `???`
The key 2.1 in alpha mode.

`0070A4` `~ANSKey2.2` `???`
The key 2.2 in alpha mode.

`0080A4` `~ANSKey2.3` `???`
The key 2.3 in alpha mode.

`0090A4` `~ANSKey2.4` `???`
The key 2.4 in alpha mode.

`00A0A4` `~ANSKey2.5` `???`
The key 2.5 in alpha mode.

`00B0A4` `~ANSKey2.6` `???`
The key 2.6 in alpha mode.

`00C0A4` `~ANSKey3.1` `???`
The key 3.1 in alpha mode.

`00D0A4` `~ANSKey3.2` `???`
The key 3.2 in alpha mode.

`00E0A4` `~ANSKey3.3` `???`
The key 3.3 in alpha mode.

`00F0A4` `~ANSKey3.4` `???`
The key 3.4 in alpha mode.

`0100A4` `~ANSKey3.5` `???`
The key 3.5 in alpha mode.

0110A4	~ANSKey3.6	??? The key 3.6 in alpha mode.
0120A4	~ANSKey4.1	??? The key 4.1 in alpha mode.
0130A4	~ANSKey4.2	??? The key 4.2 in alpha mode.
0140A4	~ANSKey4.3	??? The key 4.3 in alpha mode.
0150A4	~ANSKey4.4	??? The key 4.4 in alpha mode.
0160A4	~ANSKey4.5	??? The key 4.5 in alpha mode.
0170A4	~ANSKey4.6	??? The key 4.6 in alpha mode.
0180A4	~ANSKey5.1	??? The key 5.1 in alpha mode.
0190A4	~ANSKey5.2	??? The key 5.2 in alpha mode.
01A0A4	~ANSKey5.3	??? The key 5.3 in alpha mode.
01B0A4	~ANSKey5.4	??? The key 5.4 in alpha mode.
01C0A4	~ANSKey5.5	??? The key 5.5 in alpha mode.
01D0A4	~ANSKey6.1	??? The key 6.1 in alpha mode.
01E0A4	~ANSKey6.2	??? The key 6.2 in alpha mode.
01F0A4	~ANSKey6.3	??? The key 6.3 in alpha mode.
0200A4	~ANSKey6.4	??? The key 6.4 in alpha mode.
0210A4	~ANSKey6.5	??? The key 6.5 in alpha mode.
0220A4	~ANSKey7.1	??? The key 7.1 in alpha mode.
0230A4	~ANSKey7.2	??? The key 7.2 in alpha mode.
0240A4	~ANSKey7.3	??? The key 7.3 in alpha mode.
0250A4	~ANSKey7.4	??? The key 7.4 in alpha mode.
0260A4	~ANSKey7.5	??? The key 7.5 in alpha mode.
0270A4	~ANSKey8.1	??? The key 8.1 in alpha mode.

0280A4	~ANSKey8.2	??? The key 8.2 in alpha mode.
0290A4	~ANSKey8.3	??? The key 8.3 in alpha mode.
02A0A4	~ANSKey8.4	??? The key 8.4 in alpha mode.
02B0A4	~ANSKey8.5	??? The key 8.5 in alpha mode.
02C0A4	~ANSKey9.1	??? The key 9.1 in alpha mode.
02D0A4	~ANSKey9.2	??? The key 9.2 in alpha mode.
02E0A4	~ANSKey9.3	??? The key 9.3 in alpha mode.
02F0A4	~ANSKey9.4	??? The key 9.4 in alpha mode.
0300A4	~ANSKey9.5	??? The key 9.5 in alpha mode.
0000A5	~ALSKey1.1	??? The shifted 1.1 key in alpha mode.
0010A5	~ALSKey1.2	??? The shifted 1.2 key in alpha mode.
0020A5	~ALSKey1.3	??? The shifted 1.3 key in alpha mode.
0030A5	~ALSKey1.4	??? The shifted 1.4 key in alpha mode.
0040A5	~ALSKey1.5	??? The shifted 1.5 key in alpha mode.
0050A5	~ALSKey1.6	??? The shifted 1.6 key in alpha mode.
0060A5	~ALSKey2.1	??? The shifted 2.1 key in alpha mode.
0070A5	~ALSKey2.2	??? The shifted 2.2 key in alpha mode.
0080A5	~ALSKey2.3	??? The shifted 2.3 key in alpha mode.
0090A5	~ALSKey2.4	??? The shifted 2.4 key in alpha mode.
00A0A5	~ALSKey2.5	??? The shifted 2.5 key in alpha mode.
00B0A5	~ALSKey2.6	??? The shifted 2.6 key in alpha mode.
0170A5	~ALSKey4.6	??? The shifted 4.6 key in alpha mode.
00C0A5	~ALSKey3.1	??? The shifted 3.1 key in alpha mode.

00D0A5	~ALSKey3.2	??? The shifted 3.2 key in alpha mode.
00E0A5	~ALSKey3.3	??? The shifted 3.3 key in alpha mode.
00F0A5	~ALSKey3.4	??? The shifted 3.4 key in alpha mode.
0100A5	~ALSKey3.5	??? The shifted 3.5 key in alpha mode.
0110A5	~ALSKey3.6	??? The shifted 3.6 key in alpha mode.
0120A5	~ALSKey4.1	??? The shifted 4.1 key in alpha mode.
0130A5	~ALSKey4.2	??? The shifted 4.2 key in alpha mode.
0140A5	~ALSKey4.3	??? The shifted 4.3 key in alpha mode.
0150A5	~ALSKey4.4	??? The shifted 4.4 key in alpha mode.
0160A5	~ALSKey4.5	??? The shifted 4.5 key in alpha mode.
0180A5	~ALSKey5.1	??? The shifted 5.1 key in alpha mode.
0190A5	~ALSKey5.2	??? The shifted 5.2 key in alpha mode.
01A0A5	~ALSKey5.3	??? The shifted 5.3 key in alpha mode.
01B0A5	~ALSKey5.4	??? The shifted 5.4 key in alpha mode.
01C0A5	~ALSKey5.5	??? The shifted 5.5 key in alpha mode.
01D0A5	~ALSKey6.1	??? The shifted 6.1 key in alpha mode.
01E0A5	~ALSKey6.2	??? The shifted 6.2 key in alpha mode.
01F0A5	~ALSKey6.3	??? The shifted 6.3 key in alpha mode.
0200A5	~ALSKey6.4	??? The shifted 6.4 key in alpha mode.
0210A5	~ALSKey6.5	??? The shifted 6.5 key in alpha mode.
0220A5	~ALSKey7.1	??? The shifted 7.1 key in alpha mode.
0230A5	~ALSKey7.2	??? The shifted 7.2 key in alpha mode.
0240A5	~ALSKey7.3	??? The shifted 7.3 key in alpha mode.

0250A5	~ALSKey7.4	??? The shifted 7.4 key in alpha mode.
0260A5	~ALSKey7.5	??? The shifted 7.5 key in alpha mode.
0270A5	~ALSKey8.1	??? The shifted 8.1 key in alpha mode.
0280A5	~ALSKey8.2	??? The shifted 8.2 key in alpha mode.
0290A5	~ALSKey8.3	??? The shifted 8.3 key in alpha mode.
02A0A5	~ALSKey8.4	??? The shifted 8.4 key in alpha mode.
02B0A5	~ALSKey8.5	??? The shifted 8.5 key in alpha mode.
02C0A5	~ALSKey9.1	??? The shifted 9.1 key in alpha mode.
02D0A5	~ALSKey9.2	??? The shifted 9.2 key in alpha mode.
02E0A5	~ALSKey9.3	??? The shifted 9.3 key in alpha mode.
02F0A5	~ALSKey9.4	??? The shifted 9.4 key in alpha mode.
0300A5	~ALSKey9.5	??? The shifted 9.5 key in alpha mode.
0090A1	~MissingKey1	
00B0A1	~MissingKey2	
0090A2	~MissingKey3	
00B0A2	~MissingKey4	

5.5 Labels

81937	INT_00
8193C	INT_01
81941	INT_02
81946	INT_03
8194B	INT_04
81950	INT_05
81955	INT_06
8195A	INT_07
8195F	INT_08
81964	INT_09
81969	INT_0A

```
8196E      INT_0B
81973      INT_0C
81978      INT_0D
8197D      INT_0E
81982      INT_0F
```

5.6 LastBut

```
81DB0      LastBut0
81DB5      LastBut1
81DBA      LastBut2
81DBF      LastBut3
81DC4      LastBut4
81DC9      LastBut5
81DCE      LastBut6
81DD3      LastBut7
81DD8      LastBut8
81DDD      LastBut9
81DE2      LastBut10
81DE7      LastBut11
81DEC      LastBut12
81DF1      LastBut13
81DF6      LastBut14
81DFB      LastBut15
81E00      LastBut16
81E05      LastBut17
81E0A      LastBut18
81E0F      LastBut19
81E14      LastBut20
81E19      LastBut21
81E1E      LastBut22
81E23      LastBut23
81E28      LastBut24
81E2D      LastBut25
81E32      LastBut26
81E37      LastBut27
81E3C      LastBut28
81E41      LastBut29
```

```
81E46     LastBut30
81E4B     LastBut31
81E50     LastBut32
```

5.7 x

```
0620F2   ~xAngle
0590F2   ~xAns
0920F2   ~xArea
0630F2   ~xAxes
0650F2   ~xConnect
06C0F2   ~xCoord
0610F2   ~xDate
0810F2   ~xDigits
0910F2   ~xExtremum
05A0F2   ~xFIT
03A0AB   ~xFmList
03B0AB   ~xFmMat
0800F2   ~xFormat
0640F2   ~xGrid
05B0F2   ~xHAngle
05D0F2   ~xHDigits
05C0F2   ~xHFormat
0740F2   ~xHTick
0660F2   ~xHighRes
09F0F2   ~xHisWidth
0A10F2   ~xHmax
0A00F2   ~xHmin
0760F2   ~xHzoom
05F0F2   ~xIerr
0710F2   ~xIndep
0690F2   ~xInvCursor
0900F2   ~xIsect
06A0F2   ~xLabels
0230F2   ~xMAXS
01B0F2   ~xMEANS
0270F2   ~xMEANX
02A0F2   ~xMEANY
```

```
0240F2    ~xMEDIAN
0220F2    ~xMINS
0210F2    ~xNS
09B0F2    ~xNmax
09A0F2    ~xNmin
0820F2    ~xNoteText
07E0F2    ~xNumCol
07F0F2    ~xNumFont
07B0F2    ~xNumIndep
07D0F2    ~xNumRow
0780F2    ~xNumStart
0790F2    ~xNumStep
07A0F2    ~xNumType
07C0F2    ~xNumZoom
0830F2    ~xPage
0840F2    ~xPageNum
0300F2    ~xRELERR
05E0F2    ~xRadixMark
0680F2    ~xRecenter
08F0F2    ~xRoot
01F0F2    ~xSSDEV
01D0F2    ~xSVARS
0280F2    ~xSX
0290F2    ~xSX2
02D0F2    ~xSXY
02B0F2    ~xSY
02C0F2    ~xSY2
09C0F2    ~xSeqPlot
0670F2    ~xSimult
0930F2    ~xSlope
09D0F2    ~xStatMode
09E0F2    ~xStatPlot
01C0F2    ~xTOTS
0960F2    ~xTStep
0980F2    ~xThetaMax
0970F2    ~xThetaMin
0990F2    ~xThetaStep
0600F2    ~xTime
```

```
0950F2    ~xTmax
0940F2    ~xTmin
06B0F2    ~xTracing
0750F2    ~xVTick
0770F2    ~xVzoom
0720F2    ~xXcross
06E0F2    ~xXmax
06D0F2    ~xXmin
0730F2    ~xYcross
0700F2    ~xYmax
06F0F2    ~xYmin
0150AB    ~x>COL
0170AB    ~x>DIAG
0B7002    ~x>DISPLAY
0B9002    ~x>PLOT
0130AB    ~x>ROW
0570AB    ~xACOT
0580AB    ~xACSC
0590AB    ~xASEC
0BA002    ~xBLANKGROB
0470AB    ~xBOXW
03B701    ~xCHECK
0450AB    ~xCOBWEB
0160AB    ~xCOL>
0370AB    ~xCONCAT
0540AB    ~xCOT
0560AB    ~xCSC
01E0AB    ~xCSWP
04D0AB    ~xCUBICFIT
0280F1    ~xDFLTNOTE
0290F1    ~xDFLTPICT
0180AB    ~xDIAG>
0B6002    ~xDISPLAY>
0A6002    ~xERASEA
0430AB    ~xFRACTION
0210F1    ~xFUNCSYMB
0080F1    ~xFUNCTAB
0460AB    ~xHIST
```

```
04E0AB    ~xLOGISFIT
0530AB    ~xMKMAT
0230F1    ~xPARAMSYMB
00A0F1    ~xPARAMTAB
0B8002    ~xPLOT>
0220F1    ~xPOLARSYMB
0090F1    ~xPOLARTAB
05B0AB    ~xPOLYFORM
04B0AB    ~xPOWERFIT
05C0AB    ~xPoly
04C0AB    ~xQUADFIT
0320F1    ~xREADNOTE
0330F1    ~xREADPICT
05A0AB    ~xRECURSE
0140AB    ~xROW>
0550AB    ~xSEC
0240F1    ~xSEQSYMB
00B0F1    ~xSEQTAB
0270F1    ~xSOLVESYMB
00E0F1    ~xSOLVETAB
0440AB    ~xSTAIRSTEP
0260F1    ~xSTAT2SYMB
00D0F1    ~xSTAT2TAB
0250F1    ~xSTATSYMB
00C0F1    ~xSTATTAB
0500AB    ~xStat1Var
0510AB    ~xStat2Var
03C701    ~xUNCHECK
04F0AB    ~xUSERFIT
0340F1    ~xUndefined
0330AB    ~x\85LIST
02F0AB    ~x\9BLIST
0350AB    ~x\9CLIST
014701    ~xx>DISPLAY
017701    ~xx>GROB
016701    ~xx>PLOT
018701    ~xxARC
003701    ~xxBEEP
```

```
00F701    ~xxBOX
01D701    ~xxBREAK
03F701    ~xxCHOOSE
023701    ~xxCOL+
022701    ~xxCOL-
025701    ~xxCSWP
03C0AB    ~xxDEG
038701    ~xxDEMO
002701    ~xxDISP
013701    ~xxDISPLAY>
031701    ~xxDO1VSTATS
034701    ~xxDO2VSTATS
009701    ~xxERASE
00A701    ~xxERASEPLOT
01F701    ~xxFREEZE
040701    ~xxGETKEY
010701    ~xxGOR
03E0AB    ~xxGRAD
012701    ~xxGROBNOT
011701    ~xxGXOR
041701    ~xxHELP
01C701    ~xxINPUT
0520AB    ~xxITERATE
02D701    ~xxLIBEVAL
00D701    ~xxLINE
02A701    ~xxMATEDIT
02B701    ~xxMKGROB
03E701    ~xxMSGBOX
00C701    ~xxPIXOFF
00B701    ~xxPIXON
015701    ~xxPLOT>
086002    ~xxPOS
01B701    ~xxPRDISPLAY
019701    ~xxPRSTC
01A701    ~xxPRVAR
03D0AB    ~xxRAD
036701    ~xxRANM
026701    ~xxRCI
```

```
027701    ~xxRCIJ
008701    ~xxRDM
005701    ~xxRDZ
007701    ~xxREPL
021701    ~xxROW+
020701    ~xxROW-
024701    ~xxRSWP
039701    ~xxRULES
029701    ~xxRUNPGM
035701    ~xxSELECT
033701    ~xxSETDEPEND
030701    ~xxSETFREQ
032701    ~xxSETINDEP
02F701    ~xxSETSAMPLE
03A701    ~xxSETVIEWS
085002    ~xxSIZE
01E701    ~xxSTOP
006701    ~xxSUB
02E701    ~xxSYSEVAL
00E701    ~xxTLINE
028701    ~xxTO
001701    ~xxTSTR
037701    ~xxVERSION
004701    ~xxWAIT
000701    ~xxWSLOG
02C701    ~xxZEROGROB
```

5.8 Rest

```
81CE8      !>ARRAY
BA700      #+#2-
829E5      #=Lookup
818F1      #DIV
8042E      #Error:
818F6      #MOD
818EC      #NEG
813A6      #NOT
81905      #ODD
```

```
813AB      #OR
803E8      #ObTypeBase
80424      #Warning:NL
813B0      #XOR
A4CEF      %%-1
8307F      &&
84A2E      1GETLAM#0=
82E59      1GETapndcpl
8253F      2CDispList
825DF      2CKeyOK
82571      2Col?Case2Col
82535      2ColChoose
80046      2STR
826ED      ?AdjFocusPos
82639      ?DispMoreU/D
829BD      ?FixFieldKeys
8289B      ?GetFObTypes
82BF7      ?NoTaskSwDef
81CAC      ABSCOERCE
800DC      ARRYCMP
800FA      ARRYLIST
8012C      ARRYSYM
82143      AddEq$
8214D      AlDrawMenu
81A7C      AllowPRLCD
BD42C      Angle@
832C3      AngleField
832B9      AngleLabel
8211B      AnyDABad?
833BD      ApName$>Id
833B3      ApNameId>$
8339F      ApNameId>Id
81DA6      AppDir+Offs
BDF75      Area@
81EB4      AttnPOSCOMP
81711      BAK>HOME
81D79      BREAK
BD8BE      BTRow1@
```

```
3949A      BadIfEdit
8199B      BinLookup
8291D      BlankClient
82927      BlankHelp
81E5A      Bubble
81987      CAND
01C0F1    ~CLRTOPICLAM
81C1B      COMPROMID
8198C      COR
81DA1      CalcDir+Offs
82DFF      CatNot
82DF5      CatPgm
82DEB      CatThisCxt
829C7      Choose&DoTask
833EF      ChooseVEntry
833F9      ChooseVExit
83403      ChooseViewUI
81DAB      Contxt+Offs
81B30      CopyRegCOb
8123D      DOINDIR
821E3      DOLCD>g0
82FD5      DOROOT
821ED      DOSHOWIT1
821F7      DOSHOWIT1C
82C01      DOWAIT
8287D      DecompNoNL
82869      DecompOb
82189      DefauPtXit
82193      DefauTrcInit
7B7D8      DefaultHint
E03A1      DemoFrames
BD940      Digits@
9165A      DispMsgBox
8219D      DispXFunc
821A7      DispYFunc
82909      DoAlert&Query
833C7      DoApletLib
8257B      DoAs2Col
```

83313	DoCApNoteV	
832F5	DoCApPlotSV	
832D7	DoCApPlotV	
8331D	DoCApSketchV	
83309	DoCApSymbSV	
832EB	DoCApSymbV	
832FF	DoCApTableSV	
832E1	DoCApTableV	
832CD	DoCurrAplet	
821C5	DoDemo	
825E9	DoDispBorder	
82783	DoDispField	
825FD	DoDispList	
825F3	DoDispPrompt	
8250D	DoEditLCancel	
82517	DoEditLOK	
824F9	DoEditLine	
82AFD	DoGetObFrSto	
82B2F	DoIOErrAlert	
82B25	DoIOStatusBox	
82E09	DoInApLbCxt	
82E1D	DoInNotCxt	
82E13	DoInPgmCxt	
827E7	DoKeyCheck	
827D3	DoKeyChoos/Ck	
827DD	DoKeyChoose	
828C3	DoMatEdit	
828FF	DoQueryBox	
82AE9	DoRecv/GetOb	(id →) Receive an object. FIXME: Stack diagram not confirmed.
82AF3	DoRecvObFrEls	
82611	DoRestCovWin	
829F9	DoResultTab	
83039	DoRomPtrKey	
82607	DoSaveCovWin	
82AAD	DoSendOb	(ob →) Send an object. FIXME: Stack diagram not confirmed.

```
82ACB     DoSendObToCDi
82AD5     DoSendObToDir
82AC1     DoSendObToSto
82913     DoSpecAlert
67B3E     Docrunchc
829DB     Done?CkNoNul:
25DF8     DropJunk
7D3ED     DummyMenuErr
82B39     ElsieGet
82B4D     ElsiePkt
82B43     ElsieSend
82689     EmptyList?
82585     EmptyRList?
7D46A     EnsureMenuOff
834FD     EnterGraphView
834F3     EnterTextView
2B7F5     EraseGraph
82201     EraseGrob
803D4     Err#Chr00
82959     EvalPart1
82963     EvalPart2
8296D     EvalPart3
BDFBB     Extremum@
82981     FAreaBad?
82E63     FOURpsh
803CA     FSTFLOATROM#
82AA3     FTypeCheck?
82A99     FTypeFixedL?
82A8F     FTypeList?
82A85     FTypeText?
82215     FUNCPLOT
82837     FitLeftSmF
82841     FitRightSmF
BD8FF     Format@
8220B     FuncPLoop
83327     FuncPSetup1
83331     FuncSplitViewers
81ABD     GDISP?
```

```
8225B     GROBBUFF!
8226F     GROBVIEW
82DD7     GetBVars.1
828AF     GetChoiceFmt
828A5     GetChoiceList
82675     GetDASpecFlag
8302F     GetElNoRomp
83395     GetLastNotThis
82855     GetNextFId
82B1B     GetObType$
82B11     GetObTypePr$
8293B     GetPart1
82945     GetPart2
8294F     GetPart3
8285F     GetPrevFId
6E8C2     GraphTableUI
6E908     GraphZoomUI
82931     GtoField
82DC3     HOME>BAK
BD328     HTick@
BD3AA     HZoom@
C0383     HisWidth@
C040F     Hmax@
C03C9     Hmin@
827AB     IFECheck
827BF     IFEChoosByChr
827B5     IFEChooseNext
8275B     IFEDispClient
827A1     IFEDispHelp
8276F     IFEDispLabel
82765     IFEDispTitle
8278D     IFEUnShowSel
82751     IFEdLineMenu
82747     IFMenu
833A9     Id>ApNameId
83467     InAplet?
81D97     InApletF?
82567     Init2ColMets
```

```
82279     InitIndep
8255D     InitListMets
82283     Init_window
833DB     InpFormVEntry
833E5     InpFormVExit
833D1     InpFormViewUI
82797     InvertField
8282D     IsIFMenu?
82891     IsNullField?
82C51     IsTaskSwKey?
BDF2F     Isect@
82977     KeyFace
81E55     LASTBUTN
8269D     LEDispBorder
826CF     LESetIDecomp
826C5     LESetItem
826D9     LESetRowWidth
80AF4     LF$
82625     LHighlight
80528     LSTBIMACROM#
8262F     LUnHighlight
BD530     LastEq@
BD4EF     LastIndep@
BD46D     LastX@
BD4AE     LastY@
83511     LeaveGraphView
83507     LeaveTextView
826E3     ListBar
81F90     ListNames
BE001     ListOfEqs@
8315B     M+prep
8338B     Make1stAplet
83449     MakeNoteVTtl
8342B     MakeNumSVTtl
83421     MakeNumVTtl
83417     MakePlotSVTtl
8340D     MakePlotVTtl
83453     MakeSketVTtl
```

```
8343F     MakeSymbSVTtl
83435     MakeSymbVTtl
829B3     MakeTitleBar
8345D     MakeViewTitle
82D37     MemStoAns
7D42E     MiniVar
8264D     MoreDown?
8261B     MoreU/D?
82643     MoreUp?
81D33     NULLargcase
81FC7     NUMEVAL1
83381     NextApOfType
828EB     NextTextLine
818DD     NotAndbitmap
82DCD     OB>BAK
81518     OVEREQcase
818D3     Orbitmap
82CBF     POLSetUIExt
8171B     PORT0ADDR
421AF     PageDnNGrob
4216B     PageDnYGrob
42127     PageUpNGrob
420E3     PageUpYGrob
3D3F6     ParseExpr
822DD     PickedEqs@
829D1     PlaceFVals
BD5B2     Plot2Flg@
BD5F3     PlotCKSUM@
BD571     PlotFlag@
83237     PlotOvViewUI
83219     PlotViewUI
82D41     Port0Ids
8333B     PrepCurrAplet
82521     PrepEdLKeyOb
82B6B     PrintHist
82B89     PrintLcd
82B75     PrintVar
82B7F     PrintVarOb
```

```
83359     PurgeAplet
83363     PurgeApletOb
818E2     Put3x5
818E7     Put5x7
8267F     PutDASpecFlag
818CE     Putbitmap
8049C     REALLISTREAL
8044C     REALSTRID
81E82     REDIMPREP
81B2B     RPICK
83377     RclAplet
8336D     RclAplet?Err
82887     RclFieldVal
828B9     RclResetVal
C0527     ReCalcFlag@
831FB     Replace_List
BDEA3     Root@
82B61     SEFINISH
62D36     SETPLOTENV
99EF3     STArryList
9C0B9     STBigKeyDef
9A7B8     STDoMedium
9A03D     STDoSmall
9BFA8     STEditKeyDef
97C29     STErrorGrob
99E30     STInitCols
9BFD8     STInsKeyDef
9B5E8     STJump
9B6BA     STJumpN
BD0B7     STO_tTYPE
9C4B2     STSortCol
9C1FA     STStatsKeyDef
9A024     STTableDisp
99FFC     STTableExit
9AEBB     STTableHKeys
99D95     STTableInit
80F74     STypeAnyChr
80F2E     STypeDirChr
```

```
80F5F     STypeLibChr
80F51     STypeLstChr
80F58     STypeMatChr
80F3C     STypeNteChr
80F43     STypePgmChr
80F66     STypeTgtChr
80F4A     STypeVarChr
80F6D     STypeVecChr
80F35     STypeZapChr
81D9C     SV_?defined
83471     SV_actual
8347B     SV_calledit
83485     SV_getbody
8348F     SV_keycancel
83499     SV_keycheck
834A3     SV_keyedit
834AD     SV_keyeval
834B7     SV_keyok
834C1     SV_keyshow
834CB     SV_setgrob19
834D5     SV_setitem0
C02F7     SVarType@
9C2CC     SWAPStatFlag
8030C     SYMSTR
81FE5     SafeCrunch%
831A1     SafeDecomp
82553     SaveListMets
82823     Sel&DispNextF
82805     SelNextField
8280F     SelPrevField
83345     SelectAplet
C6B6B     Seq_eval0
C6BC5     Seq_eval1
C6E45     Seq_eval2
C6FEE     Seq_evaln
820FD     SetDA2Temp
82125     SetDAsBad
82995     SetFAreaBad
```

```
8298B     SetFAreaOK
829A9     SetFAreasBad
8299F     SetFAreasOK
81D8D     SetInAplet
64184     SetRedrawFlag
8234B     Set_window
BDEE9     Slope@
BD634     Split1CKSUM@
BD675     Split2CKSUM@
BD87D     SplitRow@
82355     SplitTraceInit
C04E1     Stat2Flag@
C061C     StatFit@
C056D     StatFont@
C0455     StatMark@
C0586     StatMisc[]@
C049B     StatModel@
C033D     StatPlot@
C0298     StatType!
C02B1     StatType@
0190F1   ~StdApEntry
82873     StdDecompNoNL
82B07     StdIOProc
83223     StdTableViewUI
82369     StndXYCoord?
82D69     StoAns@
82D73     StoAns@Drp
8334F     StoAplet
831AB     StripTicks
82549     SwapL/RMets
8322D     SymbViewUI
82373     TBColL3x5
8237D     TBColL5x7
82387     TBColR3x5
82391     TBColR5x7
8239B     TBDrawF3x5
823A5     TBDrawF5x7
82503     TBEdSfKeys
```

```
823AF      TBErrorGrob
823B9      TBFormat
9216B      TBFuncDecomp
823C3      TBInv3x5C0
823CD      TBInv3x5C1
823D7      TBInv3x5C2
823E1      TBInv3x5C3
823EB      TBInv5x7C0
823F5      TBInv5x7C1
823FF      TBInv5x7C2
82409      TBRollD3x5
82413      TBRollD5x7
8241D      TBRollU3x5
82427      TBRollU5x7
83241      TBStdZooms
8324B      TBStndSplitInfo
BD83C      TCol1@
82431      TDDat3x5C1
8243B      TDDat3x5C2
82445      TDDat3x5C3
8244F      TDDat3x5C4
82459      TDDat3x5C5
82463      TDDatD3x5
8246D      TDDatL3x5
82477      TDDatR3x5
82481      TDDatU3x5
8248B      TDDataF3x5
BD779      TFlags@
041A7      THIRTYSIX
80050      THREEFOUR
BD6B6      TStart@
BD6F7      TStep@
BD738      TZoom@
83255      TablSetViewUI
BD7FB      Tcol@
82CFB      TempMenuBuff
```

Makes some new menus that surely comes from METAKERNEL FIXME: What is meant by this?

```
82495     TraceY
BD7BA     Trow@
824BD     UnpackGrob
5CAD5     Update_tTYPE
82D7D     UserSto
80474     VLMAlarmMsg
8046A     VLMUserKeys
80488     VLMcmdlmsg
80460     VLMhistmsg
8047E     VLMlastargs
80442     VLMmsg
80438     VLMpurgemsg
8044C     VLMstkmsg
80456     VLMundomsg
BD369     VTick@
BD3EB     VZoom@
8284B     WidthSmF
82DAF     XPURGEp
82DB9     XPURGEp0
82D9B     XRCLp
095A5     XRCLp0
82D87     XSTOp
82D91     XSTOp0
BD14D     Xmax!
BD265     Xmax2@
BD161     Xmax@
BD10C     Xmin!
BD224     Xmin2@
BD120     Xmin@
818D8     Xorbitmap
BD2E7     Ymax2@
BD1E3     Ymax@
BD2A6     Ymin2@
BD1A2     Ymin@
6E94E     ZoomAutoUI
831F1     alg=
82E9F     apndit
8310B     argswap
```

```
83101     argswap&&
83115     argswapnext
81FA4     blackbox
8216B     check_xrange
82175     check_yrange
82EA9     chooselst
82EB3     completed
8191E     d*
8190F     d<
81919     d<=
8190A     d>
81932     d>%
81914     d>=
81923     dDIV
81928     dMOD
0BCEE     dirstrucchk
8192D     dmuldiv
852EC     doptr!
852D0     doptr@
82EBD     dvbind
82EC7     elsiename
82EDB     filename
BE03D     funcCache@
82A17     grobCheck
82A21     grobCheckX
82A2B     grobInvChk
82A35     grobInvChkX
82A53     grobInvNoChk
82A49     grobInvUnChkX
82661     grobMoreDown
82657     grobMoreUp
8266B     grobNoMore
826F7     grobPOBox2
82701     grobPOBox3
8270B     grobPOBox4
82715     grobPOBox5
8271F     grobPOBoxP2
82729     grobPOBoxP3
```

```
82733     grobPOBoxP4
82A71     grobQueryIcon
82A7B     grobTitleBar
82A3F     grobUnCheckX
80208     idntany
80000     illnameerr
8024E     lamany
802B2     library
0BD53     loopdirck
83057     metaDUP
83129     metainsert
82EE5     newsymbPA
8311F     numargs
82EEF     oneexpr
82EF9     onestring
82F03     optfilename
82B93     prx1
8306B     psh1
83093     psharg
83089     pshmonop
80A0C     rbrac
82E31     realPA
83043     resolved
81D0B     roll3RD
81D1F     roll4TH
81D1A     rollNTH
8001E     rrp
82319     s1>
82E3B     solverTTT
83205     startSQFORM
830E3     syminner&
830F7     syminner&N
830ED     syminnertwo
83025     tTYPE@
82F0D     threeexprs
82F17     tok;triand
82F21     toktriand
834DF     topic_install
```

```
834E9     topic_uninstall
82F2B     twoexprs
82F35     twostrings
81D15     unroll3RD
81D24     unroll4TH
81D29     unrollNTH
830D9     unsyminner
83147     unsymone
83133     verysyminner
8313D     veryunsymin
83151     vunsymfcn
835C5     {NoteText}
835CF     {NumVars}
835D9     {SketchSet}
0660AB   ~{StatVars}
0060E8   ~::args
0130E8   ~?NULLSETDIM
02C0A1   ~AttentionKe
0830F0   ~CHECKEXISTS
08F0F0   ~COLCTDER
0940F0   ~COLCTFCNAP
0920F0   ~COLCTIFTE
0900F0   ~COLCTINTG
0930F0   ~COLCTQUOTE
0910F0   ~COLCTSUM
0DA002   ~COMPLEXDUMM
0340AB   ~ChkDaList
0100A1   ~DArrow
0170F1   ~DefauStat2T
0160F1   ~DefauStatTy
01A0A2   ~DoExponent
0000A1   ~DoMenuKey1N
0010A1   ~DoMenuKey2N
0020A1   ~DoMenuKey3N
0030A1   ~DoMenuKey4N
0040A1   ~DoMenuKey5N
0050A1   ~DoMenuKey6N
01F0E8   ~DoNumeric:
```

```
0180A1    ~Enter/Again
0100A2    ~FarDArrow
00F0A2    ~FarLArrow
0110A2    ~FarRArrow
00A0A2    ~FarUArrow
01D0F1    ~GETPLTLABLS
0150F1    ~GetTypeText
0010F0    ~I:-InvSin
0090F0    ~I:-InvSinh
0170F0    ~I:Acosh
0160F0    ~I:Asinh
0130F0    ~I:Atan
0030F0    ~I:LnTan
00A0F0    ~I:LnTanh
00C0F0    ~I:Tanh
0060F0    ~INTGACOS
00F0F0    ~INTGALOG
0050F0    ~INTGASIN
0070F0    ~INTGATAN
0020F0    ~INTGCOS
00B0F0    ~INTGCOSH
0190F0    ~INTGDER
00E0F0    ~INTGEXPM
0120F0    ~INTGINV
0100F0    ~INTGLN
0110F0    ~INTGLOG
0180F0    ~INTGSIGN
0000F0    ~INTGSIN
0080F0    ~INTGSINH
0140F0    ~INTGSQ
0150F0    ~INTGSQRT
0040F0    ~INTGTAN
00D0F0    ~INTGTANH
00F0A1    ~LArrow
0300E8    ~LIXRecv
0310E8    ~LIXSend
0330C2    ~MAKEEPIDN
0000B9    ~MiscIFMsg
```

```
0210E8   ~NotHidden
01E0F1   ~POINTERR
01F0F1   ~POINTEXIT
0DB002   ~POLARDUMMY
0000BE   ~PlotIFMsg
0060A1   ~PlotViewKey
0080C0   ~PolyNSymbF
0070C0   ~PolyNSymbP
0110A1   ~RArrow
0620F0   ~REIM[]*
01B0F1   ~SETTOPICLAM
01A0F1   ~SolveApEntr
0000BF   ~SolveIFMsg
0000BB   ~StatIFMsg
0070A1   ~SymbViewKey
0080A1   ~TableViewKe
00C0A1   ~TopicLibKey
00A0A1   ~UpArrow
0200E8   ~UseHidden{}
0780F0   ~WHEREDER
0770F0   ~WHEREFCNAPP
0790F0   ~WHEREIFTE
07C0F0   ~WHEREINTG
07D0F0   ~WHERESUM
07A0F0   ~WHEREWHERE
0210AB   ~XEQDIAG>L
0220AB   ~XEQDIAG>R
0230AB   ~XEQRANM
0020AB   ~XEQSYMLIN
0810F0   ~XEQXDPTCH
08A0F0   ~XEVALp*
0890F0   ~XEVALp?
08C0F0   ~XPURGEp*
08B0F0   ~XPURGEp?
0880F0   ~XRCLp*
0850F0   ~XRCLp?
08E0F0   ~XRCLp?acc>
0870F0   ~XRCLpL
```

```
0860F0    ~XRCLpNL
07E0F0    ~XSTOCHECK
07F0F0    ~XSTOCHECK10
0820F0    ~Xcont
04D0F0    ~adjdiv
04B0F0    ~adjdivsign
04C0F0    ~adjsign
02E0E8    ~arryspec
06B0AB    ~bad^
08D0F0    ~convertaddr
02D0E8    ~copysub
0960F0    ~covD/DCROSS
0970F0    ~covD/DDOT
0B90F0    ~covD/DINTG
0C90F0    ~covINV*
0C60F0    ~covINV+
0C70F0    ~covINV-
0CA0F0    ~covINV/
0C80F0    ~covINV=
0D40F0    ~covINVALOG
0CF0F0    ~covINVCOS
0D20F0    ~covINVCOSH
0CD0F0    ~covINVEXP
0D50F0    ~covINVEXPM1
0CE0F0    ~covINVSIN
0D10F0    ~covINVSINH
0D00F0    ~covINVTAN
0D30F0    ~covINVTANH
0CB0F0    ~covINV^
0CC0F0    ~covINV^X
0800F0    ~covLBSTO
0C50F0    ~covMANATG
0C10F0    ~covMANCSIV
0BF0F0    ~covMANEXP
0BD0F0    ~covMANMENU*
0BC0F0    ~covMANMENU+
0C30F0    ~covMANMENUC
0C20F0    ~covMANMENUE
```

0C00F0	~covMANMENUL	
0C40F0	~covMANTRG	
07B0F0	~covWSPLIT	
0760F0	~covmanCOL	
0840F0	~covmetaLIBS	
00D0E8	~docmdlist	
0040E8	~doidseqn	
00F0E8	~dolatorre	
0110E8	~dolatorre+	
0120E8	~dolatorre2	
0190E8	~dolist+	
0020E8	~dontuple	
0030E8	~dontuple#	Internal DOSUBS (comp ob # → {}) Takes objects from comp in groups of # and evals ob on them. The results are returned as a list.
05F0AB	~dopcoeff	
05E0AB	~dopolydiv	
05D0AB	~dopolyz	
00A0E8	~doptrseqn	
06D0AB	~dopval	
0010E8	~dosecntuple	
0080E8	~dosecseqn	
0000E8	~doseq	
00B0E8	~doseqn	
00C0E8	~doseqn#	
01A0E8	~dosort	
0100E8	~elsielists?	
02F0E8	~etorc	
0250C2	~lauserQR	
0160E8	~lolatorre	
0150E8	~lslatorre	
0600AB	~metapolyz	
0240E8	~morerepl	
0230E8	~moresub	
0260E8	~mrepll	
0270E8	~mrepll+	
0250E8	~mrepln	
02A0E8	~msubll	

```
02B0E8   ~msubll+
0280E8   ~msubln
0290E8   ~msubnl
02C0E8   ~msubnn
0070F1   ~nBOXWHISKER
0000F1   ~nFUNCTION
00F0F1   ~nFUNCTOPIC
0050F1   ~nHISTOGRAM
0030F1   ~nPARAMETRIC
0110F1   ~nPARAMTOPIC
0020F1   ~nPOLAR
0100F1   ~nPOLARTOPIC
0180F1   ~nPTYPE>PINF
0040F1   ~nSCATTER
0120F1   ~nSEQTOPIC
0060F1   ~nSEQUENCE
0010F1   ~nSOLVE
0140F1   ~nSOLVETOPIC
0130F1   ~nSTATTOPIC
0680AB   ~ns*spec
0170E8   ~ollatorre
0180E8   ~ollatorre+
0040C0   ~pcoef
0050C0   ~pcoefacc1
0060C0   ~pcoefacc2
06E0AB   ~pdiv
0030C0   ~peval
0640AB   ~poly*
0620AB   ~poly+
0700AB   ~poly/
06A0AB   ~poly^
06C0AB   ~poly^#
0630AB   ~polyneg
0610AB   ~polyprep
0010C0   ~proot
0000C0   ~proot_c
0020C0   ~proot_r
0070E8   ~ptrargs
```

```
0310C2    ~rMAKEPCOPY
0320C2    ~rPACKARRYD
06F0AB    ~resymb
0D7002    ~rpnAPPLY
0CE002    ~rpnDER
0D1002    ~rpnINTG
0D4002    ~rpnWHERE
0200F1    ~runalias?
0050E8    ~seqid
00E0E8    ~seqnargs
0140E8    ~sllatorre
0670AB    ~sn*spec
0690AB    ~ss*spec
0310F1    ~topic_CAPLE
02A0F1    ~topic_NONE
0090E8    ~udfargs
02C0F1    ~view01_CAPL
02D0F1    ~view23_CAPL
02E0F1    ~view45_CAPL
02F0F1    ~view6_CAPLE
0300F1    ~view7_CAPLE
02B0F1    ~view_NONE
01D0E8    ~{$}>{id}
01E0E8    ~{$}>{lam}
01B0E8    ~{id}>{$}
01C0E8    ~{lam}>{$}
83165      APPEND_SPC
81A77      AlowPrlcdCl
82AB7      DoSendObToEl
82DE1      GetApDirLst
```

6 UserRPL Commands

6.1 A-F

0270AB ~xCHOOSE
(title {elems} pos → ob 1)
(title {elems} pos → 0)
User-Defined Choose Box Cmd
--
Creates a user-defined choose box
--
Related: INFORM,NOVAL

01C0AB ~xCOL+
([[]] [[]]' n → [[]]'')
([] x n → []')
Insert Column Cmd
--
Inserts an array (vector or matrix) into a matrix (one or more elements into a vector) at the position indicated by nindex, and returns the modified array.
--
[[mat]]1 [mat]2 nidx → [[mat]]3
[[mat]]1 [vec]col nidx → [[mat]]2
[vec]1 nelement nidx → [vec]2
--
Related: COL-,CSWP,ROW+,ROW-

01B0AB ~xCOL-
([] n → []' xn)
([[]] n → [[]]' [vn])
Delete Column Cmd
--
Deletes column n of a matrix (or element n of a vector), and returns the modified matrix (or vector) and the deleted column (or element).
--
Related: COL+,CSWP,ROW+,ROW-

0030AB ~xCOND
([[n*n]] → x)
Conditional Number Cmd
--
Returns the 1-norm (column norm) condition number of a square matrix.
--
Related: SNRM,SRAD,TRACE

01E0AB ~xCSWP

0360AB ~xDOLIST

({1}...{n} n prog → {})
({1}...{n} prog → {} (n=1))
Do to List Cmd
--
Applies commands, programs, or user-defined functions to lists.
--
{lst}1 ...{lst}n n ≪prog≫ → {res}
{lst}1 ...{lst}n n cmd → {res}
{lst}1 ...{lst}n n name → {res}
{lst}1 ...{lst}n ≪prog≫ → {res}
{lst}1 ...{lst}n cmd → {res}
{lst}1 ...{lst}n name → {res}
--
Related: DOSUBS,ENDSUB,NSUB,STREAM

02E0AB ~xDOSUBS

({} n prog → {}')
({} prog → {}' (n=1))
Do to Sublist Cmd
--
Applies a program or command to groups of elements in a list.
--
{list}1 n ≪prog≫ → {list}2
{list}1 n command → {list}2
{list}1 n name → {list}2
{list}1 ≪prog≫ → {list}2
{list}1 command → {list}2
{list}1 name → {list}2
--
Related: DOLIST,ENDSUB,NSUB,STREAM

0090AB ~xEGV

([[]] → [[evect]]' [evals])
Eigenvalues and Eigenvectors Command
--
Computes the eigenvalues and right eigenvectors for a square matrix.
--
Related: EGVL

00A0AB ~xEGVL

([[]] → [egval])
Eigenvalues Cmd
--
Computes the eigenvalues of a square matrix.
--
Related: EGV

0310AB ~xENDSUB
(→ x)
Ending Sublist Cmd
--
Provides a way to access the total number of sublists contained in the list used by DOSUBS.
--
Related: DOSUBS,NSUB

81FC2 xEVAL
(ob → ?)
Evaluate Object Cmd
--
Evaluates the object.
--
obj → (see below)
Obj. Type Effects of Evaluation Local Name Recalls the contents of the variable. Global Name Calls the contents of the variable: ; A name is evaluated. A program is evaluated. A directory becomes the current directory. Other objects are put on the stack. If no variable exists for a given name, evaluating the name returns the name to the stack. Program. Enters each object in the program: Names are evaluated (unless quoted). ed). Cmds are evaluated. Other objects are put on the stack. List Enters each object in the list: Names are evaluated. Cmds are evaluated. Programs are evaluated. Other objects are put on the stack. Tagged If the tag specifies a port, recalls and evaluates the specified object. Otherwise, puts the untagged object on the stack. Algebraic Enters each object in the algebraic expression: Names are evaluated. Cmds are evaluated. Other objects are put on the stack. Cmd, Func, XLIB Name Evaluates the specified object. Other Objects Puts the object on the stack.
--
Related: →NUM,SYSEVAL

6.2 G-M

02B0AB ~xHEAD
({} → ob)
($ → $')
First Listed Element Cmd
--
Returns the first element of a list or string.
--
Related: TAIL

0010AB ~xLININ
(symb var → 0/1)
Linear Test Func
--
Tests whether an algebraic is structurally linear for a given variable.

00F0AB ~xLQ
([[]] → [[L]] [[Q]] [[P]])
LQ Factorization of a Matrix Cmd
--
Returns the LQ factorization of an nm matrix.
--
Related: LSQ,QR

0080AB ~xLSQ
([B] [[A]] → []')
([[B]] [[A]] → [[]]')
Least Squares Solution Cmd
--
Returns the minimum norm least squares solution to any system of linear equations where A X = B
--
Related: LQ,RANK,QR,/

00D0AB ~xLU
([[]] → [[L]] [[U]] [[P]])
LU Dec of a Sq. Matrix Cmd
--
Returns the LU decomposition of a square matrix.
--
Related: DET,INV,LSQ,/

0280AB ~xMSGBOX
($ →)
Message Box Cmd
--
Creates a user-defined message box.
--
Related: CHOOSE,INFORM,PROMPT

6.3 N-S

0300AB ~xNSUB
(→ npos)
Number of Sublist Cmd
--
Provides a way to access the current sublist position during an iteration of a program or command applied using DOSUBS.
--
Related: DOSUBS,ENDSUB

0250AB ~xPCOEF
([roots] → [coefs])
Monic Polynomial Coefficients Cmd
--
Returns the coefficients of a monic polynomial (a polynomial with a leading coefficient of 1) having specific roots.
--
Related: PEVAL,PROOT

0260AB ~xPEVAL
([coefs] x → x')
Polynomial Evaluation Cmd
--
Evaluates an n-degree polynomial at x.
--
Related: PCOEFF,PROOT

03D701 ~xPINIT
(→)
Port Initialize Cmd
--
Initializes all currently active ports. Does not affect data already stored in a port.

0240AB ~xPROOT
([coefs] → [roots])
Polynomial Roots Cmd
--
Returns all roots of an n-degree polynomial having real or complex roots.
--
Related: PCOEFF,PEVAL

0200F2 ~xPSDEV
(→ xpsdev)
(→ {x1...xn})
Population Standard Deviation Cmd
--
Calculates the population standard deviation of each of the m columns of coordiate values in the current stastics matrix (reserved variable ΣDAT).
--
<REF>TEXT:Reserved|ΣDAT
--
Related: MEAN,PCOV,PVAR,SDEV,TOT,VAR

00E0AB ~xQR
([[]] → [[Q]] [[R]] [[P]])
QR Factorization of a Matrix Cmd
--
Returns the QR factorization of an nm matrix.
--
Related: LQ,LSQ

0070AB ~xRANK
([[]] → n)
Matrix Rank Cmd
--
Returns the rank of a rectangular matrix.
--
Related: LQ,LSQ,QR

0120AB ~xRANM
({m n} → [[]])
([[]] → [[]]')
Random Matrix Cmd
--
Returns a matrix of specified dimensions that contains random integers in the range -9 through 9.
--
{ m n } → [[rand mat]]mn
[[mat]]mn → [[rand mat]]mn
--
Related: RAND,RDZ

01F0AB ~xRCI
([[]] x nrow → [[]]')
[] x n []'
Multiply Row by Constant Cmd
--
Multiplies row n of a matrix (or element n of a vector) by a const x, and returns the modified matrix.
--
Related: RCIJ

0200AB ~xRCIJ
([[]] x n* n+ → [[]]')
([] x n* n+ → []')
Add Multiplied Row Cmd
--
Multiplies row n* of a matrix by a constant x, adds this product to row n+ of the matrix, and returns the modified matrix. Or, multiplies element n* of a vector by a constant x, adds this product to element n+ of the vector, and returns the modified vector.
--
Related: RCI

0380AB ~xREVLIST
({1...n} → {n...1}')
Reverse List Cmd
--
Reverses the order of the elements in a list.
--
Related: SORT

0190AB ~xROW-
([[]] nrow → [[]]' [])
([] n → []' elt)
Delete Row Cmd
--
Deletes row n of a matrix (or element n of a vector), and returns the modified matrix (or vector) and the deleted row (or element).
--
Related: COL-,COL+,ROW-,RSWP

01A0AB ~xROW+
([[]] [[]]' n → [[]]'')
([[]] [] n → [[]]')
([] n n' → [])
Insert Row Cmd
--
Inserts an array into a matrix (or one or more numbers into a vector) at a position indicated by nindex, and returns the modified matrix (or vector).
--
Related: COL-,COL+,ROW-,RSWP

0110AB ~xRREF
([[]] → [[]]')
Reduced Row Echelon Form Cmd
--
Converts a rectangular matrix to a reduced row echelon form.

01D0AB ~xRSWP
([]/[[]] i j → []/[[]])
Row Swap Cmd
--
Swaps rows i and j of a matrix and returns the modified matrix, or swaps elements i and j of a vector and returns the modified vector.
--
Related: CSWP,ROW+,ROW-

0100AB ~xSCHUR
([[]] → [[Q]] [[T]])
Schur Decomp. of Squ. Matrix Cmd
--
Returns the Schur decomposition of a square matrix.
--
Related: LQ,LU,QR,SVD,SVL,TRN

02D0AB ~xSEQ
(prog var start end incr → {})
Sequential Calculation Cmd
--
Returns a list of results generated ated by repeatedly executing prog using index var over the range start to end, in increments of incr.
--
Related: DOSUBS,STREAM

0060AB ~xSNRM
([] → x)
Spectral Norm Cmd
--
Returns the spectral norm of an array.
--
Related:
ABS,CNRM,COND,RNRM,SRAD,TRACE

0390AB ~xSORT
({} → {}')
Ascending Order Sort Cmd
--
Sorts the elements in a list in ascending order.
--
Related: REVLIST

0050AB ~xSRAD
([[]] → x)
Spectral Radius Cmd
--
Returns the spectral radius of a square matrix.
--
Related: COND,SNRM,TRACE

0320AB ~xSTREAM
({} prog → x)
Stream Execution Cmd
--
Moves the first two elements from the list onto the stack, and executes prog. The moves the next element (if any) onto the stack, and executes obj again using the previous result and the new element. Repeats this until the list is exhausted, and returns the final result.
--
Related: DOSUBS

00B0AB ~xSVD
([[]] → [[U]] [[V]] [S])
Singular Value Decomposition Cmd
--
Returns the sigular value decomposition of an mn matrix.
--
Related: DIAG→,MIN,SVL

00C0AB ~xSVL
([[]] → [])
Singular Values Cmd
--
Returns the singular values of an mn matrix.
--
Related: MIN,SVD

6.4 T-Z

02C0AB ~xTAIL
({} → {}')
($ → $')
Last Listed Elements Cmd
--
Returns all but the first element of a list or string.
--
Related: HEAD

0040AB ~xTRACE
([[]] → x)
Matrix Trace Cmd
--
Returns the trace of a square matrix.

0000AB ~xVERSION
(→ $ $)
Software Version Cmd
--
Displays the software version and copyright message.

02A0AB ~xXRECV
(name →)
XModem Receive Cmd
--
Prepares the HP 48 to receive an object via XModem. The received object is stored in the given name.
--
Related: BAUD,RECV,RECN,SEND,XSEND

0290AB ~xXSEND
(name →)
XModem Send Cmd
--
Sends a copy of the named object via XModem.
--
Related: BAUD,RECN,RECV,SEND,XRECV

6.5 Non A-Z

7 ML Entry Points

7.1 General Purpose

```
06816      SAVPTR              D0 to RPLTOP
                               D1 to DSKTOP
                               B to RETTOP
                               D to FREETOP
                               Clear carry
05123      GETPTRLOOP          <see>GETPTR , Loop to RPL
01140      DisableIntr         Disable interrupts.
```

7.2 Errors

7.2.1 Generating Errors

7.3 Hexadecimal Math

7.4 Memory Handling

7.4.1 General Memory Handling Routines

```
03019      SKIPOB              Skip object in D0, clears ST1,
                               clears carry, P=0
                               --> D0 = addr past object
                               Uses: A.A C.A P ST1 RSTK2
81228      $jSKIPBODY
```

7.4.2 Moving and Swapping Memory Areas

7.4.3 Allocating Memory in TEMPOB

7.4.4 Working with Memory

7.5 Display

7.6 Popping and Pushing

7.6.1 Pointers

```
03A86      PUSHA                 Push A, Loop
```

7.6.2 TRUE and FALSE

```
DEC0B      GPPushFLoop           <see>GETPTR , Push FALSE, Loop
DEBF2      GPPushTLoop           <see>GETPTR , Push TRUE, Loop
81117      $jGPOvrWrFLp
81125      $jGPOvrWrTLp
8118E      $jOvrWrF/TLp
81195      $jOvrWrFLoop
8119C      $jOvrWrT/FLp
811A3      $jOvrWrTLoop
811FE      $jPshF/TLoop
81205      $jPshT/FLoop
```

7.6.3 System Binary Integers (BINT)

```
066BC      POP#                  Pop # to A.A
```

7.6.4 Real and Complex Numbers

```
22706      PUSH%LOOP             ( A -> %push ) Push A as %,
                                 <see>GETPTRLOOP
```

7.7 Keyboard Handling

7.8 Various ML Entries

7.9 Object Types

029E8	DOARRY	Array prologue
		5 size
		5 prologue of objects
		5 # of dimensions
		5n dimensions
		.. objects (content only)
02B62	DOBAK	Backup prologue
		5 size
		2 # of chars in name
		.. name
		.. object
		5 DOBINT
		5 CRC
		Apparently unused on the 49
02911	DOBINT	BINT prologue
		5 number (hex)
029BF	DOCHAR	Character prologue
		2 character
02977	DOCMP	Complex number prologue
		3 real exponent
		12 real mantissa
		1 real sign
		3 complex exponent
		12 complex mantissa
		1 complex sign
02DCC	DOCODE	Code prologue
		5 length
		.. machine code
02D9D	DOCOL	Secondary prologue
		.. objects
		5 SEMI
02A2C	DOCSTR	String prologue
		5 length
		.. characters
0299D	DOECMP	Long complex prologue
		5 real exponent
		15 real mantissa
		1 real sign
		5 complex exponent
		15 complex mantissa
		1 complex sign
02955	DOEREL	Long real prologue
		5 exponent
		15 mantissa
		1 sign

02ADA	DOEXT	Unit object prologue .. object (usually a real) .. unit 5 SEMI
02B1E	DOGROB	GROB prologue 5 size 5 height 5 width
02A4E	DOHSTR	HXS prologue 5 length .. hex digits, reverse order aka: DOHXS
02E48	DOIDNT	Global name (ID) prologue 2 # of characters .. characters
02E6D	DOLAM	Local name (LAM) prologue see <see>DOIDNT
02A0A	DOLNKARRY	Linked array prologue Not used by the system.
02B40	DOLIB	Library prologue 5 size 2 # of characters .. name 2 # of characters (unless 0) 3 library ID 5 hash table offset 5 message table offset 5 link table offset 5 config object offset .. contents 4 CRC ; XLIBs: 1 or 3: kind 3 library ID 3 command ID .. object -- <REF>TEXT:Libraries
02A74	DOLIST	List prologue see <see>DOCOL
02933	DOREAL	Real number prologue 3 exponent 12 mantissa 1 sign
02E92	DOROMP	XLIB prologue 3 library ID 3 command #

02A96	DORRP	Directory prologue Home directory: 3 # of attached libs n*[3 library ID 5 address of hash table 5 address of message table] 5 offset of last object *[5 offset to previous object 00000 for the first one 2 # of characters .. name of object 2 # of characters .. object] ; Subdirectories: 3 # of attached library 7FF if none 5 offset of last object .. same as above
02AB8	DOSYMB	Symbolic prologue .. objects 5 SEMI
02AFC	DOTAG	Tagged object prologue 2 # of chars in tag .. tag .. object
02B88	DOEXT0	
02BAA	DOEXT1	aka: DOACPTR
02BCC	DOEXT2	
02BEE	DOEXT3	
02C10	DOEXT4	

8 RAM entries

Note that pointers (->...) are always 5 nibbles wide.

8.1 UART buffering

F02B8	uart_buffer	UART buffer area (512)

9 Miscellaneous Entries

9.1 Various Matrix operations

```
00D0C3    ~la+COL
00E0C3    ~la+COLs
0110C3    ~la+ELEM
0100C3    ~la+ELEMc
00F0C3    ~la+ELEMr
00C0C3    ~la+RCsLP
00A0C3    ~la+ROW
00B0C3    ~la+ROWs
0090C3    ~la-COL
0070C3    ~la-ROW
0140C3    ~la>COL
0170C3    ~la>DIAG
0130C3    ~la>ELEM
0120C3    ~la>ROW
0200C2    ~laBPMUTE
0160C3    ~laCOL>
04D0C2    ~laCOND
04E0C2    ~laCONDdone
0040C3    ~laCSWP
0180C3    ~laDIAG>
0510C2    ~laDT
0070C2    ~laECQhQAQh
0040C2    ~laEGV
0020C2    ~laEGVL
0030C2    ~laEGVL%%
05E0C2    ~laEGetTiny
01B0C2    ~laEgCQRI
01C0C2    ~laEgCQRIk1
01A0C2    ~laEgCSchur
0060C2    ~laEgHF
00A0C2    ~laEgIsoVal
0110C2    ~laEgM*G2
00F0C2    ~laEgM*RG3
```

```
01E0C2    ~laEgPrep
00C0C2    ~laEgQRIk0
0080C2    ~laEgQkHA
00B0C2    ~laEgRQRI
00D0C2    ~laEgRQRIk1
0090C2    ~laEgRSchur
00E0C2    ~laEgRWilk3
0130C2    ~laEgRotR
0190C2    ~laEgSclCls
0120C2    ~laEgValr
0180C2    ~laEgVcPair
0160C2    ~laEgVcSngl
0170C2    ~laEgVcUrhs
01D0C2    ~laEgVecC
0150C2    ~laEgVecR
0100C2    ~laEgWilk2
01F0C2    ~laFPMUTE
0210C2    ~laFSCALE
0550C2    ~laIV
0570C2    ~laIVF
0560C2    ~laIVc
0220C2    ~laLSQ
0590C2    ~laLsdScale
02B0C2    ~laQ2hX
0260C2    ~laQRF
02E0C2    ~laQRSVecUp
0300C2    ~laQRmaxSV
02F0C2    ~laQRminSV
02D0C2    ~laQRrank
0270C2    ~laQhA
0280C2    ~laQhB
0360C2    ~laRANK
0000C3    ~laRANM
0050C3    ~laRCI
0060C3    ~laRCIJ
0150C3    ~laROW>
02A0C2    ~laRQF
05B0C2    ~laRREF
```

```
02C0C2    ~laRSVF
0020C3    ~laRSWP
0030C3    ~laRSWP2
0010C3    ~laRanInt
05D0C2    ~laRedHere?
05C0C2    ~laRedRow
0520C2    ~laRefineDT
0050C2    ~laSCHUR
0290C2    ~laSETDIAG
0340C2    ~laSNORM
0350C2    ~laSNORM%%
0000C2    ~laSRAD
0530C2    ~laSV
0380C2    ~laSVD
0370C2    ~laSVL
0540C2    ~laSVc
05A0C2    ~laScIntRnd
0140C2    ~laSchur2
03C0C2    ~laSvdAPPhV
03D0C2    ~laSvdAPk
03F0C2    ~laSvdBDQR
0420C2    ~laSvdBDirC
0490C2    ~laSvdCROTL
0480C2    ~laSvdCROTR
0410C2    ~laSvdFDirC
03E0C2    ~laSvdGPROT
0430C2    ~laSvdGShft
03A0C2    ~laSvdLtUBD
04A0C2    ~laSvdPSort
04B0C2    ~laSvdPrep
0400C2    ~laSvdQR2x2
0470C2    ~laSvdQRB
0460C2    ~laSvdQRF
0450C2    ~laSvdQRSB
0440C2    ~laSvdQRSF
0390C2    ~laSvdUBD
03B0C2    ~laSvdUQhQA
04C0C2    ~laTRACE
```

```
0580C2    ~laULHSVF
0240C2    ~laUserLQ
0500C2    ~laUserLU
0230C2    ~laUserQR
0010C2    ~laVMAX%%
04F0C2    ~laVMAXJRP
0080C3    ~laVec-
```

9.2 Undescribed Entry Points

```
81AB8      !MATTRNnc
81AAE      !REDIMTEMP
81AB3      !REDIMUSER
8185B      DropSysObs
81FB8      EVALCRUNCH
81C39      InitEnab
80A18      lbrac
81BD0      LINECHANGE
81856      NEWINDEP
82CC9      POLErrorTrap
028FC      PRLG
8301B      PTYPE>PINFO
82FE9      SYMBNUMSOLVE
82DA5      X@
```

10 Entries sorted by address

Here follows a list of entries sorted by address. Six-digit addresses are always sorted after five-digit addresses. The six-digit addresses for rompointers and flashpointers consist of the pointer number (first three digits) and the flashbank/library id (last three digits). Sorting of these addresses uses first the flashbank/library id and then the pointer number, so 000123 will be sorted after FFF122.

```
01140   DisableIntr
028FC   PRLG
02911   DOBINT
02933   DOREAL
02955   DOEREL
02977   DOCMP
0299D   DOECMP
029BF   DOCHAR
029E8   DOARRY
02A0A   DOLNKARRY
02A2C   DOCSTR
02A4E   DOHSTR
02A4E   DOHXS
02A74   DOLIST
02A96   DORRP
02AB8   DOSYMB
02ADA   DOEXT
02AFC   DOTAG
02B1E   DOGROB
02B40   DOLIB
02B62   DOBAK
02B88   DOEXT0
02BAA   DOEXT1
02BAA   DOACPTR
02BCC   DOEXT2
02BEE   DOEXT3
02C10   DOEXT4
02D9D   DOCOL
02DCC   DOCODE
02E48   DOIDNT
02E6D   DOLAM
02E92   DOROMP
03019 SKIPOB
0312B SEMI
03A81 TRUE
03A86 PUSHA
03AC0 FALSE
03FDB TYPEREAL
03FE5 TYPECMP
03FEF TYPELIST
03FF9 TYPEIDNT
04003 TYPECOL
0400D TYPESYMB
04017 TYPERRP
04021 TYPELAM
0402B TYPEEREL
04035 TYPEEXT
0403F any
0403F BINT0
0403F ZERO
04049 real
04049 BINT1
04049 MEMERR
04049 ONE
04053 BINT2
04053 TWO
04053 cmp
0405D BINT3
0405D str
0405D THREE
04067 FOUR
04067 arry
04067 BINT4
04071 FIVE
04071 BINT5
04071 list
0407B id
0407B idnt
0407B BINT6
0407B SIX
04085 lam
04085 SEVEN
04085 BINT7
0408F seco
0408F EIGHT
0408F BINT8
04099 symb
04099 BINT9
04099 NINE
040A3 BINT10
040A3 sym
040A3 TEN
040AD hxs
040AD BINT11
040AD ELEVEN
040B7 BINT12
040B7 grob
040B7 TWELVE
040C1 THIRTEEN
040C1 TAGGED
040C1 BINT13
040CB FOURTEEN
040CB unitob
040CB EXT
040CB BINT14
040D5 rompointer
```

040D5 FIFTEEN
040D5 BINT15
040DF REALOB
040DF BINT16
040DF SIXTEEN
040E9 BINT17
040E9 SEVENTEEN
040E9 2REAL
040E9 REALREAL
040F3 EIGHTEEN
040F3 BINT18
040FD NINETEEN
040FD BINT19
04107 BINT20
04107 TWENTY
04111 TWENTYONE
04111 BINT21
0411B BINT22
0411B TWENTYTWO
04125 BINT23
04125 TWENTYTHREE
0412F TWENTYFOUR
0412F BINT24
04139 BINT25
04139 TWENTYFIVE
04143 REALSYM
04143 BINT26
04143 TWENTYSIX
0414D TWENTYSEVEN
0414D BINT27
04157 TWENTYEIGHT
04157 BINT28
04161 BINT29
04161 TWENTYNINE
0416B REALEXT
0416B THIRTY
0416B BINT30
04175 THIRTYONE
04175 BINT31
0417F BINT32
0417F THIRTYTWO
04189 BINT33
04189 THIRTYTHREE
04193 BINT34
04193 THIRTYFOUR
0419D BINT35
0419D THIRTYFIVE
041A7 THIRTYSIX
041B1 THIRTYSEVEN
041B1 BINT37
041BB BINT38
041BB THIRTYEIGHT
041C5 BINT39
041C5 THIRTYNINE
041CF BINT40
041CF FORTY
041CF FOURTY
041D9 FORTYONE
041D9 BINT41
041E3 BINT42
041E3 FORTYTWO
041ED FORTYTHREE
041ED BINT43
05123 GETPTRLOOP
055B5 NULLHXS
055BF NULL$
055C9 NULL{}
055D3 NULLSYMB
055DD NULL::
066BC POP#
06816 SAVPTR
0744F ZERO_DO
0745A ONE_DO
07467 #1+_ONE_DO
08452 RAM-WORDNAME
095A5 XRCLp0
0BCEE dirstrucchk
0BD53 loopdirck
0BDC1 TrueTrue
0E909 CRLF$
10C72 CLOSEUART
117CD AllowPrlcdCl
22706 PUSH%LOOP
25DF8 DropJunk
26B24 DA3OK?NOTIT
28381 ClearLists
2B7F5 EraseGraph
33905 HARDHEIGHT
3949A BadIfEdit
3C635 DoBadKey
3C77E 'DoBadKeyT
3CC36 TakeOver
3D3F6 ParseExpr
420E3 PageUpYGrob
42127 PageUpNGrob
4216B PageDnYGrob
421AF PageDnNGrob
45611 CK0
458CA %ABSCOERCE
45AC1 CK1&Dispatch
45AD2 CK2&Dispatch
45AE3 CK3&Dispatch
45AF4 CK4&Dispatch
45B05 CK5&Dispatch
48F1C %15
48FB7 %25
4F7F8 tok:
5CAD5 Update_tTYPE
5FFF4 xC1
6001A xC2
60040 xC3
60066 xC4
6008C xC5
62D36 SETPLOTENV

64184 SetRedrawFlag
67B3E Docrunchc
69347 GETYPOS
69E2E CHECKHEIGHT
6E8C2 GraphTableUI
6E908 GraphZoomUI
6E94E ZoomAutoUI
7B7D8 DefaultHint
7D3ED DummyMenuErr
7D42E MiniVar
7D46A EnsureMenuOff
80000 BINT44
80000 illnameerr
80000 FORTYFOUR
8000A FORTYFIVE
8000A BINT45
80014 FORTYSIX
80014 BINT46
8001E rrp
8001E BINT47
8001E FORTYSEVEN
80028 BINT48
80028 FORTYEIGHT
80032 FORTYNINE
80032 BINT49
8003C FIFTY
8003C BINT50
80046 2STR
80046 FIFTYONE
80046 BINT51
80050 BINT52
80050 THREEFOUR
80050 FIFTYTWO
8005A BINT53
8005A THREEFIVE
8005A STRLIST
8005A FIFTYTHREE
80064 FIFTYFOUR
80064 BINT54
8006E FIFTYFIVE
8006E BINT55
8006E #THREESEVEN
80078 FIFTYSIX
80078 BINT56
80082 FIFTYSEVEN
80082 BINT57
8008C FIFTYEIGHT
8008C BINT58
80096 BINT59
80096 FIFTYNINE
800A0 SIXTY
800A0 BINT60
800AA SIXTYONE
800AA BINT61
800B4 BINT62
800B4 SIXTYTWO
800BE SIXTYTHREE
800BE BINT63
800BE BINT3Fh
800C8 SIXTYFOUR
800C8 BINT40h
800C8 BINT64
800C8 YHI
800D2 BINT_65d
800D2 BINT65
800D2 ARRYREAL
800D2 SIXTYFIVE
800DC FOURTWO
800DC ARRYCMP
800DC SIXTYSIX
800DC BINT66
800E6 FOURTHREE
800E6 BINT67
800F0 2ARRY
800F0 SIXTYEIGHT
800F0 BINT68
800FA BINT69
800FA FOURFIVE
800FA ARRYLIST
80104 SEVENTY
80104 ARRYID
80104 BINT70
8010E SEVENTYONE
80118 SEVENTYTWO
80122 SEVENTYTHREE
8012C ARRYSYM
8012C SEVENTYFOUR
8012C BINT74
80136 SEVENTYFIVE
80140 SEVENTYSIX
8014A SEVENTYSEVEN
80154 SEVENTYEIGHT
8015E SEVENTYNINE
8015E BINT79
80168 EIGHTY
80168 LISTOB
80168 BINT80
80172 BINT81
80172 LISTREAL
80172 EIGHTYONE
8017C BINT82
8017C LISTCMP
80186 BINT83
80186 FIVETHREE
80190 BINT84
80190 FIVEFOUR
8019A BINT85
8019A 2LIST
801A4 BINT86
801A4 LISTID
801A4 FIVESIX
801AE BINT87
801AE LISTLAM
801B8 EIGHTYEIGHT

801C2 EIGHTYNINE
801CC NINETY
801D6 BINT_91d
801D6 BINT91
801E0 NINETYTWO
801EA NINETYTHREE
801F4 NINETYFOUR
801FE NINETYFIVE
80208 BINT96
80208 idntany
80208 BINT_96d
80212 BINT97
80212 IDREAL
8021C NINETYEIGHT
8021C IDCMP
80226 NINETYNINE
80230 IDARRY
80230 ONEHUNDRED
80230 BINT100
8023A IDLIST
8023A SIXFIVE
80244 char
80244 BINT111
8024E lamany
80258 LAMREAL
80262 BINT_114
8026C BINT_115d
8026C BINT115
80276 BINT_116d
80276 BINT116
80280 BINT_117d
80280 LAMLIST
8028A BINT_122d
8028A BINT122
80294 BINT_128d
80294 BINT128
80294 BINT80h
8029E BINT_130d
8029E BINT130d
8029E XHI-1
8029E BINT130
802A8 XHI
802A8 BINT_131d
802A8 BINT131
802A8 BINT131d
802B2 library
802BC SYMBREAL
802C6 SYMBCMP
802D0 SYMBSYM
802DA SYMBUNIT
802E4 backup
802EE SYMOB
802F8 SYMREAL
80302 SYMCMP
8030C SYMSTR
80316 SYMARRY
80320 SYMLIST
8032A SYMID
80334 SYMLAM
8033E SYMSYMB
80348 SYMSYM
80352 SYMEXT
8035C HXSREAL
80366 2HXS
80370 BINTC0h
8037A 2GROB
80384 TAGGEDANY
8038E EXTREAL
80398 UNITSYM
80398 EXTSYM
803A2 2EXT
803AC ROMPANY
803B6 BINT253
803C0 BINT255d
803CA REALOBOB
803CA FSTFLOATROM#
803D4 #_258_d
803D4 #_102
803D4 Err#Chr00
803DE #SyntaxErr
803E8 BINT_263d
803E8 #ObTypeBase
803F2 REALREALOB
803FC 3REAL
80406 #_291_d
80406 #_123
80406 Err#Kill
80410 #_124
80410 #_292_d
80410 Err#NoLstStk
8041A BINT_305d
8041A #NoRoomForSt
80424 BINT_306d
80424 #Warning:NL
8042E REALSTRSTR
8042E BINT_307d
8042E #Error:
80438 VLMpurgemsg
80442 VLMmsg
8044C REALSTRID
8044C VLMstkmsg
80456 VLMundomsg
80460 VLMhistmsg
8046A VLMUserKeys
80474 VLMAlarmMsg
8047E VLMlastargs
80488 VLMcmdlmsg
80492 Err#Cont
8049C REALLISTREAL
8049C INTEGER337
804A6 CMPOBOB
804B0 Err#NoLstArg
804B0 #_205
804B0 #_517_d

804BA STRREALREAL
804C4 ARRYREALREAL
804CE ARRYREALCMP
804D8 3ARRY
804E2 ARRYLISTREAL
804EC ARRYLISTCMP
804F6 LISTREALOB
80500 LISTREALREAL
8050A LISTLISTOB
80514 IDREALOB
8051E IDLISTOB
80528 LSTBIMACROM#
80528 FSTMACROROM#
80528 LAMANYANY
80532 PROGIDREAL
8053C PROGIDCMP
80546 PROGIDLIST
80550 PROGIDEXT
8055A ATTNERR
80564 SYMREALREAL
8056E SYMREALCMP
80578 SYMREALSYM
80582 SYMCMPREAL
8058C SYMCMPCMP
80596 SYMCMPSYM
805A0 SYMIDREAL
805AA SYMIDCMP
805B4 SYMIDLIST
805BE SYMIDEXT
805C8 SYMSYMREAL
805D2 SYMSYMCMP
805DC 3SYM
805E6 XFERFAIL
805F0 PROTERR
805FA InvalServCmd
80604 Connecting
8060E Retry
80618 #CAlarmErr
80622 EXTOBOB
8062C #EXITERR
80636 MINUSONE
80640 MINUSTWO
8064A MINUSTHREE
80654 MINUSFOUR
8065E MINUSFIVE
80668 %0
8067D %1
80692 %2
806A7 %3
806BC %4
806D1 %5
806E6 %6
806FB %7
80710 %8
80725 %9
8073A %-1
8074F %-2
80764 %-3
80779 %-4
8078E %-5
807A3 %-6
807B8 %-7
807CD %-8
807E2 %-9
807F7 %PI
8080C %%PI
80826 %MAXREAL
8083B %-MAXREAL
80850 %MINREAL
80865 %-MINREAL
8087A %%0
80894 %%1
808AE %%2
808C8 %%3
808E2 %%4
808FC %%5
80916 %%.1
80930 %%.5
8094A %%10
80964 %e
80979 %.5
8098E %-.5
809A3 %10
809B8 %180
809CD %200
809E2 %360
809F7 %400
80A0C tok]
80A0C rbrac
80A18 lbrac
80A26 tok[
80A32 tok{
80A3E tok}
80A4A toksharp
80A56 tokuscore
80A62 tok$
80A6E tok&
80A7A tokESC
80A86 tok>>
80A92 tok<<
80A9E tokexponent
80AAA tokanglesign
80AB6 tokSIGMA
80AC2 tokWHERE
80ACE 14SPACES$
80AF4 LF$
80AF4 NEWLINE$
80B00 $DER
80B10 SPACE$
80B10 tok_
80B1C tokUNKNOWN
80B34 tokquote
80B40 tok’
80B4C tok,

80B58 tok.
80B64 tok;
80B70 toklparen
80B7C tokrparen
80B88 tokˆ
80B94 tok*
80BA0 tok/
80BAC tok+
80BB8 tok-
80BC4 tok=
80BD0 tokSQRT
80BDC tokDER
80BE8 tokCTGROB
80BFA tokCTSTR
80C08 tok0
80C14 tok1
80C20 tok2
80C2C tok3
80C38 tok4
80C44 tok5
80C50 tok6
80C5C tok7
80C68 tok8
80C74 tok9
80C80 CHR_00
80C87 CHR_...
80C8E CHR_’
80C95 CHR_DblQuote
80C9C CHR_#
80CA3 CHR_*
80CAA CHR_+
80CB1 CHR_,
80CB8 CHR_-
80CBF CHR_.
80CC6 CHR_/
80CCD CHR_0
80CD4 CHR_1
80CDB CHR_2
80CE2 CHR_3
80CE9 CHR_4
80CF0 CHR_5
80CF7 CHR_6
80CFE CHR_7
80D05 CHR_8
80D0C CHR_9
80D13 CHR_:
80D1A CHR_;
80D21 CHR_<
80D28 CHR_=
80D2F CHR_>
80D36 CHR_A
80D3D CHR_B
80D44 CHR_C
80D4B CHR_D
80D52 CHR_E
80D59 CHR_F
80D60 CHR_G
80D67 CHR_H
80D6E CHR_I
80D75 CHR_J
80D7C CHR_K
80D83 CHR_L
80D8A CHR_M
80D91 CHR_N
80D98 CHR_O
80D9F CHR_P
80DA6 CHR_Q
80DAD CHR_R
80DB4 CHR_S
80DBB CHR_T
80DC2 CHR_U
80DC9 CHR_V
80DD0 CHR_W
80DD7 CHR_X
80DDE CHR_Y
80DE5 CHR_Z
80DEC CHR_a
80DF3 CHR_b
80DFA CHR_c
80E01 CHR_d
80E08 CHR_e
80E0F CHR_f
80E16 CHR_g
80E1D CHR_h
80E24 CHR_i
80E2B CHR_j
80E32 CHR_k
80E39 CHR_l
80E40 CHR_m
80E47 CHR_n
80E4E CHR_o
80E55 CHR_p
80E5C CHR_q
80E63 CHR_r
80E6A CHR_s
80E71 CHR_t
80E78 CHR_u
80E7F CHR_v
80E86 CHR_w
80E8D CHR_x
80E94 CHR_y
80E9B CHR_z
80EA2 CHR_->
80EA9 CHR_<<
80EB0 CHR_>>
80EB7 CHR_Angle
80EBE CHR_Deriv
80EC5 CHR_Integral
80ECC CHR_LeftPar
80ED3 CHR_Newline
80EDA CHR_Pi
80EE1 CHR_RightPar
80EE8 CHR_Sigma
80EEF CHR_Space

80EF6 CHR_UndScore
80EFD CHR_[
80F04 CHR_]
80F0B CHR_{
80F12 CHR_}
80F19 CHR_<=
80F20 CHR_>=
80F27 CHR_<>
80F2E STypeDirChr
80F35 STypeZapChr
80F3C STypeNteChr
80F43 STypePgmChr
80F4A STypeVarChr
80F51 STypeLstChr
80F58 STypeMatChr
80F5F STypeLibChr
80F66 STypeTgtChr
80F6D STypeVecChr
80F74 STypeAnyChr
80F7B $_R<<
80F8B $_R<Z
80F9B $_XYZ
80FAB $_<<>>
80FB9 $_{}
80FD5 $_''
80FE3 $_::
80FF1 $_LRParens
80FFF $_2DQ
8100D $_ECHO
8101F $_EXIT
81031 $_Undefined
8104D $_RAD
8105D $_GRAD
81117 $jGPOvrWrFLp
81125 $jGPOvrWrTLp
8118E $jOvrWrF/TLp
81195 $jOvrWrFLoop
8119C $jOvrWrT/FLp
811A3 $jOvrWrTLoop
811FE $jPshF/TLoop
81205 $jPshT/FLoop
81228 $jSKIPBODY
8123D DOINDIR
81252 %%>%
81257 %>%%
8125C %0>
81261 %0<
81266 %0=
8126B %<
81270 %<=
81275 %>=
8127A %<>
8127F %=
81284 %>
81289 %*
8128E %+
81293 %-
81298 %/
8129D %^
812A2 %ABS
812A7 %ACOS
812AC %ACOSH
812B1 %ALOG
812B6 %ANGLE
812BB %ASIN
812C0 %ASINH
812C5 %ATAN
812CA %ATANH
812CF %CEIL
812D4 %CH
812D9 %CHS
812DE %COMB
812E3 %COS
812E8 %COSH
812ED %EXP
812F2 %EXPM1
812F7 %EXPONENT
812FC %FLOOR
81301 %FP
81306 %IP
8130B %LN
81310 %LNP1
81315 %LOG
8131A %MANTISSA
8131F %MOD
81324 %NFACT
81329 %NROOT
8132E %OF
81333 %PERM
81338 %RAN
8133D %SGN
81342 %SIN
81347 %SINH
8134C %SQRT
81351 %T
81356 %TAN
8135B %TANH
81360 #0=case
81365 OVER#=case
8136A #=case
8136F #=casedrop
81374 2SWAP
81379 SEVENROLL
81379 7ROLL
8137E EIGHTROLL
8137E 8ROLL
81383 DUMP
81388 DUP#0=csedrp
8138D EQcasedrop
81392 Embedded?
81397 GETLAMPAIR
8139C #MAX
813A1 #MIN
813A6 #NOT

```
817A2   RPIT
817A7   RPITE
817AC   SKIP
817B1   UNTIL
817B6   WHILE
817BB   ABND
817C0   DOBIND
817C5   GETLAM
817CA   PUTLAM
817CF   #*
817D4   #+
817D9   #-
817DE   #/
817E3   #AND
817E8   #1+
817ED   #1-
817F2   #2*
817F7   #2+
817FC   #2-
81801   #2/
81806   #=
8180B   #<>
81810   #<
81815   #>
8181A   #0=
8181F   #0<>
81824   AND
8182E   NOT
81833   OR
8183D   XOR
81842   EQ
81847   EQUAL
8184C   TYPE
81851   'DROPFALSE
81856   NEWINDEP
8185B   DropSysObs
81860   ABUFF
81865   DISPN
81865   BIGDISPN
8186A   DISPROW1
8186A   DISP@01
8186A   BIGDISPROW1
8186F   DISPROW2
8186F   DISP@09
8186F   BIGDISPROW2
81874   DISPROW3
81874   BIGDISPROW3
81874   DISP@17
81879   BIGDISPROW4
81879   DISP@25
81879   DISPROW4
8187E   DISPROW5
81883   DISPROW6
81888   DISPROW7
8188D   DISPROW8
81892   GBUFF
81897   GROB!
8189C   GROB!ZERO
818A1   HARDBUFF
818A6   HARDBUFF2
818AB   INVGROB
818B0   MAKEGROB
818B5   SUBGROB
818BA   $>BIGGROB
818BF   $>GROB
818C4   $>grob
818C9   #>$
818CE   Putbitmap
818D3   Orbitmap
818D8   Xorbitmap
818DD   NotAndbitmap
818E2   Put3x5
818E7   Put5x7
818EC   #NEG
818F1   #DIV
818F6   #MOD
818FB   #>=
81900   #<=
81905   #ODD
8190A   d>
8190F   d<
81914   d>=
81919   d<=
8191E   d*
81923   dDIV
81928   dMOD
8192D   dmuldiv
81932   d>%
81937   INT_00
8193C   INT_01
81941   INT_02
81946   INT_03
8194B   INT_04
81950   INT_05
81955   INT_06
8195A   INT_07
8195F   INT_08
81964   INT_09
81969   INT_0A
8196E   INT_0B
81973   INT_0C
81978   INT_0D
8197D   INT_0E
81982   INT_0F
81987   CAND
8198C   COR
81991   TopicVarN@
81996   TopicVarN!
8199B   BinLookup
819A0   CARCOMP
819A5   INNERCOMP
819AA   LENCOMP
819AF   NTHELCOMP
819B4   NULLCOMP?
```

819B9 SUBCOMP
819BE OCRC
819C3 OSIZE
819C8 &$
819CD >H$
819D2 >T$
819D7 CAR$
819DC LEN$
819E1 NULL$?
819E6 SUB$
819EB ERROR@
819F0 ERRORSTO
819F5 ERRORCLR
819FA DROPNULL$
819FF TWODROPNULL$
81A04 GETEXITMSG
81A09 EXITMSGSTO
81A0E ERRSET
81A13 ERRTRAP
81A18 ERRJMP
81A1D SETLBERR
81A22 SETMEMERR
81A27 SETROMPERR
81A2C ATTNFLGCLR
81A31 CHECKKEY
81A36 GETTOUCH
81A3B REPKEY?
81A40 GARBAGE
81A45 MEM
81A4A TOTEMPOB
81A4F INTEMNOTREF?
81A54 ClrSysFlag
81A59 ClrUserFlag
81A5E SetSysFlag
81A63 SetUserFlag
81A68 TestSysFlag
81A6D TestUserFlag
81A72 SUB$1#
81A77 AlowPrlcdCl
81A7C AllowPRLCD
81A81 SetEcma94
81A86 FLUSHKEYS
81A86 FLUSH
81A8B CLCD10
81A90 ATTN?
81A95 SLOW
81A9A VERYSLOW
81A9F VERYVERYSLOW
81AA4 MDIMS
81AA9 PUTEL
81AAE !REDIMTEMP
81AB3 !REDIMUSER
81AB8 !MATTRNnc
81ABD GDISP?
81AC2 HEIGHTENGROB
81AC7 KILLGDISP
81ACC TOADISP
81AD1 TOGDISP
81AD6 WINDOWCORNER
81ADB WINDOWDOWN
81AE0 WINDOWLEFT
81AE5 WINDOWRIGHT
81AEA WINDOWUP
81AEF WINDOWXY
81AF4 PIXOFF
81AF9 PIXOFF3
81AFE PIXON
81B03 PIXON?
81B08 PIXON3
81B0D PIXON?3
81B12 OLASTOWDOB!
81B12 OLastRomWrd!
81B17 ERRBEEP
81B1C setbeep
81B21 RDUP
81B26 RROLL
81B2B RPICK
81B30 CopyRegCOb
81B35 NEXTCOMPOB
81B3A POS$
81B3F POS$REV
81B44 POSCHR
81B49 POSCHRREV
81B4E ClrDA1Bad
81B53 ClrDA2aBad
81B58 ClrDA2bBad
81B5D ClrDA2bNoCh
81B62 DA1Bad?
81B67 DA2aBad?
81B6C DA2bNoCh?
81B71 SetDA1Bad
81B76 SetDA1IsStat
81B7B SetDA1NoCh
81B80 SetDA2aBad
81B85 SetDA2aNoCh
81B8A SetDA2bBad
81B8F SetDA2bNoCh
81B94 SetDA3Bad
81B99 SetDA3NoCh
81B9E SetDA3ValidF
81BA3 SetNoRollDA2
81BA8 INHARDROM?
81BAD GROBDIM
81BB2 LINEOFF
81BB7 LINEOFF3
81BBC LINEON
81BC1 LINEON3
81BC6 TOGLINE
81BCB TOGLINE3
81BD0 LINECHANGE
81BD5 ARSIZE
81BDA DIMLIMITS
81BDF GETATELN
81BE4 MAKEARRY

81BE9 #>CHR
81BEE #>HXS
81BF3 $>ID
81BF8 %>C%
81BFD C%>%
81C02 CHR>#
81C07 ID>$
81C0C #>ROMPTR
81C11 @
81C16 @LAM
81C1B COMPROMID
81C20 COMPILEID
81C25 CONTEXT!
81C2A CONTEXT@
81C2F CREATE
81C34 SYSCONTEXT
81C34 HOMEDIR
81C39 InitEnab
81C3E MAKERRP
81C43 OFFSRRP
81C48 ONSRRP?
81C4D PURGE
81C57 LASTRAM-WORD
81C5C PREVRAM-WORD
81C61 REPLACE
81C66 ROMPTR>#
81C6B ROMPTR@
81C70 STO
81C75 STOLAM
81C7F SYSRRP?
81C84 TOSRRP
81C89 tok8cktrior
81C8E tok8trior
81C93 nultrior
81C98 SETNONEXTERR
81C9D SETSIZEERR
81CA2 SETTYPEERR
81CA7 SETSTACKERR
81CAC ABSCOERCE
81CB1 COERCE
81CB6 UNCOERCE
81CBB COMPEVAL
81CD9 CK&DISPATCH0
81CDE CK&DISPATCH1
81CE3 CK&DISPATCH2
81CE8 !>ARRAY
81CF2 top&
81CF7 psh1&
81CFC psh1&rev
81D01 psh
81D06 roll2ND
81D0B roll3RD
81D10 unroll2ND
81D15 unroll3RD
81D1A rollNTH
81D1F roll4TH
81D24 unroll4TH
81D29 unrollNTH
81D2E ROT#1+UNROT
81D33 NULLargcase
81D38 DEPTH
81D3D DUP
81D42 2DUP
81D47 NDUP
81D4C SWAP
81D51 DROP
81D56 2DROP
81D5B NDROP
81D60 ROT
81D65 OVER
81D6A PICK
81D6F ROLL
81D74 UNROLL
81D79 BREAK
81D7E GETSERIAL
81D83 UARTBUFLEN
81D88 OpenUart?Clr
81D8D SetInAplet
81D92 ClrInAplet
81D97 InApletF?
81D9C SV_?defined
81DA1 CalcDir+Offs
81DA6 AppDir+Offs
81DAB Contxt+Offs
81DB0 LastBut0
81DB5 LastBut1
81DBA LastBut2
81DBF LastBut3
81DC4 LastBut4
81DC9 LastBut5
81DCE LastBut6
81DD3 LastBut7
81DD8 LastBut8
81DDD LastBut9
81DE2 LastBut10
81DE7 LastBut11
81DEC LastBut12
81DF1 LastBut13
81DF6 LastBut14
81DFB LastBut15
81E00 LastBut16
81E05 LastBut17
81E0A LastBut18
81E0F LastBut19
81E14 LastBut20
81E19 LastBut21
81E1E LastBut22
81E23 LastBut23
81E28 LastBut24
81E2D LastBut25
81E32 LastBut26
81E37 LastBut27
81E3C LastBut28
81E41 LastBut29

81E46 LastBut30
81E4B LastBut31
81E50 LastBut32
81E55 LASTBUTN
81E5A Bubble
81E64 MATCON
81E6E MATREDIM
81E78 MATTRN
81E82 REDIMPREP
81E8C >HCOMP
81E96 >TCOMP
81EA0 CDRCOMP
81EB4 AttnPOSCOMP
81EBE EQUALPOSCMP
81EC8 Lookup
81ED2 Lookup.1
81EDC POSCOMP
81EE6 PUTLIST
81EF0 ::N
81EFA {}N
81F04 SYMBN
81F0E EXTN
81F18 &COMP
81F22 ClrListUtil
81F2C ClearList1
81F36 ClearList2
81F40 ClearList3
81F4A ClearList4
81F54 ClearList5
81F5E ClearList6
81F68 ClearList7
81F72 ClearList8
81F7C ClearList9
81F86 ClearList0
81F90 ListNames
81F9A ABORT
81FA4 blackbox
81FAE DUP#0=case
81FB8 EVALCRUNCH
81FC2 xEVAL
81FC7 NUMEVAL1
81FD1 ORcase
81FDB FalseFalse
81FE5 SafeCrunch%
81FEF DA1OK?
81FF9 DA2aOK?
82003 DA2bOK?
8200D DA2OK?
82017 DA3OK?
82021 DAsOK?
8202B DA2aLess1OK?
82035 SetDA1Valid
8203F SetDA2aValid
82049 SetDA2bValid
82053 SetDA2Valid
8205D SetDA3Valid
82067 SetDA1Temp
82071 SetDA2aTemp
8207B SetDA3Temp
82085 SetDA2aEcho
8208F ClrDA1OK
82099 ClrDA2aOK
820A3 ClrDA2bOK
820AD ClrDA2OK
820B7 ClrDA3OK
820C1 SetDA2NoCh
820CB SetDA12NoCh
820D5 SetDA23NoCh
820DF SetDA13NoCh
820E9 SetDA12a3NoCh
820E9 SetDA12a3NCh
820F3 SetDAsNoCh
820FD SetDA2Temp
82107 SetDA12Temp
82111 SetDAsTemp
8211B AnyDABad?
82125 SetDAsBad
8212F DO#EXIT
82139 DO$EXIT
82143 AddEq$
8214D AlDrawMenu
82157 BLANKIT
82161 CHECKPICT
8216B check_xrange
82175 check_yrange
8217F CKGROBFITS
82189 DefauPtXit
82193 DefauTrcInit
8219D DispXFunc
821A7 DispYFunc
821B1 DOCLLCD
821BB DOERASE
821C5 DoDemo
821CF DODISP
821D9 DOLCD>
821E3 DOLCD>g0
821ED DOSHOWIT1
821F7 DOSHOWIT1C
82201 EraseGrob
8220B FuncPLoop
82215 FUNCPLOT
8221F GDISPCENTER
82229 GETXMIN
82233 GETYMIN
8223D GETXMAX
82247 GETYMAX
82251 GROB+#
8225B GROBBUFF!
82265 GROB>GDISP
8226F GROBVIEW
82279 InitIndep
82283 Init_window
8228D JUMPBOT
82297 JUMPTOP

827A1 IFEDispHelp
827AB IFECheck
827B5 IFEChooseNext
827BF IFEChoosByChr
827D3 DoKeyChoos/Ck
827DD DoKeyChoose
827E7 DoKeyCheck
82805 SelNextField
8280F SelPrevField
82823 Sel&DispNextF
8282D IsIFMenu?
82837 FitLeftSmF
82841 FitRightSmF
8284B WidthSmF
82855 GetNextFId
8285F GetPrevFId
82869 DecompOb
82873 StdDecompNoNL
8287D DecompNoNL
82887 RclFieldVal
82891 IsNullField?
8289B ?GetFObTypes
828A5 GetChoiceList
828AF GetChoiceFmt
828B9 RclResetVal
828C3 DoMatEdit
828EB NextTextLine
828FF DoQueryBox
82909 DoAlert&Query
82913 DoSpecAlert
8291D BlankClient
82927 BlankHelp
82931 GtoField
8293B GetPart1
82945 GetPart2
8294F GetPart3
82959 EvalPart1
82963 EvalPart2
8296D EvalPart3
82977 KeyFace
82981 FAreaBad?
8298B SetFAreaOK
82995 SetFAreaBad
8299F SetFAreasOK
829A9 SetFAreasBad
829B3 MakeTitleBar
829BD ?FixFieldKeys
829C7 Choose&DoTask
829D1 PlaceFVals
829DB Done?CkNoNul:
829E5 #=Lookup
829EF ?GetMsg
829F9 DoResultTab
82A03 LockAlpha
82A0D UnLockAlpha
82A17 grobCheck
82A21 grobCheckX
82A2B grobInvChk
82A35 grobInvChkX
82A3F grobUnCheckX
82A49 grobInvUnChkX
82A53 grobInvNoChk
82A71 grobQueryIcon
82A7B grobTitleBar
82A85 FTypeText?
82A8F FTypeList?
82A99 FTypeFixedL?
82AA3 FTypeCheck?
82AAD DoSendOb
82AB7 DoSendObToEl
82AC1 DoSendObToSto
82ACB DoSendObToCDi
82AD5 DoSendObToDir
82ADF docr
82AE9 DoRecv/GetOb
82AF3 DoRecvObFrEls
82AFD DoGetObFrSto
82B07 StdIOProc
82B11 GetObTypePr$
82B1B GetObType$
82B25 DoIOStatusBox
82B2F DoIOErrAlert
82B39 ElsieGet
82B43 ElsieSend
82B4D ElsiePkt
82B57 DOFINISH
82B61 SEFINISH
82B6B PrintHist
82B75 PrintVar
82B7F PrintVarOb
82B89 PrintLcd
82B93 prx1
82B9D OpenIO
82BA7 PUTSERIAL
82BB1 PRINT
82BBB 'DoBadKey
82BC5 ?ATTN_QUIT
82BC5 ?ATTNQUIT
82BCF ?CaseKeyDef
82BD9 Ck&DecKeyLoc
82BE3 CodePl>%rc.p
82BED ?CaseRomptr@
82BF7 ?NoTaskSwDef
82C01 DOWAIT
82C0B dowait
82C15 Box/StdLabel
82C1F DispMenu
82C29 DispMenu.1
82C3D GETKEY
82C47 InitMenu
82C51 IsTaskSwKey?
82C5B KEYINBUFFER?
82C65 Key>StdKeyOb
82C6F ModifierKey?

82C79 NoExitAction
82C8D POLSaveUI
82C97 POLKeyUI
82CA1 POLResUI&Err
82CAB POLRestoreUI
82CB5 POLSetUI
82CBF POLSetUIExt
82CC9 POLErrorTrap
82CD3 ParOuterLoop
82CDD REPEATER
82CE7 REPEATERCH
82CFB TempMenuBuff
82D05 TurnOff
82D0F WaitForKey
82D19 CHANGETYPE
82D23 CKREF
82D2D CREATEDIR
82D37 MemStoAns
82D41 Port0Ids
82D4B NULLID
82D55 StoVar
82D5F XEQSETLIB
82D69 StoAns@
82D73 StoAns@Drp
82D7D UserSto
82D87 XSTOp
82D91 XSTOp0
82D9B XRCLp
82DA5 X@
82DAF XPURGEp
82DB9 XPURGEp0
82DC3 HOME>BAK
82DCD OB>BAK
82DD7 GetBVars.1
82DE1 GetApDirLst
82DEB CatThisCxt
82DF5 CatPgm
82DFF CatNot
82E09 DoInApLbCxt
82E13 DoInPgmCxt
82E1D DoInNotCxt
82E27 GetNextToken
82E31 realPA
82E3B solverTTT
82E45 !*triand
82E4F 'Rapndit
82E59 1GETapndcpl
82E63 FOURpsh
82E77 P::N
82E8B SWAPcompSWAP
82E95 'RSWP1+
82E9F apndit
82EA9 chooselst
82EB3 completed
82EBD dvbind
82EC7 elsiename
82ED1 failed
82EDB filename
82EE5 newsymbPA
82EEF oneexpr
82EF9 onestring
82F03 optfilename
82F0D threeexprs
82F17 tok;triand
82F21 toktriand
82F2B twoexprs
82F35 twostrings
82F3F %0<>
82F49 %0>=
82F53 %D>R
82F5D %FACT
82F67 %HMS+
82F71 %HMS-
82F7B %HMS>
82F85 %>HMS
82F8F %MAX
82F99 %MIN
82FA3 %POL>%REC
82FAD %R>D
82FB7 %RANDOMIZE
82FC1 %REC>%POL
82FCB %SPH>%REC
82FD5 DOROOT
82FDF RNDXY
82FE9 SYMBNUMSOLVE
82FF3 TRCXY
82FFD ?>ROMPTR
83007 ?ROMPTR>
83011 RESOROMP
8301B PTYPE>PINFO
83025 tTYPE@
8302F GetElNoRomp
83039 DoRomPtrKey
83043 resolved
8304D NDUPN
83057 metaDUP
83061 psh&
8306B psh1
83075 psh1top&
8307F &&
83089 pshmonop
83093 psharg
8309D pshtop&
830A7 pshzer
830B1 #1-SWAP
830B1 pull
830BB pullrev
830C5 pullpsh1&
830CF reversym
830D9 unsyminner
830E3 syminner&
830ED syminnertwo
830F7 syminner&N
83101 argswap&&

8354D DTYPECSTR?
83557 DUPTYPEEXT?
83561 DUPTYPEGROB?
8356B DUPTYPEHSTR?
83575 DUPTYPEIDNT?
8357F DUPTYPELAM?
83589 DTYPELIST?
83589 DUPTYPELIST?
83593 DUPTYPEREAL?
83593 DTYPEREAL?
8359D DUPTYPEROMP?
835A7 DUPTYPERRP?
835B1 DUPTYPESYMB?
835BB DUPTYPETAG?
835C5 {NoteText}
835CF {NumVars}
835D9 {SketchSet}
835E3 2DROPFALSE
835F7 INCOMPDROP
8360B NTHCOMPDROP
8361F APPEND_SPACE
83633 7UNROLL
83647 #0=UNTIL
83674 DUP@
83688 DUPROMPTR@
8369C #=ITE
836B0 INNERDUP
836C4 NOTAND
836D8 TOTEMPSWAP
836EC ROT2DUP
83700 ROTAND
83714 ROTOVER
83728 DUPDUP
8373C OVERDUP
83750 COERCEDUP
83764 UNROTDUP
83778 4UNROLLDUP
8378C NTHCOMDDUP
837A0 OVERSWAP
837A0 OVERUNROT
837B4 ROLLSWAP
837C8 NULL$SWAP
837DC SUB$SWAP
837F0 %MAXorder
8380E ?SKIPSWAP
83822 1ABNDSWAP
8383B ROT+SWAP
8383B ROT#+SWAP
83854 4PICK+SWAP
83854 4PICK#+SWAP
8386D #+SWAP
83881 #-SWAP
83895 #1+SWAP
838A9 ZEROSWAP
838BD #1-1SWAP
838D6 ONESWAP
838EA COERCESWAP
838FE %>%%SWAP
83926 XYZ>ZTRUE
8393A 4ROLLSWAP
8394E 3PICKSWAP
83962 4PICKSWAP
83976 1GETSWAP
8398A ?SWAP
8399E !append$SWAP
839B2 NOT?SWAPDROP
839CB ?SWAPDROP
839E4 #1+NDROP
839E4 N+1DROP
839F8 ROLLDROP
83A0C MDIMSDROP
83A20 DUPROT
83A34 DROPROT
83A48 #1-ROT
83A70 4ROLLROT
83A70 FOURROLLROT
83A84 4UNROLLROT
83A98 DROPOVER
83AAC EQOVER
83AC0 #+OVER
83AD4 #-OVER
83AE8 ZEROOVER
83AFC UNROTOVER
83B10 4ROLLOVER
83B24 3PICKOVER
83B38 4PICKOVER
83B4C DUPPICK
83B60 DUPROLL
83B88 8UNROLL
83B9C 10UNROLL
83BB0 OVERARSIZE
83BC4 'ERRJMP
83BD8 caseERRJMP
83BEC ?CARCOMP
83C00 NEWLINE$&$
83C00 NEWLINE&$
83C14 #1-{}N
83C28 TWO{}N
83C3C THREE{}N
83C50 DUPINCOMP
83C64 SWAPINCOMP
83C78 DUPNULL$?
83C8C DUPNULLCOMP?
83CA0 DUPLENCOMP
83CB4 #1-SUB$
83CC8 1_#1-SUB
83CC8 1_#1-SUB$
83CDC LAST$
83CF0 #1+LAST$
83D04 DUP$>ID
83D18 SWAP%>C%
83D2C 'NOP
83D40 ::NEVAL
83D54 2GETEVAL

83D68 DROPRDROP
83D81 SWAPCOLA
83D95 XYZ>ZCOLA
83DA9 #0=?SKIP
83DC2 #1=?SKIP
83DDB #=?SKIP
83DF4 ONE_EQ
83E08 #>?SKIP
83E21 COLASKIP
83E35 NOT_UNTIL
83E4E NOT_WHILE
83E67 DUP#0<>WHILE
83E80 DUPINDEX@
83E94 SWAPINDEX@
83EA8 OVERINDEX@
83EBC SWAPLOOP
83ED5 DROPLOOP
83EEE DUP#0_DO
83F07 toLEN_DO
83F25 1GETABND
83F39 DUP1LAMBIND
83F3E 1LAMBIND
83F52 caseTRUE
83F66 TRUEFALSE
83F66 TrueFalse
83F7A FALSETRUE
83F7A FalseTrue
83F8E ZEROFALSE
83FA2 ONEFALSE
83FB6 #=casedrpfls
83FD9 casedrpfls
83FF2 case2drpfls
8400B caseFALSE
8401F ORNOT
84033 EQUALNOT
84047 2DUPEQ
8405B EQOR
8406F EQUALOR
84083 2#0=OR
840A1 OVER#0=
840B5 OVER#<
840C9 #<3
840DD DUP#<7
840F6 INNER#1=
8410A #5=
8411E #2<>
84132 OVER#>
84146 #>1
84146 ONE#>
8415A 2DUP#+
8415A DUP3PICK#+
8416E ROT#+
84182 OVER#+
84196 3PICK#+
841AA 4PICK#+
841BE ROT#-
841D2 OVER#-
841E6 INDEX@#-
841FA SWAPOVER#-
8420E ROT#1+
84222 SWAP#1-
84236 DROP#1-
8424A ERROROUT
8425E SWAP2DUP
84272 RSKIP
84290 GROB!ZERODRP
842A4 casedrptru
842BD NOTcaseTRUE
842D6 NOTcaseFALSE
842EF ?SEMIDROP
84308 SWAPUnDROP
8431C SWAPUnNDROP
84330 DUP'
84344 SWAP'
84358 DROP'
8436C OVER'
84380 STO'
84394 TRUE'
843A8 ONEFALSE'
843C1 FALSE'
843D5 #1+'
843E9 'R'R
84407 'RRDROP
84420 ONECOLA
84434 dvarlsBIND
84448 'LAMLNAMESTO
84461 'xDEREQ
8447A DUPNULL{}?
84493 DUPZERO
844A7 DUPONE
844BB SWAPONE
844CF ONEDUP
844CF ONEONE
844E3 DUPTWO
844F7 NOTcsdrpfls
84510 caseSIZEERR
84524 NcaseSIZEERR
84538 CKREAL
84551 NcaseTYPEERR
84579 'xDER
8458D %%/>%
845A1 UNCOERCE%%
845B5 DUP%0=
845C9 SWAP%%/
845DD caseDrpBadKy
845F6 caseDEADKEY
845F6 caseDoBadKey
8460F GROBDIMw
84637 XYZW>YWZX
84637 SWAP4ROLL
8464B 2DUP5ROLL
8465F SWAP3PICK
84673 3PICK3PICK
84687 SWAP4PICK

```
8469B   OVER5PICK
846AF   EQUALcasedrp
846C8   DUP#0=csDROP
846E1   jEQcase
846F5   ANDcase
84709   EQUALcase
8471D   #<case
84731   #1=case
84745   #<>case
84759   #>2case
84772   #>case
84786   j%0=case
8479A   REALcase
847AE   dARRYcase
847C2   dLISTcase
847D6   EditExstCase
847EA   ANDNOTcase
847FE   EQUALNOTcase
84812   dIDNTNcase
84826   dREALNcase
8483A   EQIT
84853   DUP#0=IT
8486C   ANDITE
84880   EQITE
84894   #0=ITE
848A8   #<ITE
848BC   #>ITE
848D0   DUP#0=ITE
848E4   UserITE
848F8   SysITE
8490C   top&Cr
84925   metaROTDUP
84939   ROTUntop&
8494D   rolltwotop&
8494D   roll2top&
84961   plDRPpZparg
84975   &$SWAP
84989   SWAPCKREF
8499D   pZpargSWAPUn
849B1   DROPNDROP
849C5   2OVER
849D9   ?Ob>Seco
849F2   Ob>Seco
84A06   2Ob>Seco
84A1A   ZEROISTOPSTO
84A1A   ExitAtLOOP
84A2E   1GETLAM#0=
84A42   OVER#1-
84A56   #=POSCOMP
84A6F   EQUALPOSCOMP
84A88   NTHOF
84A9C   EQLookup
84ABA   CHR>$
84AD3   1NULLLAM{}
84AD8   NULLLAM
84AE4   CALCCXT!
84AF4   CALCCXT@
84B04   PGMCXT!
84B14   PGMCXT@
84B24   NOTESCXT!
84B34   NOTESCXT@
84B44   apletPTR!
84B54   apletPTR@
84B64   funcPTR!
84B74   funcPTR@
84B84   polarPTR!
84B94   polarPTR@
84BA4   paramPTR!
84BB4   paramPTR@
84BC4   seqPTR!
84BD4   seqPTR@
84BE4   statPTR!
84BF4   statPTR@
84C04   solvePTR!
84C14   solvePTR@
84C24   otherPTR!
84C34   otherPTR@
84C44   TopicVar1!
84C54   TopicVar1@
84C64   TopicVar2!
84C74   TopicVar2@
84C84   TopicVar3!
84C94   TopicVar3@
84CA4   TopicVar4!
84CB4   TopicVar4@
84CC4   TopicVar5!
84CD4   TopicVar5@
84CE4   TopicVar6!
84CF4   TopicVar6@
84D04   TopicVar7!
84D14   TopicVar7@
84D24   TopicVar8!
84D34   TopicVar8@
84D44   TopicVar9!
84D54   TopicVar9@
84D64   TopicVar10!
84D74   TopicVar10@
84D84   TopicVar11!
84D94   TopicVar11@
84DA4   TopicVar12!
84DB4   TopicVar12@
84DC4   TopicVar13!
84DD4   TopicVar13@
84DE4   TopicVar14!
84DF4   TopicVar14@
84E04   TopicVar15!
84E14   TopicVar15@
84E24   TopicVar16!
84E34   TopicVar16@
84E44   TopicVar17!
84E54   TopicVar17@
84E64   TopicVar18!
84E74   TopicVar18@
84E84   TopicVar19!
```

84E94 TopicVar19@
84EA4 TopicVar20!
84EB4 TopicVar20@
84EC4 TopicVar21!
84ED4 TopicVar21@
84EE4 TopicVar22!
84EF4 TopicVar22@
84F04 TopicVar23!
84F14 TopicVar23@
84F24 TopicVar24!
84F34 TopicVar24@
84F44 TopicVar25!
84F54 TopicVar25@
84F64 TopicVar26!
84F74 TopicVar26@
84F84 TopicVar27!
84F94 TopicVar27@
84FA4 TopicVar28!
84FB4 TopicVar28@
84FC4 TopicVar29!
84FD4 TopicVar29@
84FE4 TopicVar30!
84FF4 TopicVar30@
85004 TopicVar31!
85014 TopicVar31@
85024 TopicVar32!
85034 TopicVar32@
85044 TopicVar33!
85054 TopicVar33@
85064 TopicVar34!
85074 TopicVar34@
85084 TopicVar35!
85094 TopicVar35@
850A4 TopicVar36!
850B4 TopicVar36@
850C4 TopicVar37!
850D4 TopicVar37@
850E4 TopicVar38!
850F4 TopicVar38@
85104 TopicVar39!
85114 TopicVar39@
85124 TopicVar40!
85134 TopicVar40@
85144 TopicVar41!
85154 TopicVar41@
85164 TopicVar42!
85174 TopicVar42@
85184 TopicVar43!
85194 TopicVar43@
851A4 TopicVar44!
851B4 TopicVar44@
851C4 TopicVar45!
851D4 TopicVar45@
851E4 TopicVar46!
851F4 TopicVar46@
85204 TopicVar47!
85214 TopicVar47@
85224 TopicVar48!
85234 TopicVar48@
85244 TopicVar49!
85254 TopicVar49@
85264 TopicVar50!
85274 TopicVar50@
85284 TopicVar51!
85294 TopicVar51@
852A4 TopicVar52@
852B4 TopicVar52!
852C4 TopicVar53@
852D0 doptr@
852E0 TopicVar53!
852EC doptr!
85307 TopicVar54@
85317 TopicVar54!
85327 TopicVar55@
85337 TopicVar55!
85347 TopicVar56@
85357 TopicVar56!
85367 TopicVar57@
85377 TopicVar57!
85387 TopicVar58@
85397 TopicVar58!
853A7 TopicVar59@
853B7 TopicVar59!
853C7 TopicVar60@
853D7 TopicVar60!
853E7 TopicVar61@
853F7 TopicVar61!
85407 TopicVar62@
85417 TopicVar62!
85427 TopicVar63@
85437 TopicVar63!
85447 TopicVar64@
85457 TopicVar64!
85467 TopicVar65@
85477 TopicVar65!
85487 TopicVar66@
85497 TopicVar66!
854A7 TopicVar67@
854B7 TopicVar67!
854C7 TopicVar68@
854D7 TopicVar68!
854E7 TopicVar69@
854F7 TopicVar69!
85507 TopicVar70@
85517 TopicVar70!
85527 TopicVar71@
85537 TopicVar71!
85547 TopicVar72@
85557 TopicVar72!
85567 TopicVar73@
85577 TopicVar73!
85587 TopicVar74@
85597 TopicVar74!
855A7 TopicVar75@

855B7 TopicVar75!
855C7 TopicVar76@
855D7 TopicVar76!
855E7 TopicVar77@
855F7 TopicVar77!
85607 TopicVar78@
85617 TopicVar78!
85627 TopicVar79@
85637 TopicVar79!
85647 TopicVar80@
85657 TopicVar80!
85667 TopicVar81@
85677 TopicVar81!
85687 TopicVar82@
85697 TopicVar82!
856A7 TopicVar83@
856B7 TopicVar83!
856C7 TopicVar84@
856D7 TopicVar84!
856E7 TopicVar85@
856F7 TopicVar85!
85707 TopicVar86@
85717 TopicVar86!
85727 TopicVar87@
85737 TopicVar87!
85747 TopicVar88@
85757 TopicVar88!
85767 TopicVar89@
85777 TopicVar89!
85787 TopicVar90@
85797 TopicVar90!
857A7 TopicVar91!
857B7 TopicVar91@
857C7 TOLVar1!
857D7 TOLVar1@
857E7 TOLVar2!
857F7 TOLVar2@
85807 TOLVar3!
85817 TOLVar3@
85827 TOLVar4!
85837 TOLVar4@
85847 TOLVar5!
85857 TOLVar5@
85867 TOLVar6!
85877 TOLVar6@
85887 TOLVar7!
85897 TOLVar7@
858A7 TOLVar8!
858B7 TOLVar8@
858C7 TOLVar9!
858D7 TOLVar9@
858E7 TOLVar10!
858F7 TOLVar10@
85907 TOLVar11!
85917 TOLVar11@
85927 TOLVar12!
85937 TOLVar12@
85947 TOLVar13!
85957 TOLVar13@
85967 TOLVar14!
85977 TOLVar14@
85987 TOLVar15!
85997 TOLVar15@
859A7 TOLVar16!
859B7 TOLVar16@
859C7 TOLVar17!
859D7 TOLVar17@
859E7 TOLVar18!
859F7 TOLVar18@
85A07 TOLVar19!
85A17 TOLVar19@
85A27 TOLVar20!
85A37 TOLVar20@
85A47 TOLVar21!
85A57 TOLVar21@
85A67 TOLVar22!
85A77 TOLVar22@
85A87 TOLVar23!
85A97 TOLVar23@
85AA7 TOLVar24!
85AB7 TOLVar24@
85AC7 TOLVar25!
85AD7 TOLVar25@
85AE9 TOLVar26!
85AFB TOLVar26@
85B0D TOLVar27!
85B1F TOLVar27@
85B31 TOLVar28!
85B43 TOLVar28@
85B55 TOLVar29!
85B67 TOLVar29@
85B79 TOLVar30!
85B8B TOLVar30@
85B9D TOLVar31!
85BAF TOLVar31@
85BC1 TOLVar32!
85BD3 TOLVar32@
85BE5 TOLVar33!
85BF7 TOLVar33@
85C09 TOLVar34!
85C1B TOLVar34@
85C2D TOLVar35!
85C3F TOLVar35@
85C51 TOLVar36!
85C63 TOLVar36@
85C75 TOLVar37!
85C87 TOLVar37@
85C99 TOLVar38!
85CAB TOLVar38@
85CBD TOLVar39!
85CCF TOLVar39@
85CE1 TOLVar40!
85CF3 TOLVar40@
85D05 TOLVar41!

```
85D17   TOLVar41@       85FC3 TOLVar60@       8626F TOLVar79@
85D29   TOLVar42!       85FD5 TOLVar61!       86281 TOLVar80!
85D3B   TOLVar42@       85FE7 TOLVar61@       86293 TOLVar80@
85D4D   TOLVar43!       85FF9 TOLVar62!       862A5 TOLVar81!
85D5F   TOLVar43@       8600B TOLVar62@       862B7 TOLVar81@
85D71   TOLVar44!       8601D TOLVar63!       862C9 TOLVar82!
85D83   TOLVar44@       8602F TOLVar63@       862DB TOLVar82@
85D95   TOLVar45!       86041 TOLVar64!       862ED TOLVar83!
85DA7   TOLVar45@       86053 TOLVar64@       862FF TOLVar83@
85DB9   TOLVar46!       86065 TOLVar65!       86311 TOLVar84!
85DCB   TOLVar46@       86077 TOLVar65@       86323 TOLVar84@
85DDD   TOLVar47!       86089 TOLVar66!       86335 TOLVar85!
85DEF   TOLVar47@       8609B TOLVar66@       86347 TOLVar85@
85E01   TOLVar48!       860AD TOLVar67!       86359 TOLVar86!
85E13   TOLVar48@       860BF TOLVar67@       8636B TOLVar86@
85E25   TOLVar49!       860D1 TOLVar68!       8637D TOLVar87!
85E37   TOLVar49@       860E3 TOLVar68@       8638F TOLVar87@
85E49   TOLVar50!       860F5 TOLVar69!       863A1 TOLVar88!
85E5B   TOLVar50@       86107 TOLVar69@       863B3 TOLVar88@
85E6D   TOLVar51!       86119 TOLVar70!       863C5 TOLVar89!
85E7F   TOLVar51@       8612B TOLVar70@       863D7 TOLVar89@
85E91   TOLVar52!       8613D TOLVar71!       863E9 TOLVar90!
85EA3   TOLVar52@       8614F TOLVar71@       863FB TOLVar90@
85EB5   TOLVar53!       86161 TOLVar72!       8640D TOLVar91!
85EC7   TOLVar53@       86173 TOLVar72@       8641F TOLVar91@
85ED9   TOLVar54!       86185 TOLVar73!       86431 TOLVar92!
85EEB   TOLVar54@       86197 TOLVar73@       86443 TOLVar92@
85EFD   TOLVar55!       861A9 TOLVar74!       86455 TOLVar93!
85F0F   TOLVar55@       861BB TOLVar74@       86467 TOLVar93@
85F21   TOLVar56!       861CD TOLVar75!       86479 TOLVar94!
85F33   TOLVar56@       861DF TOLVar75@       8648B TOLVar94@
85F45   TOLVar57!       861F1 TOLVar76!       8649D TOLVar95!
85F57   TOLVar57@       86203 TOLVar76@       864AF TOLVar95@
85F69   TOLVar58!       86215 TOLVar77!       864C1 TOLVar96!
85F7B   TOLVar58@       86227 TOLVar77@       864D3 TOLVar96@
85F8D   TOLVar59!       86239 TOLVar78!       864E5 TOLVar97!
85F9F   TOLVar59@       8624B TOLVar78@       864F7 TOLVar97@
85FB1   TOLVar60!       8625D TOLVar79!       86509 TOLVar98!
```

87523 TOLVar212@
87535 TOLVar213!
87547 TOLVar213@
87559 TOLVar214!
8756B TOLVar214@
8757D TOLVar215!
8758F TOLVar215@
875A1 TOLVar216!
875B3 TOLVar216@
875C5 TOLVarN!
875E8 TOLVarN@
8760B ClrAllTVars
87641 ClrAllTOLVs
8765D %0AllTopicVs
87698 %0AllTOLVars
876D3 TOLVarSet!
877A0 %0TOLVarSet
877F0 1getcxt!
87804 DoInCxt
8785E DoInCalcCxt
87877 DoInAppCxt
87890 DoInFuncCxt
878A9 DoInPolarCxt
878C2 DoInParamCxt
878DB DoInSeqCxt
878F4 DoInStatCxt
8790D DoInSolveCxt
87926 DoInOtherCxt
879F3 otherNG?
87A26 GET@tTYPER
88527 ?DispMenu
9165A DispMsgBox
9216B TBFuncDecomp
97C29 STErrorGrob
99D95 STTableInit
99E30 STInitCols
99EF3 STArryList
99FFC STTableExit
9A024 STTableDisp
9A03D STDoSmall
9A7B8 STDoMedium
9AEBB STTableHKeys
9B5E8 STJump
9B6BA STJumpN
9BFA8 STEditKeyDef
9BFD8 STInsKeyDef
9C0B9 STBigKeyDef
9C1FA STStatsKeyDef
9C2CC SWAPStatFlag
9C4B2 STSortCol
A4CEF %%-1
BA700 #+#2-
BD0B7 STO_tTYPE
BD10C Xmin!
BD120 Xmin@
BD14D Xmax!
BD161 Xmax@
BD1A2 Ymin@
BD1E3 Ymax@
BD224 Xmin2@
BD265 Xmax2@
BD2A6 Ymin2@
BD2E7 Ymax2@
BD328 HTick@
BD369 VTick@
BD3AA HZoom@
BD3EB VZoom@
BD42C Angle@
BD46D LastX@
BD4AE LastY@
BD4EF LastIndep@
BD530 LastEq@
BD571 PlotFlag@
BD5B2 Plot2Flg@
BD5F3 PlotCKSUM@
BD634 Split1CKSUM@
BD675 Split2CKSUM@
BD6B6 TStart@
BD6F7 TStep@
BD738 TZoom@
BD779 TFlags@
BD7BA Trow@
BD7FB Tcol@
BD83C TCol1@
BD87D SplitRow@
BD8BE BTRow1@
BD8FF Format@
BD940 Digits@
BDEA3 Root@
BDEE9 Slope@
BDF2F Isect@
BDF75 Area@
BDFBB Extremum@
BE001 ListOfEqs@
BE03D funcCache@
C0298 StatType!
C02B1 StatType@
C02F7 SVarType@
C033D StatPlot@
C0383 HisWidth@
C03C9 Hmin@
C040F Hmax@
C0455 StatMark@
C049B StatModel@
C04E1 Stat2Flag@
C0527 ReCalcFlag@
C056D StatFont@
C0586 StatMisc[]@
C061C StatFit@
C6B6B Seq_eval0
C6BC5 Seq_eval1
C6E45 Seq_eval2
C6FEE Seq_evaln
DDCFE 4PICK

DE46C EQcase
DE622 2'RCOLARPITE
DEBF2 GPPushTLoop
DEC0B GPPushFLoop
DF0F3 #8+
DF133 #3-
E03A1 DemoFrames
F02B8 uart_buffer
085002 ~xxSIZE
086002 ~xxPOS
0A6002 ~xERASEA
0B6002 ~xDISPLAY>
0B7002 ~x>DISPLAY
0B8002 ~xPLOT>
0B9002 ~x>PLOT
0BA002 ~xBLANKGROB
0CE002 ~rpnDER
0D1002 ~rpnINTG
0D4002 ~rpnWHERE
0D7002 ~rpnAPPLY
0DA002 ~COMPLEXDUMM
0DB002 ~POLARDUMMY
0000A1 ~DoMenuKey1N
0010A1 ~DoMenuKey2N
0020A1 ~DoMenuKey3N
0030A1 ~DoMenuKey4N
0040A1 ~DoMenuKey5N
0050A1 ~DoMenuKey6N
0060A1 ~PlotViewKey
0070A1 ~SymbViewKey
0080A1 ~TableViewKe
0090A1 ~MissingKey1
00A0A1 ~UpArrow
00B0A1 ~MissingKey2
00C0A1 ~TopicLibKey
00D0A1 ~NSKey3.2
00E0A1 ~NSKey3.3
00F0A1 ~LArrow
0100A1 ~DArrow
0110A1 ~RArrow
0120A1 ~NSKey4.1
0130A1 ~NSKey4.2
0140A1 ~NSKey4.3
0150A1 ~NSKey4.4
0160A1 ~NSKey4.5
0170A1 ~NSKey4.6
0180A1 ~Enter/Again
0190A1 ~NSKey5.2
01A0A1 ~NSKey5.3
01B0A1 ~NSKey5.4
01C0A1 ~NSKey5.5
01D0A1 ~NSKey6.1
01E0A1 ~NSKey6.2
01F0A1 ~NSKey6.3
0200A1 ~NSKey6.4
0210A1 ~NSKey6.5
0220A1 ~NSKey7.1
0230A1 ~NSKey7.2
0240A1 ~NSKey7.3
0250A1 ~NSKey7.4
0260A1 ~NSKey7.5
0270A1 ~NSKey8.1
0280A1 ~NSKey8.2
0290A1 ~NSKey8.3
02A0A1 ~NSKey8.4
02B0A1 ~NSKey8.5
02C0A1 ~AttentionKe
02D0A1 ~NSKey9.2
02E0A1 ~NSKey9.3
02F0A1 ~NSKey9.4
0300A1 ~NSKey9.5
0000A2 ~LSKey1.1
0010A2 ~LSKey1.2
0020A2 ~LSKey1.3
0030A2 ~LSKey1.4
0040A2 ~LSKey1.5
0050A2 ~LSKey1.6
0060A2 ~LSKey2.1
0070A2 ~LSKey2.2
0080A2 ~LSKey2.3
0090A2 ~MissingKey3
00A0A2 ~FarUArrow
00B0A2 ~MissingKey4
00C0A2 ~LSKey3.1
00D0A2 ~LSKey3.2
00E0A2 ~LSKey3.3
00F0A2 ~FarLArrow
0100A2 ~FarDArrow
0110A2 ~FarRArrow
0120A2 ~LSKey4.1
0130A2 ~LSKey4.2
0140A2 ~LSKey4.3
0150A2 ~LSKey4.4
0160A2 ~LSKey4.5
0170A2 ~LSKey4.6
0180A2 ~LSKey5.1
0190A2 ~LSKey5.2
01A0A2 ~DoExponent
01B0A2 ~LSKey5.4
01C0A2 ~LSKey5.5
01D0A2 ~LSKey6.1
01E0A2 ~LSKey6.2
01F0A2 ~LSKey6.3
0200A2 ~LSKey6.4
0210A2 ~LSKey6.5
0220A2 ~LSKey7.1
0230A2 ~LSKey7.2
0240A2 ~LSKey7.3
0250A2 ~LSKey7.4
0260A2 ~LSKey7.5
0270A2 ~LSKey8.1
0280A2 ~LSKey8.2
0290A2 ~LSKey8.3
02A0A2 ~LSKey8.4

02B0A2 ~LSKey8.5
02C0A2 ~LSKey9.1
02D0A2 ~LSKey9.2
02E0A2 ~LSKey9.3
02F0A2 ~LSKey9.4
0300A2 ~LSKey9.5
0000A4 ~ANSKey1.1
0010A4 ~ANSKey1.2
0020A4 ~ANSKey1.3
0030A4 ~ANSKey1.4
0040A4 ~ANSKey1.5
0050A4 ~ANSKey1.6
0060A4 ~ANSKey2.1
0070A4 ~ANSKey2.2
0080A4 ~ANSKey2.3
0090A4 ~ANSKey2.4
00A0A4 ~ANSKey2.5
00B0A4 ~ANSKey2.6
00C0A4 ~ANSKey3.1
00D0A4 ~ANSKey3.2
00E0A4 ~ANSKey3.3
00F0A4 ~ANSKey3.4
0100A4 ~ANSKey3.5
0110A4 ~ANSKey3.6
0120A4 ~ANSKey4.1
0130A4 ~ANSKey4.2
0140A4 ~ANSKey4.3
0150A4 ~ANSKey4.4
0160A4 ~ANSKey4.5
0170A4 ~ANSKey4.6
0180A4 ~ANSKey5.1
0190A4 ~ANSKey5.2
01A0A4 ~ANSKey5.3
01B0A4 ~ANSKey5.4
01C0A4 ~ANSKey5.5
01D0A4 ~ANSKey6.1
01E0A4 ~ANSKey6.2
01F0A4 ~ANSKey6.3
0200A4 ~ANSKey6.4
0210A4 ~ANSKey6.5
0220A4 ~ANSKey7.1
0230A4 ~ANSKey7.2
0240A4 ~ANSKey7.3
0250A4 ~ANSKey7.4
0260A4 ~ANSKey7.5
0270A4 ~ANSKey8.1
0280A4 ~ANSKey8.2
0290A4 ~ANSKey8.3
02A0A4 ~ANSKey8.4
02B0A4 ~ANSKey8.5
02C0A4 ~ANSKey9.1
02D0A4 ~ANSKey9.2
02E0A4 ~ANSKey9.3
02F0A4 ~ANSKey9.4
0300A4 ~ANSKey9.5
0000A5 ~ALSKey1.1
0010A5 ~ALSKey1.2
0020A5 ~ALSKey1.3
0030A5 ~ALSKey1.4
0040A5 ~ALSKey1.5
0050A5 ~ALSKey1.6
0060A5 ~ALSKey2.1
0070A5 ~ALSKey2.2
0080A5 ~ALSKey2.3
0090A5 ~ALSKey2.4
00A0A5 ~ALSKey2.5
00B0A5 ~ALSKey2.6
00C0A5 ~ALSKey3.1
00D0A5 ~ALSKey3.2
00E0A5 ~ALSKey3.3
00F0A5 ~ALSKey3.4
0100A5 ~ALSKey3.5
0110A5 ~ALSKey3.6
0120A5 ~ALSKey4.1
0130A5 ~ALSKey4.2
0140A5 ~ALSKey4.3
0150A5 ~ALSKey4.4
0160A5 ~ALSKey4.5
0170A5 ~ALSKey4.6
0180A5 ~ALSKey5.1
0190A5 ~ALSKey5.2
01A0A5 ~ALSKey5.3
01B0A5 ~ALSKey5.4
01C0A5 ~ALSKey5.5
01D0A5 ~ALSKey6.1
01E0A5 ~ALSKey6.2
01F0A5 ~ALSKey6.3
0200A5 ~ALSKey6.4
0210A5 ~ALSKey6.5
0220A5 ~ALSKey7.1
0230A5 ~ALSKey7.2
0240A5 ~ALSKey7.3
0250A5 ~ALSKey7.4
0260A5 ~ALSKey7.5
0270A5 ~ALSKey8.1
0280A5 ~ALSKey8.2
0290A5 ~ALSKey8.3
02A0A5 ~ALSKey8.4
02B0A5 ~ALSKey8.5
02C0A5 ~ALSKey9.1
02D0A5 ~ALSKey9.2
02E0A5 ~ALSKey9.3
02F0A5 ~ALSKey9.4
0300A5 ~ALSKey9.5
0000AB ~xVERSION
0010AB ~xLININ
0020AB ~XEQSYMLIN
0030AB ~xCOND
0040AB ~xTRACE
0050AB ~xSRAD
0060AB ~xSNRM
0070AB ~xRANK
0080AB ~xLSQ
0090AB ~xEGV

00A0AB ~xEGVL
00B0AB ~xSVD
00C0AB ~xSVL
00D0AB ~xLU
00E0AB ~xQR
00F0AB ~xLQ
0100AB ~xSCHUR
0110AB ~xRREF
0120AB ~xRANM
0130AB ~x>ROW
0140AB ~xROW>
0150AB ~x>COL
0160AB ~xCOL>
0170AB ~x>DIAG
0180AB ~xDIAG>
0190AB ~xROW-
01A0AB ~xROW+
01B0AB ~xCOL-
01C0AB ~xCOL+
01D0AB ~xRSWP
01E0AB ~xCSWP
01F0AB ~xRCI
0200AB ~xRCIJ
0210AB ~XEQDIAG>L
0220AB ~XEQDIAG>R
0230AB ~XEQRANM
0240AB ~xPROOT
0250AB ~xPCOEF
0260AB ~xPEVAL
0270AB ~xCHOOSE
0280AB ~xMSGBOX
0290AB ~xXSEND
02A0AB ~xXRECV
02B0AB ~xHEAD
02C0AB ~xTAIL
02D0AB ~xSEQ
02E0AB ~xDOSUBS
02F0AB ~x\9BLIST
0300AB ~xNSUB
0310AB ~xENDSUB
0320AB ~xSTREAM
0330AB ~x\85LIST
0340AB ~ChkDaList
0350AB ~x\9CLIST
0360AB ~xDOLIST
0370AB ~xCONCAT
0380AB ~xREVLIST
0390AB ~xSORT
03A0AB ~xFmList
03B0AB ~xFmMat
03C0AB ~xxDEG
03D0AB ~xxRAD
03E0AB ~xxGRAD
0430AB ~xFRACTION
0440AB ~xSTAIRSTEP
0450AB ~xCOBWEB
0460AB ~xHIST
0470AB ~xBOXW
04B0AB ~xPOWERFIT
04C0AB ~xQUADFIT
04D0AB ~xCUBICFIT
04E0AB ~xLOGISFIT
04F0AB ~xUSERFIT
0500AB ~xStat1Var
0510AB ~xStat2Var
0520AB ~xxITERATE
0530AB ~xMKMAT
0540AB ~xCOT
0550AB ~xSEC
0560AB ~xCSC
0570AB ~xACOT
0580AB ~xACSC
0590AB ~xASEC
05A0AB ~xRECURSE
05B0AB ~xPOLYFORM
05C0AB ~xPoly
05D0AB ~dopolyz
05E0AB ~dopolydiv
05F0AB ~dopcoeff
0600AB ~metapolyz
0610AB ~polyprep
0620AB ~poly+
0630AB ~polyneg
0640AB ~poly*
0660AB ~{StatVars}
0670AB ~sn*spec
0680AB ~ns*spec
0690AB ~ss*spec
06A0AB ~poly^
06B0AB ~bad^
06C0AB ~poly^#
06D0AB ~dopval
06E0AB ~pdiv
06F0AB ~resymb
0700AB ~poly/
0000B9 ~MiscIFMsg
0000BB ~StatIFMsg
0000BE ~PlotIFMsg
0000BF ~SolveIFMsg
0000C0 ~proot_c
0010C0 ~proot
0020C0 ~proot_r
0030C0 ~peval
0040C0 ~pcoef
0050C0 ~pcoefacc1
0060C0 ~pcoefacc2
0070C0 ~PolyNSymbP
0080C0 ~PolyNSymbF
0000C2 ~laSRAD
0010C2 ~laVMAX%%
0020C2 ~laEGVL
0030C2 ~laEGVL%%
0040C2 ~laEGV
0050C2 ~laSCHUR

```
0000E8  ~doseq
0010E8  ~dosecntuple
0020E8  ~dontuple
0030E8  ~dontuple#
0040E8  ~doidseqn
0050E8  ~seqid
0060E8  ~::args
0070E8  ~ptrargs
0080E8  ~dosecseqn
0090E8  ~udfargs
00A0E8  ~doptrseqn
00B0E8  ~doseqn
00C0E8  ~doseqn#
00D0E8  ~docmdlist
00E0E8  ~seqnargs
00F0E8  ~dolatorre
0100E8  ~elsielists?
0110E8  ~dolatorre+
0120E8  ~dolatorre2
0130E8  ~?NULLSETDIM
0140E8  ~sllatorre
0150E8  ~lslatorre
0160E8  ~lolatorre
0170E8  ~ollatorre
0180E8  ~ollatorre+
0190E8  ~dolist+
01A0E8  ~dosort
01B0E8  ~{id}>{$}
01C0E8  ~{lam}>{$}
01D0E8  ~{$}>{id}
01E0E8  ~{$}>{lam}
01F0E8  ~DoNumeric:
0200E8  ~UseHidden{}
0210E8  ~NotHidden
0220E8  ~INTEMPOB?
0230E8  ~moresub
0240E8  ~morerepl
0250E8  ~mrepln
0260E8  ~mrepll
0270E8  ~mrepll+
0280E8  ~msubln
0290E8  ~msubnl
02A0E8  ~msubll
02B0E8  ~msubll+
02C0E8  ~msubnn
02D0E8  ~copysub
02E0E8  ~arryspec
02F0E8  ~etorc
0300E8  ~LIXRecv
0310E8  ~LIXSend
0000F0  ~INTGSIN
0010F0  ~I:-InvSin
0020F0  ~INTGCOS
0030F0  ~I:LnTan
0040F0  ~INTGTAN
0050F0  ~INTGASIN
0060F0  ~INTGACOS
0070F0  ~INTGATAN
0080F0  ~INTGSINH
0090F0  ~I:-InvSinh
00A0F0  ~I:LnTanh
00B0F0  ~INTGCOSH
00C0F0  ~I:Tanh
00D0F0  ~INTGTANH
00E0F0  ~INTGEXPM
00F0F0  ~INTGALOG
0100F0  ~INTGLN
0110F0  ~INTGLOG
0120F0  ~INTGINV
0130F0  ~I:Atan
0140F0  ~INTGSQ
0150F0  ~INTGSQRT
0160F0  ~I:Asinh
0170F0  ~I:Acosh
0180F0  ~INTGSIGN
0190F0  ~INTGDER
01A0F0  ~mT->=
01B0F0  ~m<-T=
01C0F0  ~mAFlq
01D0F0  ~mAFqq
01E0F0  ~mAFrq
01F0F0  ~m<->+
0200F0  ~m<->-
0210F0  ~m<->*
0220F0  ~m<->/
0230F0  ~m<-A-+
0240F0  ~m<-A--
0250F0  ~m<-A/*
0260F0  ~m<-A//
0270F0  ~m<-A^*
0280F0  ~mA->+-
0290F0  ~mA->--
02A0F0  ~mA->*/
02B0F0  ~mA->//
02C0F0  ~mA->^^
02D0F0  ~mD->/+
02E0F0  ~mD->/-
02F0F0  ~mD->^+
0300F0  ~mD->^-
0310F0  ~mD->E+
0320F0  ~mD->E-
0330F0  ~mD->L*
0340F0  ~mD->L/
0350F0  ~m[]CHS*
0360F0  ~m[]CHS/
0370F0  ~m[]CHSL
0380F0  ~m[]INV^
0390F0  ~m[]INVE
03A0F0  ~m1/[]*
03B0F0  ~m1/[]/
03C0F0  ~m1/[]^
03D0F0  ~m1/[]E
03E0F0  ~m-[]L
03F0F0  ~m-[]*
```

0400F0 ~m-[]/
0410F0 ~m-[]+
0420F0 ~m-[]-
0430F0 ~mE^*
0440F0 ~mE^/
0450F0 ~mE[]^
0460F0 ~mL*^
0470F0 ~mL[]*
0480F0 ~mL[]/
0490F0 ~m<-M+*
04A0F0 ~m<-M-*
04B0F0 ~adjdivsign
04C0F0 ~adjsign
04D0F0 ~adjdiv
04E0F0 ~m<-M*^
04F0F0 ~m<-M/^
0500F0 ~m<-M*E
0510F0 ~m<-M/E
0520F0 ~m<-M+L
0530F0 ~m<-M-L
0540F0 ~mM->op
0550F0 ~m<T>+
0560F0 ~m<T>*
0570F0 ~m<-T+-
0580F0 ~m<-T*/
0590F0 ~m<-[+-
05A0F0 ~m<-[*/
05B0F0 ~m]->+-
05C0F0 ~m]->*/
05D0F0 ~m->[]<-+-
05E0F0 ~m->[]<-*/
05F0F0 ~mCONJ[]
0600F0 ~mRE[]
0610F0 ~mIM[]
0620F0 ~REIM[]*
0630F0 ~m->TRG
0640F0 ~m->DEFSIN
0650F0 ~m->DEFCOS
0660F0 ~m->DEFTAN
0670F0 ~m->DEFASIN
0680F0 ~m->DEFACOS
0690F0 ~m->DEFATAN
06A0F0 ~m->DEFSINH
06B0F0 ~m->DEFCOSH
06C0F0 ~m->DEFTANH
06D0F0 ~m->DEFASINH
06E0F0 ~m->DEFACOSH
06F0F0 ~m->DEFATANH
0700F0 ~mSIN+
0710F0 ~mCOS+
0720F0 ~mTAN+
0730F0 ~mSINH+
0740F0 ~mCOSH+
0750F0 ~mTANH+
0760F0 ~covmanCOL
0770F0 ~WHEREFCNAPP
0780F0 ~WHEREDER
0790F0 ~WHEREIFTE
07A0F0 ~WHEREWHERE
07B0F0 ~covWSPLIT
07C0F0 ~WHEREINTG
07D0F0 ~WHERESUM
07E0F0 ~XSTOCHECK
07F0F0 ~XSTOCHECK10
0800F0 ~covLBSTO
0810F0 ~XEQXDPTCH
0820F0 ~Xcont
0830F0 ~CHECKEXISTS
0840F0 ~covmetaLIBS
0850F0 ~XRCLp?
0860F0 ~XRCLpNL
0870F0 ~XRCLpL
0880F0 ~XRCLp*
0890F0 ~XEVALp?
08A0F0 ~XEVALp*
08B0F0 ~XPURGEp?
08C0F0 ~XPURGEp*
08D0F0 ~convertaddr
08E0F0 ~XRCLp?acc>
08F0F0 ~COLCTDER
0900F0 ~COLCTINTG
0910F0 ~COLCTSUM
0920F0 ~COLCTIFTE
0930F0 ~COLCTQUOTE
0940F0 ~COLCTFCNAP
0960F0 ~covD/DCROSS
0970F0 ~covD/DDOT
0B90F0 ~covD/DINTG
0BC0F0 ~covMANMENU+
0BD0F0 ~covMANMENU*
0BF0F0 ~covMANEXP
0C00F0 ~covMANMENUL
0C10F0 ~covMANCSIV
0C20F0 ~covMANMENUE
0C30F0 ~covMANMENUC
0C40F0 ~covMANTRG
0C50F0 ~covMANATG
0C60F0 ~covINV+
0C70F0 ~covINV-
0C80F0 ~covINV=
0C90F0 ~covINV*
0CA0F0 ~covINV/
0CB0F0 ~covINV^
0CC0F0 ~covINV^X
0CD0F0 ~covINVEXP
0CE0F0 ~covINVSIN
0CF0F0 ~covINVCOS
0D00F0 ~covINVTAN
0D10F0 ~covINVSINH
0D20F0 ~covINVCOSH
0D30F0 ~covINVTANH
0D40F0 ~covINVALOG
0D50F0 ~covINVEXPM1
0000F1 ~nFUNCTION

0010F1 ~nSOLVE
0020F1 ~nPOLAR
0030F1 ~nPARAMETRIC
0040F1 ~nSCATTER
0050F1 ~nHISTOGRAM
0060F1 ~nSEQUENCE
0070F1 ~nBOXWHISKER
0080F1 ~xFUNCTAB
0090F1 ~xPOLARTAB
00A0F1 ~xPARAMTAB
00B0F1 ~xSEQTAB
00C0F1 ~xSTATTAB
00D0F1 ~xSTAT2TAB
00E0F1 ~xSOLVETAB
00F0F1 ~nFUNCTOPIC
0100F1 ~nPOLARTOPIC
0110F1 ~nPARAMTOPIC
0120F1 ~nSEQTOPIC
0130F1 ~nSTATTOPIC
0140F1 ~nSOLVETOPIC
0150F1 ~GetTypeText
0160F1 ~DefauStatTy
0170F1 ~DefauStat2T
0180F1 ~nPTYPE>PINF
0190F1 ~StdApEntry
01A0F1 ~SolveApEntr
01B0F1 ~SETTOPICLAM
01C0F1 ~CLRTOPICLAM
01D0F1 ~GETPLTLABLS
01E0F1 ~POINTERR
01F0F1 ~POINTEXIT
0200F1 ~runalias?
0210F1 ~xFUNCSYMB
0220F1 ~xPOLARSYMB
0230F1 ~xPARAMSYMB
0240F1 ~xSEQSYMB
0250F1 ~xSTATSYMB
0260F1 ~xSTAT2SYMB
0270F1 ~xSOLVESYMB
0280F1 ~xDFLTNOTE
0290F1 ~xDFLTPICT
02A0F1 ~topic_NONE
02B0F1 ~view_NONE
02C0F1 ~view01_CAPL
02D0F1 ~view23_CAPL
02E0F1 ~view45_CAPL
02F0F1 ~view6_CAPLE
0300F1 ~view7_CAPLE
0310F1 ~topic_CAPLE
0320F1 ~xREADNOTE
0330F1 ~xREADPICT
0340F1 ~xUndefined
0000F2 ~xa
0010F2 ~xb
0020F2 ~xc
0030F2 ~xd
0040F2 ~xE
0050F2 ~xf
0060F2 ~xg
0070F2 ~xh
0080F2 ~xI
0090F2 ~xj
00A0F2 ~xk
00B0F2 ~xl
00C0F2 ~xm
00D0F2 ~xn
00E0F2 ~xo
00F0F2 ~xp
0100F2 ~xq
0110F2 ~xr
0120F2 ~xs
0130F2 ~xt
0140F2 ~xu
0150F2 ~xv
0160F2 ~xw
0170F2 ~xx
0180F2 ~xy
0190F2 ~xz
01A0F2 ~xtheta
01B0F2 ~xMEANS
01C0F2 ~xTOTS
01D0F2 ~xSVARS
01F0F2 ~xSSDEV
0200F2 ~xPSDEV
0210F2 ~xNS
0220F2 ~xMINS
0230F2 ~xMAXS
0240F2 ~xMEDIAN
0250F2 ~xQ1
0260F2 ~xQ3
0270F2 ~xMEANX
0280F2 ~xSX
0290F2 ~xSX2
02A0F2 ~xMEANY
02B0F2 ~xSY
02C0F2 ~xSY2
02D0F2 ~xSXY
0300F2 ~xRELERR
0310F2 ~xz0
0320F2 ~xz1
0330F2 ~xz2
0340F2 ~xz3
0350F2 ~xz4
0360F2 ~xz5
0370F2 ~xz6
0380F2 ~xz7
0390F2 ~xz8
03A0F2 ~xz9
03B0F2 ~xm0
03C0F2 ~xm1
03D0F2 ~xm2
03E0F2 ~xm3
03F0F2 ~xm4
0400F2 ~xm5

0410F2 ~xm6
0420F2 ~xm7
0430F2 ~xm8
0440F2 ~xm9
0450F2 ~xl0
0460F2 ~xl1
0470F2 ~xl2
0480F2 ~xl3
0490F2 ~xl4
04A0F2 ~xl5
04B0F2 ~xl6
04C0F2 ~xl7
04D0F2 ~xl8
04E0F2 ~xl9
04F0F2 ~xg0
0500F2 ~xg1
0510F2 ~xg2
0520F2 ~xg3
0530F2 ~xg4
0540F2 ~xg5
0550F2 ~xg6
0560F2 ~xg7
0570F2 ~xg8
0580F2 ~xg9
0590F2 ~xAns
05A0F2 ~xFIT
05B0F2 ~xHAngle
05C0F2 ~xHFormat
05D0F2 ~xHDigits
05E0F2 ~xRadixMark
05F0F2 ~xIerr
0600F2 ~xTime
0610F2 ~xDate
0620F2 ~xAngle
0630F2 ~xAxes
0640F2 ~xGrid
0650F2 ~xConnect
0660F2 ~xHighRes
0670F2 ~xSimult
0680F2 ~xRecenter
0690F2 ~xInvCursor
06A0F2 ~xLabels
06B0F2 ~xTracing
06C0F2 ~xCoord
06D0F2 ~xXmin
06E0F2 ~xXmax
06F0F2 ~xYmin
0700F2 ~xYmax
0710F2 ~xIndep
0720F2 ~xXcross
0730F2 ~xYcross
0740F2 ~xHTick
0750F2 ~xVTick
0760F2 ~xHzoom
0770F2 ~xVzoom
0780F2 ~xNumStart
0790F2 ~xNumStep
07A0F2 ~xNumType
07B0F2 ~xNumIndep
07C0F2 ~xNumZoom
07D0F2 ~xNumRow
07E0F2 ~xNumCol
07F0F2 ~xNumFont
0800F2 ~xFormat
0810F2 ~xDigits
0820F2 ~xNoteText
0830F2 ~xPage
0840F2 ~xPageNum
0850F2 ~idE1
0860F2 ~idE2
0870F2 ~idE3
0880F2 ~idE4
0890F2 ~idE5
08A0F2 ~idE6
08B0F2 ~idE7
08C0F2 ~idE8
08D0F2 ~idE9
08E0F2 ~idE0
08F0F2 ~xRoot
0900F2 ~xIsect
0910F2 ~xExtremum
0920F2 ~xArea
0930F2 ~xSlope
0940F2 ~xTmin
0950F2 ~xTmax
0960F2 ~xTStep
0970F2 ~xThetaMin
0980F2 ~xThetaMax
0990F2 ~xThetaStep
09A0F2 ~xNmin
09B0F2 ~xNmax
09C0F2 ~xSeqPlot
09D0F2 ~xStatMode
09E0F2 ~xStatPlot
09F0F2 ~xHisWidth
0A00F2 ~xHmin
0A10F2 ~xHmax
0A20F2 ~xS1mark
0A30F2 ~xS2mark
0A40F2 ~xS3mark
0A50F2 ~xS4mark
0A60F2 ~xS5mark
0A70F2 ~xS1fit
0A80F2 ~xS2fit
0A90F2 ~xS3fit
0AA0F2 ~xS4fit
0AB0F2 ~xS5fit
0AC0F2 ~xU1
0AD0F2 ~idU1
0AE0F2 ~xU2
0AF0F2 ~idU2
0B00F2 ~xU3
0B10F2 ~idU3
0B20F2 ~xU4

0B30F2 ~idU4
0B40F2 ~xU5
0B50F2 ~idU5
0B60F2 ~xU6
0B70F2 ~idU6
0B80F2 ~xU7
0B90F2 ~idU7
0BA0F2 ~xU8
0BB0F2 ~idU8
0BC0F2 ~xU9
0BD0F2 ~idU9
0BE0F2 ~xU0
0BF0F2 ~idU0
0C00F2 ~xF1
0C10F2 ~idF1
0C20F2 ~xF2
0C30F2 ~idF2
0C40F2 ~xF3
0C50F2 ~idF3
0C60F2 ~xF4
0C70F2 ~idF4
0C80F2 ~xF5
0C90F2 ~idF5
0CA0F2 ~xF6
0CB0F2 ~idF6
0CC0F2 ~xF7
0CD0F2 ~idF7
0CE0F2 ~xF8
0CF0F2 ~idF8
0D00F2 ~xF9
0D10F2 ~idF9
0D20F2 ~xF0
0D30F2 ~idF0
0D40F2 ~xX1
0D50F2 ~idX1
0D60F2 ~xY1
0D70F2 ~idY1
0D80F2 ~xX2
0D90F2 ~idX2
0DA0F2 ~xY2
0DB0F2 ~idY2
0DC0F2 ~xX3
0DD0F2 ~idX3
0DE0F2 ~xY3
0DF0F2 ~idY3
0E00F2 ~xX4
0E10F2 ~idX4
0E20F2 ~xY4
0E30F2 ~idY4
0E40F2 ~xX5
0E50F2 ~idX5
0E60F2 ~xY5
0E70F2 ~idY5
0E80F2 ~xX6
0E90F2 ~idX6
0EA0F2 ~xY6
0EB0F2 ~idY6
0EC0F2 ~xX7
0ED0F2 ~idX7
0EE0F2 ~xY7
0EF0F2 ~idY7
0F00F2 ~xX8
0F10F2 ~idX8
0F20F2 ~xY8
0F30F2 ~idY8
0F40F2 ~xX9
0F50F2 ~idX9
0F60F2 ~xY9
0F70F2 ~idY9
0F80F2 ~xX0
0F90F2 ~idX0
0FA0F2 ~xY0
0FB0F2 ~idY0
0FC0F2 ~xR1
0FD0F2 ~idR1
0FE0F2 ~xR2
0FF0F2 ~idR2
1000F2 ~xR3
1010F2 ~idR3
1020F2 ~xR4
1030F2 ~idR4
1040F2 ~xR5
1050F2 ~idR5
1060F2 ~xR6
1070F2 ~idR6
1080F2 ~xR7
1090F2 ~idR7
10A0F2 ~xR8
10B0F2 ~idR8
10C0F2 ~xR9
10D0F2 ~idR9
10E0F2 ~xR0
10F0F2 ~idR0
1100F2 ~xD1
1110F2 ~xD2
1120F2 ~xD3
1130F2 ~xD4
1140F2 ~xD5
1150F2 ~xD6
1160F2 ~xD7
1170F2 ~xD8
1180F2 ~xD9
1190F2 ~xD0
11A0F2 ~xH1
11B0F2 ~xH2
11C0F2 ~xH3
11D0F2 ~xH4
11E0F2 ~xH5
11F0F2 ~xS1
1200F2 ~xS2
1210F2 ~xS3
1220F2 ~xS4
1230F2 ~xS5
1240F2 ~xs1

Entry Index

!

#

$

%

&

’

:

?

@

{

+

>

0

1

2

3

4

5

6

7

8

9

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

R

S

T

U

V

W

X

Y

Z