HP39G Entry Reference

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

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

```
040E9      BINT16                  16d 10h
                                   aka: REALOB, SIXTEEN
040F3      BINT17                  17d 11h
                                   aka: SEVENTEEN, 2REAL, REALREAL
040FD      BINT18                  18d 12h
                                   aka: EIGHTEEN
04107      BINT19                  19d 13h
                                   aka: NINETEEN
04111      BINT20                  20d 14h
                                   aka: TWENTY
0411B      BINT21                  21d 15h
                                   aka: TWENTYONE
04125      BINT22                  22d 16h
                                   aka: TWENTYTWO
0412F      BINT23                  23d 17h
                                   aka: TWENTYTHREE
04139      BINT24                  24d 18h
                                   aka: TWENTYFOUR
04143      BINT25                  25d 19h
                                   aka: TWENTYFIVE
0414D      BINT26                  26d 1Ah
                                   aka: REALSYM, TWENTYSIX
04157      BINT27                  27d 1Bh
                                   aka: TWENTYSEVEN
04161      BINT28                  28d 1Ch
                                   aka: TWENTYEIGHT
0416B      BINT29                  29d 1Dh
                                   aka: TWENTYNINE
04175      BINT30                  30d 1Eh
                                   aka: REALEXT, THIRTY
0417F      BINT31                  31d 1Fh
                                   aka: THIRTYONE
04189      BINT32                  32d 20h
                                   aka: THIRTYTWO
04193      BINT33                  33d 21h
                                   aka: THIRTYTHREE
0419D      BINT34                  34d 22h
                                   aka: THIRTYFOUR
041A7      BINT35                  35d 23h
                                   aka: THIRTYFIVE
```

041B1	BINT36	36d 24h aka: TTHIRTYSIX
041BB	BINT37	37d 25h aka: THIRTYSEVEN
041C5	BINT38	38d 26h aka: THIRTYEIGHT
041CF	BINT39	39d 27h aka: THIRTYNINE
041D9	BINT40	40d 28h aka: FORTY, FOURTY
041E3	BINT41	41d 29h aka: FORTYONE
041ED	BINT42	42d 2Ah aka: FORTYTWO
041F7	BINT43	43d 2Bh aka: FORTYTHREE
04201	BINT44	44d 2Ch aka: FORTYFOUR
0420B	BINT45	45d 2Dh aka: FORTYFIVE
04215	BINT46	46d 2Eh aka: FORTYSIX
0421F	BINT47	47d 2Fh aka: FORTYSEVEN
04229	BINT48	48d 30h aka: FORTYEIGHT
04233	BINT49	49d 31h aka: FORTYNINE
0423D	BINT50	50d 32h aka: FIFTY
04247	BINT51	51d 33h aka: FIFTYONE
04251	BINT52	52d 34h aka: FIFTYTWO
0425B	BINT53	53d 35h aka: FIFTYTHREE, STRLIST, THREEFIVE
04265	BINT54	54d 36h aka: FIFTYFOUR
0426F	BINT55	55d 37h aka: FIFTYFIVE

```
0426F        #THREESEVEN          55d 37h
04279        BINT56               56d 38h
                                  aka: FIFTYSIX

04283        BINT57               57d 39h
                                  aka: FIFTYSEVEN

0428D        BINT58               58d 3Ah
                                  aka: FIFTYEIGHT

04297        BINT59               59d 3Bh
                                  aka: FIFTYNINE

042A1        BINT60               60d 3Ch
                                  aka: SIXTY

042AB        BINT61               61d 3Dh
                                  aka: SIXTYONE

042B5        BINT62               62d 3Eh
                                  aka: SIXTYTWO

042BF        BINT63               63d 3Fh
                                  aka: SIXTYTHREE

042BF        BINT3Fh              64d 3Fh
042C9        BINT64               64d 40h
                                  aka: BINT40h, SIXTYFOUR, YHI

042D3        BINT65               65d 41h
                                  aka: ARRYREAL

042D3        BINT_65d             65d 41h
042D3        SIXTYFIVE            65d 41h
042DD        BINT66               66d 42h
                                  aka: FOURTWO

042DD        SIXTYSIX             66d 42h
042E7        BINT67               67d 43h
                                  aka: FOURTHREE

042F1        BINT68               68d 44h
                                  aka: SIXTYEIGHT

042F1        2ARRY                68d 44h
042FB        BINT69               69d 45h
                                  aka: FOURFIVE

04305        BINT70               70d 46h
                                  aka: SEVENTY

04305        ARRYID               70d 46h
0430F        BINT71               71d 47h
0430F        SEVENTYONE           71d 47h
04319        BINT72               72d 48h
```

```
04319      SEVENTYTWO           72h 48d
04323      BINT73               73d 49h
04323      SEVENTYTHREE         73d 49h
0432D      BINT74               74d 4Ah
                                aka: SEVENTYFOUR
04337      BINT75               75d 4Bh
04337      SEVENTYFIVE          75d 4Bh
04341      BINT76               76d 4Ch
04341      SEVENTYSIX           76d 4Ch
0434B      SEVENTYSEVEN         77d 4Dh
0434B      BINT77               77d 4Dh
04355      SEVENTYEIGHT         78d 4Eh
04355      BINT78               78d 4Eh
0435F      BINT79               79d 4Fh
                                aka: SEVENTYNINE
04369      BINT80               80d 50h
                                aka: EIGHTY
04369      LISTOB               80d 50h
04373      BINT81               81d 51h
                                aka: EIGHTYONE, LISTREAL
0437D      BINT82               82d 52h
                                aka: LISTCMP
04387      BINT83               83d 53h
                                aka: FIVETHREE
04391      BINT84               84d 54h
                                aka: FIVEFOUR
0439B      BINT85               85d 55h
                                aka: 2LIST
043A5      BINT86               86d 56h
                                aka: FIVESIX
043A5      LISTID               86d 56h
043AF      BINT87               87d 57h
                                aka: LISTLAM
043B9      BINT88               88d 58h
043B9      EIGHTYEIGHT          88d 58h
043C3      BINT89               89d 59h
043C3      EIGHTYNINE           89d 59d
043CD      BINT90               90d 5Ah
043CD      NINETY               90d 5Ah
```

```
043D7       BINT91                    91d 5Bh
                                      aka: BINT_91d
043E1       BINT92                    92d 5Ch
043E1       NINETYTWO                 92d 5Ch
043EB       BINT93                    93d 5Dh
043EB       NINETYTHREE               93d 5Dh
043F5       BINT94                    94d 5Eh
043F5       NINETYFOUR                94d 5Eh
043FF       BINT95                    95d 5Fh
043FF       NINETYFIVE                95d 5Fh
04409       BINT96                    96d 60h
                                      aka: BINT_96d
04413       BINT97                    97d 61h
                                      aka: IDREAL
0441D       BINT98                    98d 62h
0441D       IDCMP                     98d 62h
0441D       NINETYEIGHT               98d 62h
04427       BINT99                    99d 63h
04427       NINETYNINE                99d 63h
04431       BINT100                   100d 64h
                                      aka: ONEHUNDRED
04431       IDARRY                    100d 64h
0443B       BINT101                   101d 65h
0443B       IDLIST                    101d 65h
0443B       SIXFIVE                   101d 65h
04445       BINT102                   102d 66h
0444F       BINT103                   103d 67h
04459       BINT104                   104d 68h
04463       BINT105                   105d 69h
0446D       BINT106                   106d 6Ah
04477       BINT107                   107d 6Bh
04481       BINT108                   108d 6Ch
0448B       BINT109                   109d 6Dh
04495       BINT110                   110d 6Eh
0449F       BINT111                   111d 6Fh
                                      aka: char
044A9       BINT112                   112d 70h
044B3       BINT113                   113d 71h
044B3       LAMREAL                   113d 71h
```

```
044BD        BINT114              114d 72h
044BD        BINT_114             114d 72hd
044C7        BINT115              115d 73h
                                  aka: BINT_115d
044D1        BINT116              116d 74h
                                  aka: BINT_116d
044DB        BINT117              117d 75h
044E5        LAMLIST              117d 75h
044DB        BINT_117d            117d 75h
044E5        BINT118              118d 76h
044EF        BINT119              119d 77h
044F9        BINT120              120d 78h
04503        BINT121              121d 79h
0450D        BINT122              122d 7Ah
                                  aka: BINT_122d
04517        BINT123              123d 7Bh
04521        BINT124              124d 7Ch
0452B        BINT125              125d 7Dh
04535        BINT126              126d 7Eh
0453F        BINT127              127d 7Fh
```

2.1.2 Built-in BINTS 127-255

```
04549        BINT128              128d 80h
                                  aka: BINT80h
04549        BINT_128d            128d 80h
04553        BINT129              129d 81h
0455D        BINT130              130d 82h
                                  aka: BINT130d, BINT_130d, XHI-1
04567        BINT131              131d 83h
                                  aka: BINT_131d, BINT131d, XHI
04571        BINT132              132d 84h
0457B        BINT133              133d 85h
04585        BINT134              134d 86h
0458F        BINT135              135d 87h
04599        BINT136              136d 86h
045A3        BINT137              137d 89h
045AD        BINT138              138d 8Ah
045B7        BINT139              139d 8Bh
045C1        BINT140              140d 8Ch
```

```
045CB       BINT141               141d 8Dh
045D5       BINT142               142d 8Eh
045DF       BINT143               143d 8Fh
045E9       BINT144               144d 90h
045F3       SYMBREAL              145d 91h
045F3       BINT145               145d 91h
045FD       SYMBCMP               146d 92h
045FD       BINT146               146d 92h
04607       BINT147               147d 93h
04611       BINT148               148d 94h
0461B       BINT149               149d 95h
04625       BINT150               150d 96h
0462F       BINT151               151d 97h
04639       BINT152               152d 98h
04643       BINT153               153d 99h
0464D       SYMBSYM               154d 9Ah
0464D       BINT154               154d 9Ah
04657       BINT155               155d 9Bh
04661       BINT156               156d 9Ch
0466B       BINT157               157d 9Dh
04675       SYMBUNIT              158d 9Eh
04675       BINT158               158d 9Eh
0467F       BINT159               159d 9Fh
0467F       backup                159d 9Fh
04689       SYMOB                 160d A0h
04689       BINT160               160d A0h
04693       SYMREAL               161d A1h
04693       BINT161               161d A1h
0469D       SYMCMP                162d A2h
0469D       BINT162               162d A2h
046A7       BINT163               163d A3h
046B1       SYMARRY               164d A4h
046B1       BINT164               164d A4h
046BB       SYMLIST               165d A5h
046BB       BINT165               165d A5h
046C5       SYMID                 166d A6h
046C5       BINT166               166d A6h
046CF       SYMLAM                167d A7h
```

```
046CF        BINT167              167d A7h
046D9        BINT168              168d A8h
046E3        SYMSYMB              169d A9h
046E3        BINT169              169d A9h
046ED        SYMSYM               170d AAh
046ED        BINT170              170d AAh
046F7        BINT171              171d ABh
04701        BINT172              172d ACh
0470B        BINT173              173d ADh
04715        SYMEXT               174d AEh
04715        BINT174              174d AEh
0471F        BINT175              175d AFh
04729        BINT176              176d B0h
04733        HXSREAL              177d B1h
04733        BINT177              177d B1h
0473D        BINT178              178d B2h
04747        BINT179              179d B3h
04751        BINT180              180d B4h
0475B        BINT181              181d B5h
04765        BINT182              182d B6h
0476F        BINT183              183d B7h
04779        BINT184              184d B8h
04783        BINT185              185d B9h
0478D        BINT186              186d BAh
04797        2HXS                 187d BBh
04797        BINT187              187d BBh
047A1        BINT188              188d BCh
047AB        BINT189              189d BDh
047B5        BINT190              190d BEh
047BF        BINT191              191d BFh
047C9        BINTC0h              192d C0h
047C9        BINT192              192d C0h
047D3        BINT193              193d C1h
047DD        BINT194              194d C2h
047E7        BINT195              195d C3h
047F1        BINT196              196d C4h
047FB        BINT197              197d C5h
04805        BINT198              198d C6h
```

```
0480F      BINT199              199d C7h
04819      BINT200              200d C8h
04823      BINT201              201d C9h
0482D      BINT202              202d CAh
04837      BINT203              203d CBh
04841      2GROB                204d CCh
04841      BINT204              204d CCh
0484B      BINT205              205d CDh
04855      BINT206              206d CEh
0485F      BINT207              207d CFh
04869      TAGGEDANY            208d D0h
04869      BINT208              208d D0h
04873      BINT209              209d D1h
0487D      BINT210              210d D2h
04887      BINT211              211d D3h
04891      BINT212              212d D4h
0489B      BINT213              213d D5h
048A5      BINT214              214d D6h
048AF      BINT215              215d D7h
048B9      BINT216              216d D8h
048C3      BINT217              217d D9h
048CD      BINT218              218d DAh
048D7      BINT219              219d DBh
048E1      BINT220              220d DCh
048EB      BINT221              221d DDh
048F5      BINT222              222d DEh
048FF      BINT223              223d DFh
04909      BINT224              224d E0h
04913      EXTREAL              225d E1h
04913      BINT225              225d E1h
0491D      BINT226              226d E2h
04927      BINT227              227d E3h
04931      BINT228              228d E4h
0493B      BINT229              229d E5h
04945      BINT230              230d E6h
0494F      BINT231              231d E7h
04959      BINT232              232d E8h
04963      BINT233              233d E9h
```

```
0496D      EXTSYM                234d EAh
0496D      UNITSYM               234d EAh
0496D      BINT234               234d EAh
04977      BINT235               235d EBh
04981      BINT236               236d ECh
0498B      BINT237               237d EDh
04995      2EXT                  238d EEh
04995      BINT238               238d EEh
0499F      BINT239               239d EFh
049A9      ROMPANY               240d F0h
049A9      BINT240               240d F0h
049B3      BINT241               241d F1h
049BD      BINT242               242d F2h
049C7      BINT243               243d F3h
049D1      BINT244               244d F4h
049DB      BINT245               245d F5h
049E5      BINT246               246d F6h
049EF      BINT247               247d F7h
049F9      BINT248               248d F8h
04A03      BINT249               249d F9h
04A0D      BINT250               250d FAh
04A17      BINT251               251d FBh
04A21      BINT252               252d FCh
04A2B      BINT253               253d FDh
04A35      BINT254               254d FEh
04A3F      BINT255               255d FFh
04A3F      BINT255d              255d FFh
```

2.1.3 Built-in BINTS 256-

```
04A49      REALOBOB              256d 100h
04A49      BINT256               256d 100h
11C0D      #_102                 258d 102h
11C0D      #_258_d               258d 102h
11C17      #SyntaxErr            262d 106h
11C21      BINT_263d             263d 107h
11C2B      REALREALOB            272d 110h
11C35      3REAL                 273d 111h
```

```
11C3F        Err#Kill               291d 123h
11C3F        #_291_d                291d 123h
11C3F        #_123                  291d 123h
11C49        #_124                  292d 124h
11C49        Err#NoLstStk           292d 124h
11C49        #_292_d                292d 124h
11C53        #NoRoomForSt           305d 131h
11C53        BINT_305d              305d 131h
11C5D        BINT_306d              306d 132h
11C67        REALSTRSTR             307d 133h
11C67        BINT_307d              307d 133h
11CCB        Err#Cont               318d 13Eh
11CD5        INTEGER337             337d 151h
11CDF        CMPOBOB                512d 200h
11CE9        Err#NoLstArg           517d 205h
11CE9        #_205                  517d 205h
11CE9        #_517_d                517d 205h
11CF3        STRREALREAL            785d 311h
11CFD        ARRYREALREAL           1041d 411h
11D07        ARRYREALCMP            1042d 412h
11D11        3ARRY                  1092d 444h
11D1B        ARRYLISTREAL           1105d 451h
11D25        ARRYLISTCMP            1106d 452h
11D2F        LISTREALOB             1296d 510h
11D39        LISTREALREAL           1297d 511h
11D43        LISTLISTOB             1360d 550h
11D4D        IDREALOB               1552d 610h
11D57        IDLISTOB               1616d 650h
11D61        FSTMACROROM#           1792d 700h
11D61        LAMANYANY              1792d 700h
11D89        PROGIDREAL             2145d 861h
11D93        PROGIDCMP              2146d 862h
11D9D        PROGIDLIST             2149d 865h
11DA7        PROGIDEXT              2158d 86Eh
11DB1        ATTNERR                2563d A03h
11DCF        SYMREALREAL            2577d A11h
11DD9        SYMREALCMP             2578d A12h
11DE3        SYMREALSYM             2586d A1Ah
```

```
11DED      SYMCMPREAL              2593d A21h
11DF7      SYMCMPCMP               2594d A22h
11E01      SYMCMPSYM               2602d A2Ah
11E0B      SYMIDREAL               2657d A61h
11E15      SYMIDCMP                2658d A62h
11E1F      SYMIDLIST               2661d A65h
11E29      SYMIDEXT                2670d A6Eh
11E33      SYMSYMREAL              2721d AA1h
11E3D      SYMSYMCMP               2722d AA2h
11E47      3SYM                    2730d AAAh
11E51      XFERFAIL                3078d C06h
11E5B      PROTERR                 3079d C07h
11E65      InvalServCmd            3080d C08h
11E6F      Connecting              3082d C0Ah
11E79      Retry                   3083d C0Bh
11E83      #CAlarmErr              3583d DFFh
11E8D      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
11E97      #EXITERR                458752d 70000h
11EC9      MINUSFIVE               1048571d FFFFBh
11EBF      MINUSFOUR               1048572d FFFFCh
11EB5      MINUSTHREE              1048573d FFFFDh
11EAB      MINUSTWO                1048574d FFFFEh
11EA1      MINUSONE                1048575d FFFFFh
```

2.1.4 Pushing Several BINTs

```
0E996      ZEROZERO                ( → #0 #0 )
```

10358	ONEONE	(→ #1 #1)
		aka: `ONEDUP`
0E0E0	DROPZERO	(ob → #0)
0E0F9	2DROP00	(ob ob → #0 #0)
0E4C1	DROPONE	(ob → #1)
1031C	DUPZERO	(ob → ob ob #0)
10330	DUPONE	(ob → ob ob #1)
1036C	DUPTWO	(ob → ob ob #2)
10344	SWAPONE	(ob ob' → ob' ob #1)
0F714	ZEROSWAP	(ob → #0 ob)
0F95A	ZEROOVER	(ob → ob #0 ob)
0FE17	ZEROFALSE	(→ #0 F)
0F741	ONESWAP	(ob → #1 ob)
0FE2B	ONEFALSE	(→ #1 F)

2.1.5 Conversion

13244	COERCE	(% → #)
0F5BB	COERCEDUP	(% → # #)
0F755	COERCESWAP	(ob % → # ob)
13231	%ABSCOERCE	(% → #)
0620A	CHR>#	(chr → #)

2.1.6 Arithmetic Functions

03DBC	#+	(# #' → #+#')
03DEF	#1+	(# → #+1)
03E2D	#2+	(# → #+2)
0E151	#3+	(# → #+3)
0E156	#4+	(# → #+4)
0E15B	#5+	(# → #+5)
0E16A	#8+	(# → #+8)
03DE0	#-	(# #' → #-#')
03E0E	#1-	(# → #-1)
03E4E	#2-	(# → #-2)
0E133	#3-	(# → #-3)
03F12	#*	(# #' → #*#')
13A13	#*OVF	(# #' → #*#')
		$0 \leq result \leq FFFFF$

03E6F	#2*	(# → #*2)
03F47	#/	(# #’ → #r #q)
03E8E	#2/	(# → #/2)
		Rounded down.
0E0A6	#-#2/	(# #’ → (#-#’)/2)
0E350	#+DUP	(# #’ → #+#’ #+#’)
0F6D8	#+SWAP	(ob # #’ → #+#’ ob)
0F932	#+OVER	(ob # #’ → ob #+#’ ob)
0F6EC	#-SWAP	(ob # #’ → #-#’ ob)
0F946	#-OVER	(ob # #’ → ob #-#’ ob)
0F700	#1+SWAP	(ob # → #+1 ob)
0F8BA	#1-ROT	(ob ob’ # → ob’ #-1 ob)
0F728	#1-1SWAP	(# → 1 #-1)
		Returns the bint ONE and the result.
0E466	DUP#1+	(# → # #+1)
0FFE3	2DUP#+	(# #’ → # #’ #+#’)
		aka: DUP3PICK#+
100BF	DROP#1-	(# ob → #-1)
0E30F	SWAP#-	(# #’ → #’-#)
0E47F	SWAP#1+	(meta ob → meta&ob)
		aka: SWP1+
14E94	’RSWP1+	(# → nob #+1)
		nob is the next object in the runstream.
100AB	SWAP#1-	(# ob → ob #-1)
10083	SWAPOVER#-	(# #’ → #’ #-#’)
1000B	OVER#+	(# #’ → # #’+#)
1005B	OVER#-	(# #’ → # #’-#)
108B2	OVER#1-	(# #’ → # #’ #’’)
0FFF7	ROT#+	(# ob #’ → ob #’+#)
10047	ROT#-	(# ob #’ → ob #’-#)
10097	ROT#1+	(# ob ob’ → ob ob’ #+1)
150FC	ROT#1+UNROT	(# ob ob’ → #+1 ob ob’)
0F6A6	ROT#+SWAP	(# ob #’ → #’+# ob)
		aka: ROT+SWAP
1001F	3PICK#+	(# ob #’ → # ob #’+#)
10033	4PICK#+	(# ob1 ob2 #’ → # ob1 ob2 #’+#)
0F6BF	4PICK#+SWAP	(# ob1 ob2 #’ → # ob1 #’+# ob2)
		aka: 4PICK+SWAP
0E065	#MIN	(# #’ → #’’)
0E071	#MAX	(# #’ → #’’)
03EE1	#AND	(# #’ → #’’)
		Bitwise AND.

2.1.7 Tests

03D19	#=	(# #' → flag)
03D4E	#<>	(# #' → flag)
03CE4	#<	(# #' → flag)
37101	#<=	(# #' → flag)
03D83	#>	(# #' → flag)
370E8	#>=	(# #' → flag)
03CC7	#0<>	(# → flag)
03CA6	#0=	(# → flag)
0DE61	#1<>	(# → flag)
0DE52	#1=	(# → flag)
0FFA7	#2<>	(# → flag)
0DE45	#2=	(# → flag)
0DE34	#3=	(# → flag)
0FF93	#5=	(# → flag)
0FF52	#<3	(# → flag)
0FFCF	#>1	(# → flag) aka: ONE#>
0E416	2DUP#<	(# #' → # #' flag)
0E44C	2DUP#>	(# #' → # #' flag)
0FC7D	ONE_EQ	(# → flag) Uses EQ test.
0DDBC	OVER#=	(# #' → # flag)
0E430	2DUP#=	(# #' → # #' flag)
0FF2A	OVER#0=	(# #' → # #' flag)
0DE11	DUP#0=	(# → # flag)
0FF3E	OVER#<	(# #' → # flag)
0DE70	DUP#1=	(# → # flag)
0FFBB	OVER#>	(# #' → # flag)
0DE7F	DUP#0<>	(# → # flag)
0FF66	DUP#<7	(# → # flag) Returns TRUE if the argument is smaller than #7.
0FF0C	2#0=OR	(# # → flag) Returns TRUE if either argument is zero.

2.2 Real Numbers

2.2.1 Built-in Real Numbers

```
120A6      %-MAXREAL              -9.99E499
1204D      %-9                    -9
12038      %-8                    -8
12023      %-7                    -7
1200E      %-6                    -6
11FF9      %-5                    -5
11FE4      %-4                    -4
11FCF      %-3                    -3
11FBA      %-2                    -2
11FA5      %-1                    -1
120D0      %-MINREAL              -1E-499
11ED3      %0                     0
120BB      %MINREAL               1E-499
121E4      %.5                    .5
121F9      %-.5                   -.5
11EE8      %1                     1
11EFD      %2                     2
121CF      %e                     e
11F12      %3                     3
12062      %PI                    π
11F27      %4                     4
11F3C      %5                     5
11F51      %6                     6
11F66      %7                     7
11F7B      %8                     8
11F90      %9                     9
1220E      %10                    10
12277      %15                    15
12349      %25                    25
12388      %180                   180
1239D      %200                   200
123C7      %400                   400
123B2      %360                   360
12091      %MAXREAL               9.99E499
```

2.2.2 Built-in Extended Real Numbers

```
120E5       %%0                     0
12181       %%.1                    0.1
1219B       %%.5                    0.5
120FF       %%1                     1
12119       %%2                     2
12133       %%3                     3
12077       %%PI                    π
1214D       %%4                     4
12167       %%5                     5
121B5       %%10                    10
```

2.2.3 Conversion

```
1B3FC       %>%%                    ( % → %% )
0F769       %>%%SWAP                ( ob % → %% ob )
1B3EB       %%>%                    ( %% → % )
13319       UNCOERCE                ( # → % )
10411       UNCOERCE%%              ( # → %% )
06530       C%>%                    ( C% → %re %im )
```

2.2.4 Real Functions

```
1B895       %+                      ( % %' → %+%' )
1B8A2       %-                      ( % %' → %-%' )
1B8DD       %*                      ( % %' → %*%' )
1B91F       %/                      ( % %' → %/%' )
1B991       %^                      ( % %' → %^%' )
1B821       %ABS                    ( % → %' )
1B841       %CHS                    ( % → -% )
1B7F8       %SGN                    ( % → -1/0/1 )
```

$$1BA2A \quad \text{\%SQRT} \quad\quad (\% \rightarrow \sqrt{a}\%)$$

```
1BA50       %EXP                    ( % → e^% )
1BA63       %EXPM1                  ( % → e^%-1 )
1BA8F       %LN                     ( % → LN% )
1BAC8       %LNP1                   ( % → LN(%+1) )
```

1BAA2	%LOG	(% → LOG%)
1BADB	%ALOG	(% → 10^%)
1BB10	%SIN	(% → SIN%)
1BB61	%COS	(% → COS%)
1BBB2	%TAN	(% → TAN%)
1BBE2	%ASIN	(% → ASIN%)
1BC12	%ACOS	(% → ACOS%)
1BC42	%ATAN	(% → ATAN%)
1BCCF	%SINH	(% → SINH%)
1BCFB	%COSH	(% → COSH%)
1BD0E	%TANH	(% → TANH%)
1BD21	%ASINH	(% → ASINH%)
1BD34	%ACOSH	(% → ACOSH%)
1BD47	%ATANH	(% → ATANH%)
1B851	%MANTISSA	(% → %mant)
1BD5A	%EXPONENT	(% → %expn)
1BE6E	%FP	(% → %frac)
1BE81	%IP	(% → %int)
1BEA7	%FLOOR	(% → %maxint <=%)
1BE94	%CEIL	(% → %minint >=%)
1BAFD	%MOD	(% %' → %rem)
1BC59	%ANGLE	(%x %y → %ang)
1C44A	RNDXY	(% %places → %')
1C45E	TRCXY	(% %places → %')
1BD83	%COMB	(% %' → COMB(%,%'))
1BD96	%PERM	(% %' → PERM(%,%'))
1BD6D	%NFACT	(% → %!) Calculates factorial of number.
1BFE5	%FACT	(% → gamma(%+1)) Calculates gamma(x+1).
1B9A2	%NROOT	(% %n → %') Calculates the %nth root of the real number. Equivalent to user function XROOT.
1B62F	%MIN	(% %' → %lesser)
1B616	%MAX	(% %' → %greater)
0F65B	%MAXorder	(% %' → %max %min)
1BEE3	%RAN	(→ %random) Returns next random number.

1BF65	%RANDOMIZE	(%seed →)
		System level RDZ: seeds the random number generator.
1B8EA	%OF	(% %' → %'/% * 100)
1B92C	%T	(% %' → %pctotal)
1B951	%CH	(% %' → %pcchange)
1B543	%D>R	(%deg → %rad)
1B576	%R>D	(%rad → %deg)
1C3AF	%REC>%POL	(%r %ang → %x %y)
1C3DC	%POL>%REC	(%x %y → %r %ang)
1C413	%SPH>%REC	(%r %ang %ph → %x %y %z)

2.2.5 Extended Real Functions

10439	SWAP%%/	(%% %%' → %%'')
103FD	%%/>%	(%% %%' → %)

2.2.6 Tests

1B7E2	%=	(% %' → flag)
1B7ED	%<>	(% %' → flag)
1B792	%<	(% %' → flag)
1B7D7	%<=	(% %' → flag)
1B7AB	%>	(% %' → flag)
1B7C1	%>=	(% %' → flag)
1B68C	%0=	(% → flag)
10425	DUP%0=	(% → flag)
1B6F0	%0<>	(% → flag)
		Can be used to change a user flag into a system flag.
1B659	%0<	(% → flag)
1B6BA	%0>	(% → flag)
1B718	%0>=	(% → flag)

2.3 Complex Numbers

2.3.1 Conversion

```
0642E       %>C%                      ( %re %im → C% )
0FB91       SWAP%>C%                  ( %im %re → C% )
```

2.4 Character Strings

2.4.1 Built-in Characters

```
10A1C       CHR_00                    '\00', CHR 0d 00h
                                      The NULL character.
10C6F       CHR_Newline               '\0a', CHR 10d 0Ah
10A23       CHR_...                   '...', CHR 31d 1Fh
10C8B       CHR_Space                 ' ', CHR 32d 20h
                                      The space character.
10A31       CHR_DblQuote              '"', CHR 34d 22h
10A38       CHR_#                     '#', CHR 35d 23h
10C68       CHR_LeftPar               '(', CHR 40d 28h
10C7D       CHR_RightPar              ')', CHR 41d 29h
10A3F       CHR_*                     '*', CHR 42d 2Ah
10A46       CHR_+                     '+', CHR 43d 2Bh
10A4D       CHR_,                     ',', CHR 44d 2Ch
10A54       CHR_-                     '-', CHR 45d 2Dh
10A5B       CHR_.                     '.', CHR 46d 2Eh
10A62       CHR_/                     '/', CHR 47d 2Fh
10A69       CHR_0                     '0', CHR 48d 30h
10A70       CHR_1                     '1', CHR 49d 31h
10A77       CHR_2                     '2', CHR 50d 32h
10A7E       CHR_3                     '3', CHR 51d 33h
10A85       CHR_4                     '4', CHR 52d 34h
10A8C       CHR_5                     '5', CHR 53d 35h
10A93       CHR_6                     '6', CHR 54d 36h
10A9A       CHR_7                     '7', CHR 55d 37h
10AA1       CHR_8                     '8', CHR 56d 38h
10AA8       CHR_9                     '9', CHR 57d 39h
10AAF       CHR_:                     ':', CHR 58d 3Ah
10AB6       CHR_;                     ';', CHR 59d 3Bh
10ABD       CHR_<                     '<', CHR 60d 3Ch
10AC4       CHR_=                     '=', CHR 61d 3Dh
10ACB       CHR_>                     '>', CHR 62d 3Eh
10AD2       CHR_A                     'A', CHR 65d 41h
```

```
10AD9        CHR_B                  ’B’, CHR 66d 42h
10AE0        CHR_C                  ’C’, CHR 67d 43h
10AE7        CHR_D                  ’D’, CHR 68d 44h
10AEE        CHR_E                  ’E’, CHR 69d 45h
10AF5        CHR_F                  ’F’, CHR 70d 46h
10AFC        CHR_G                  ’G’, CHR 71d 47h
10B03        CHR_H                  ’H’, CHR 72d 48h
10B0A        CHR_I                  ’I’, CHR 73d 49h
10B11        CHR_J                  ’J’, CHR 74d 4Ah
10B18        CHR_K                  ’K’, CHR 75d 4Bh
10B1F        CHR_L                  ’L’, CHR 76d 4Ch
10B26        CHR_M                  ’M’, CHR 77d 4Dh
10B2D        CHR_N                  ’N’, CHR 78d 4Eh
10B34        CHR_O                  ’O’, CHR 79d 4Fh
10B3B        CHR_P                  ’P’, CHR 80d 50h
10B42        CHR_Q                  ’Q’, CHR 81d 51h
10B49        CHR_R                  ’R’, CHR 82d 52h
10B50        CHR_S                  ’S’, CHR 83d 53h
10B57        CHR_T                  ’T’, CHR 84d 54h
10B5E        CHR_U                  ’U’, CHR 85d 55h
10B65        CHR_V                  ’V’, CHR 86d 56h
10B6C        CHR_W                  ’W’, CHR 87d 57h
10B73        CHR_X                  ’X’, CHR 88d 58h
10B7A        CHR_Y                  ’Y’, CHR 89d 59h
10B81        CHR_Z                  ’Z’, CHR 90d 5Ah
10C99        CHR_[                  ’[’, CHR 91d 5Bh
10CA0        CHR_]                  ’]’, CHR 93d 5Dh
10C92        CHR_UndScore           ’_’, CHR 95d 5Fh
10B88        CHR_a                  ’a’, CHR 97d 61h
10B8F        CHR_b                  ’b’, CHR 98d 62h
10B96        CHR_c                  ’c’, CHR 99d 63h
10B9D        CHR_d                  ’d’, CHR 100d 64h
10BA4        CHR_e                  ’e’, CHR 101d 65h
10BAB        CHR_f                  ’f’, CHR 102d 66h
10BB2        CHR_g                  ’g’, CHR 103d 67h
10BB9        CHR_h                  ’h’, CHR 104d 68h
10BC0        CHR_i                  ’i’, CHR 105d 69h
10BC7        CHR_j                  ’j’, CHR 106d 6Ah
```

```
10BCE      CHR_k                     'k', CHR 107d 6Bh
10BD5      CHR_l                     'l', CHR 108d 6Ch
10BDC      CHR_m                     'm', CHR 109d 5Dh
10BE3      CHR_n                     'n', CHR 110d 6Eh
10BEA      CHR_o                     'o', CHR 111d 6Fh
10BF1      CHR_p                     'p', CHR 112d 70h
10BF8      CHR_q                     'q', CHR 113d 71h
10BFF      CHR_r                     'r', CHR 114d 72h
10C06      CHR_s                     's', CHR 115d 73h
10C0D      CHR_t                     't', CHR 116d 74h
10C14      CHR_u                     'u', CHR 117d 75h
10C1B      CHR_v                     'v', CHR 118d 76h
10C22      CHR_w                     'w', CHR 119d 77h
10C29      CHR_x                     'x', CHR 120d 78h
10C30      CHR_y                     'y', CHR 121d 79h
10C37      CHR_z                     'z', CHR 122d 7Ah
10CA7      CHR_{                     '{', CHR 123d 7Bh
10CAE      CHR_}                     '{', CHR 125d 7Dh
```

```
10C53      CHR_Angle                 '∠', CHR 128d 80h
10C61      CHR_Integral              '∫', CHR 132d 84h
10C5A      CHR_Deriv                 '∂', CHR 136d 88h
10C3E      CHR_->                    '→', CHR 141d 8Dh
10C45      CHR_<<                    '≪', CHR 171d ABh
10C4C      CHR_>>                    '≫', CHR 187d BBh
10C76      CHR_Pi                    'π', CHR 135d 87h
10C84      CHR_Sigma                 'Σ', CHR 133d 85h
10CB5      CHR_<=                    '≤', CHR 137d 89h
10CBC      CHR_>=                    '≥', CHR 138d 8Ah
10CC3      CHR_<>                    '≠', CHR 139d 8Bh
10A2A      CHR_'
```

2.4.2 Built-in Strings

```
05D98      NULL$                     ""
                                     Empty string.
124E0      SPACE$                    " "
                                     aka: tok_
1249E      14SPACES$                 "              "
                                     String of 14 spaces.
```

```
124C4      NEWLINE$              "\0a"
                                 Newline.
16C46      CRLF$                 "\0d\0a"
                                 Carriage return and line feed.
12540      toklparen             "("
1254C      tokrparen             ")"
123F6      tok[                  "["
123DC      tok]                  "]"
12402      tok{                  "{"
1240E      tok}                  "}"
12462      tok<<                 "≪"
12456      tok>>                 "≫"
10D8D      $_LRParens            "()"
10D55      $_{}                  "{}"
10D47      $_<<>>                "≪≫"
10D71      $_''                  "''"
                                 Two single quotes.
10D7F      $_::                  "::"
10D9B      $_2DQ                 """"
                                 Two double quotes.
1251C      tok,                  ","
12510      tok'                  "'"
                                 One single quote.
12588      tok-                  "-"
12528      tok.                  "."
12594      tok=                  "="
125D8      tok0                  "0"
125E4      tok1                  "1"
125F0      tok2                  "2"
125FC      tok3                  "3"
12608      tok4                  "4"
12614      tok5                  "5"
12620      tok6                  "6"
1262C      tok7                  "7"
12534      tok;                  ";"
12638      tok8                  "8"
12644      tok9                  "9"
1244A      tokESC                "\1B"
                                 Escape character.
1246E      tokexponent           "E"
```

12504	tokquote	"""
		One double quote.
1241A	toksharp	"#"
12432	tok$	"$"
1243E	tok&	"&"
12564	tok*	"*"
1257C	tok+	"+"
12570	tok/	"/"
1247A	tokanglesign	"$\angle$"
125AC	tokDER	"∂"
124D0	$DER	"der"
12486	tokSIGMA	"Σ"
125A0	tokSQRT	"$\sqrt{a}$"
12426	tokuscore	"_"
12492	tokWHERE	"|"
12558	tok^	"^"
17357	tok:	":"
10DE9	$_RAD	"RAD"
10DF9	$_GRAD	"GRAD"
10D37	$_XYZ	"XYZ"
10D27	$_R<Z	"R$\angle$Z"
		"R<angle>Z"
10D17	$_R<<	"R$\angle\angle$"
		"R<angle><angle>"
10DBB	$_EXIT	"EXIT"
10DA9	$_ECHO	"ECHO"
10DCD	$_Undefined	"Undefined"
125B8	tokCTGROB	"GROB"
125CA	tokCTSTR	"C$"
124EC	tokUNKNOWN	"UNKNOWN"

2.4.3 Built-in Strings with Stack Manipulation

0F633	NULL$SWAP	(ob → $ ob)
		NULL$, then SWAP.
05550	DROPNULL$	(ob → NULL$)
		DROP then NULL$.
05569	TWODROPNULL$	(ob ob' → NULL$)
		2DROP then NULL$.

2.4.4 Conversion

21A2C	#>$	(# → $)
		Creates string from the bint (decimal).
063C9	ID>$	(id/lam → $)
		Converts identifier into string.

2.4.5 Management

0622E	#>CHR	(# → chr)
		Returns character with the specified ASCII code.
109DD	CHR>$	(chr → $* Strings)
		Converts a character into a string.
05DEF	LEN$	($ → #length)
		Returns length in bytes.
0E336	DUPLEN$	($ → $ #)
		DUP then LEN$.
0FA79	NEWLINE$&$	($ → "$\0a")
		Appends newline character to string. aka: NEWLINE&$
058A6	CAR$	($ → chr)
		($ → "")
		Returns first character of string as a string, or NULL$ for null string.
05925	CDR$	($ → $')
		Returns string without first character, or NULL$ for null string.
0EE54	POS$	($ $find start# → #pos)
		($ $find start# → #0)
		Search for $find in $search, starting at position #start. Returns position of $find or 0 if not found. Same entry as POSCHR.
0EE54	POSCHR	($search chr #start → #pos)
		($search chr #start → #0)
		Same entry as <REF>POS$.
0EE60	POS$REV	($ $find #limit → #pos)
		($ $find #limit → #0)
		Searches backwards from #limit to #1. Same entry as <REF>POSCHRREV.
0EE60	POSCHRREV	($seach chr #start → #pos)
		($seach chr #start → #0)
		Same entry as <REF>POS$REV.
05EEC	SUB$	($ #start #end → $')
		Returns substring between specified positions.

0FB2D	#1-SUB$	($ #start #end+#1 → $') Does #1- and then SUB$.
0FB41	1_#1-SUB$	($ #end → $') Returns substring with the first #end characters. aka: 1_#1-SUB
0FB55	LAST$	($ #start → $') Returns substring from the specified start position to the end (inclusive).
0FB69	#1+LAST$	($ #start-#1 → $') Returns substring from the specified start position to the end (exclusive).
0F647	SUB$SWAP	(ob $ # #' → $' ob) SUB$ then SWAP.
14370	SUB$1#	($ #pos → #') Returns bint with ASCII code of character at the specified position.
0D7E5	EXPAND	(hxs #nibs → hxs') Appends #nibs zero nibbles to the hxs.
05947	&$	($ $' → $+$') Concatenates two strings.
107E5	&$SWAP	(ob $ $' → $+$' ob) &$ then SWAP.
0DF21	!append$	($ $' → $+$') Tries &$, if not enough memory does !!append$?.
0F809	!append$SWAP	(ob $ $' → $+$' ob) !append$ then SWAP.
0DF4B	!!append$	($ $' → $+$') Tries appending "in place".
05A0F	>H$	($ chr → $') Prepends character to string
05AA2	>T$	($ chr → $') Appends character to string.
0F48A	APPEND_SPACE	($ → $') Appends space to string.
0DE9A	SWAP&$	($ $' → $'+$) Concatenates two strings.

2.4.6 Parsing Strings

0C558	!*triand	(T T →) (F T → F T <SEMI>)
0C5DD	tok8cktrior	($1 $1 → :: $1 <Ob1> ;) ($1 $2 → :: $1 <Ob2> <Rest> ;)

0C5E9	tok8trior	(GNT data $1 $1 → :: GNT data GetNextToken ;)
		(GNT data $1 $2 → :: $1 <Ob1> <Rest> ;)
3F068	nultrior	(NULL$ → :: ;)
		($ → :: $ <Ob1> <Rest> ;)
3F0A0	GetNextToken	(hxs-mask $ #start → hxs-mask $ #next $token)

2.4.7 Decompilation

20CB9	EDITDECOMP$	(ob → $)

Calls `setStdEditWid` and the decompiles for editing like <REF>editdecomp$w.

20DC3	DECOMP$	(ob → $)

Calls <REF>setStdWid and decompiles entire object (UserRPL components only). Breaks the string into lines using `DcompWidth` as width.

205CB	DO>STR	($ → $)
		(ob → $)

Internal version of →STR.

3F345	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

05D28	NULL$?	(ob → flag)
0FAF1	DUPNULL$?	(ob → ob flag)

2.5 HEX Strings

2.5.1 Conversion

06185	#>HXS	(# → hxs)

Length will be five.

2.5.2 General Functions

```
05D8E      NULLHXS              HXS 0
```
Puts a null hxs in the stack.
```
0D7E5      EXPAND               ( hxs #nibs → hxs' )
```
Appends #nibs zero nibbles to the hxs.

2.6 Arrays

2.6.1 General Functions

```
171E2      ARSIZE               ( [] → # )
```
Returns number of elements as a bint.
```
171EE      DIMLIMITS            ( [] → {#n #m} )
```
Returns list of array dimensions.
```
0371D      GETATELN             ( # [] → ob T )
                                ( # [] → F )
```
Gets one element from array.
```
17206      MATCON               ( [%] % → [%]' )
                                ( [C%] C% → [C%]' )
```
Replace all elements of [F%] by F%.
```
1722A      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.
```
0F877      MDIMSDROP            ( [2D] → #m #n )
```
MDIMS followed by DROP.
```
0FA22      OVERARSIZE           ( [] ob → [] ob #elts )
```
Does OVER then <REF>ARSIZE.
```
18DBC      PUTEL                ( [%] % # → [%]' )
                                ( [C%] C% # → [C%]' )
```
Puts element at specified position.   Converts to
"short" before. Warning: no copy to tempob first.
```
17212      MATREDIM             ( [F%] {#n #m} → [F%]' )
17214E     MATTRN               ( [F%] → [F%]' )
```
Transposes matrix.

2.6.2 Conversion

```
171FA      MAKEARRY             ( {#n #m} ob → [] )
```
Makes array with all elements initialized to ob.

2.7 Unit Objects

2.7.1 Creating Units

05C3A	EXTN	(ob1..obn #n → u)

Builds a unit object.

2.8 Composites

2.8.1 General Operations

059D3	&COMP	(comp comp' → comp'')

Concatenates two composites.

05AAE	>TCOMP	(comp ob → comp+ob)

Adds ob to tail (end) of composite.

05A7A	>HCOMP	(comp ob → ob+comp)

Adds ob to head (beginning) of composite.

05842	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.

0FA65	?CARCOMP	(comp T → ob)
		(comp F → comp)

If the flag is TRUE, does CARCOMP.

0590C	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.

05E34	LENCOMP	(comp → #n)

Returns length of composite (number of objects).

0FB19	DUPLENCOMP	(comp → comp #n)

Does DUP then <REF>LENCOMP.

05D70	NULLCOMP?	(comp → flag)

If the composite is empty, returns TRUE.

0FB05	DUPNULLCOMP?	(comp → comp flag)

Does DUP then <REF>NULLCOMP?.

05E6F	NTHELCOMP	(comp #i → ob T)
		(comp #i → F)

Returns specified element of composite and TRUE, or just FALSE if it could not be found.

0F476	NTHCOMPDROP	(comp #i → ob)

Does <REF>NTHELCOMP then DROP.

0F5F7	NTHCOMDDUP	(comp #i → ob ob)

Does <REF>NTHCOMPDROP then DUP.

0ED0F	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 ;`
108DF	EQUALPOSCOMP	(comp ob → #pos)
		(comp ob → #0)
		POSCOMP with EQUAL as test.
108DF	EQUALPOSCMP	(comp ob → #pos)
		(comp ob → #0)
108F8	NTHOF	(ob comp → #i)
		(ob comp → #0)
		Does SWAP then <REF>EQUALPOSCOMP.
108C6	#=POSCOMP	(comp # → #i)
		(comp # → #0)
		POSCOMP with #= as test.
05FDA	SUBCOMP	(comp #m #n → comp')
		Returns a sub-composite. Makes all index checks first.
0E8C2	Embedded?	(ob1 ob2 → flag)
		Returns TRUE if ob2 is embedded in, or is the same as, ob1. Otherwise returns FALSE.
0EDF0	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 ;`

0EE09	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.
1090C	EQLookup	(ob comp → nextob T)
		(ob comp → ob F)
		Lookup with EQ as test.

0F15A	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

05C12	{}N	(obn..ob1 #n → { obn..ob1 })
05BFE	::N	(ob1..obn #n → :: ob1..obn ;)
05C26	SYMBN	(ob1..obn #n → sym)
1077C	top&Cr	(meta1 meta2 → symb)
		Does top& then <REF>SYMBN: .
05C3A	EXTN	(ob1..obn #n → u)
		Builds a unit object.
16747	P{}N	(ob1..obn #n → {})
		Build list with possible garbage collection.
1677E	P::N	(ob1..obn #n → seco)
		Build seco with possible garbage collection.

2.8.3 Exploding

05C68	INNERCOMP	(comp → obn..ob1 #n)
0FAC9	DUPINCOMP	(comp → comp obn..ob1 #n)
0FADD	SWAPINCOMP	(comp obj → obj obn..ob1 #n)
0F462	INCOMPDROP	(comp → obn..ob1)
0F51B	INNERDUP	(comp → obn..ob1 #n #n)
0FF7F	INNER#1=	(comp → obn..ob1 flag)

2.8.4 Lists

05DA2	NULL{}	(→ {})
		Pushes a null list to the stack.
102EF	DUPNULL{}?	({} → {} flag)
0FAA1	TWO{}N	(ob1 ob2 → { ob1 ob2 })
0FAB5	THREE{}N	(ob1 ob2 ob3 → { ob1 ob2 ob3 })
0FA8D	#1-{}N	(ob1..obn #n+1 → {})
3F043	PUTLIST	(ob #i {} → {}')
		Replaces object at specified position. Assumes valid #i.

2.8.5 Secondaries

05DB6	NULL::	(→ :: ;)
		Returns null secondary.
10862	Ob>Seco	(ob → :: ob ;)
		Does ONE then <REF>::N.
10849	?Ob>Seco	(ob → :: ob ;)
		If the object is not a secondary, does Ob>Seco.
10876	2Ob>Seco	(ob1 ob2 → :: ob1 ob2 ;)
		Does TWO then <REF>::N.
0FBB9	::NEVAL	(ob1..obn #n → ?)
		Does <REF>::N then <REF>EVAL.

2.9 Meta Objects

2.9.1 Stack Functions

0326E	NDROP	(1..n #n →)
10821	DROPNDROP	(1..n #n ob →)
0F84F	#1+NDROP	(ob 1..n #n →)
		aka: N+1DROP
15068	psh	(meta1 meta2 → meta2 meta1)
		Should be called swap.
3F2BD	roll2ND	(meta1 meta2 meta3 → meta2 meta3 meta1)
		Should be called rot.
3F1A4	unroll2ND	(meta1 meta2 meta3 → meta3 meta1 meta2)
		Should be called unrot.
10191	SWAPUnNDROP	(meta1 meta2 → meta2)
		Should be called swapdrop.
1017D	SWAPUnDROP	(meta1 meta2 → meta2 ob1..obn)
		Swaps two metas and drops the count. Should be called swapDROP.
10795	metaROTDUP	(meta1 meta2 meta3 → meta2 meta3 meta1 meta1)
		Should be called rotdup.

2.9.2 Combining Functions

3F077	top&	(meta1 meta2 → meta1&meta2)
3F170	pshtop&	(meta1 meta2 → meta2&meta1)
107A9	ROTUntop&	(meta1 meta2 meta3 → meta2 meta3&meta1)

107BD	roll2top&	(meta1 meta2 meta3 → meta3 meta1&meta2) aka: rolltwotop&
3F2BA	psh&	(meta1 meta2 meta3 → meta1&meta3 meta2)

2.9.3 Meta and Object Operations

0E47F	SWAP#1+	(meta ob → meta&ob) aka: SWP1+
0D05D	get1	(ob meta → meta ob)
3F06B	psh1top&	(meta ob → ob&meta)
14F70	pull	(meta&ob → meta ob) aka: #1-SWAP
3F070	pullrev	(ob&meta → meta ob)
14F98	psh1&	(meta1 meta2 ob → ob&meta1 meta2)
3F1D2	psh1&rev	(meta1 meta2 ob → ob&meta1 meta2)
3F06F	pullpsh1&	(meta1 meta2&ob → ob&meta1 meta2)
3F06C	pshzer	(meta → #0 meta)

2.10 Symbolics

2.10.1 General Operations

05C26	SYMBN	(ob1..obn #n → sym)
05DAC	NULLSYMB	(→ sym) Puts a null algebraic in the stack.
17C61	symcomp	(ob → ob') If ob is symbolic, does nothing, otherwise ONE SYMBN.
3F2AC	SWAPcompSWAP	(ob ob' → ob'' ob') Does SWAP symcomp SWAP.

2.10.2 Derivatives

2.10.3 Meta Symbolics Functions

1080D	pZpargSWAPUn	(meta → M_rest M_last) <REF>pshzerpsharg then <REF>psh .
107D1	plDRPpZparg	(meta&ob → M_last M_rest) Drops ob then calls <REF>pshzerpsharg .

2.11 Library and Backup Objects

2.11.1 Rompointers

0876D	#>ROMPTR	(#lib #cmd → ROMPTR) Creates rompointer.
095D2	ROMPTR>#	(ROMPTR → #lib #cmd) Splits rompointer.
087B6	ROMPTR@	(ROMPTR → ob T) (ROMPTR → F) Recalls contents of rompointer.
0F4F3	DUPROMPTR@	(ROMPTR → ROMPTR ob T) (ROMPTR → ROMPTR F) Does DUP then ROMPTR@.
0E5DC	?>ROMPTR	(ob → ob') If ROM-WORD? and TYPECOL? then RPL@.
0E5FF	?ROMPTR>	(ob → ob') If <REF>TYPEROMP? and content exists <REF>INHARDROM? then return contents.
161CC	RESOROMP	(→ ob) Recalls contents of next object in the runstream (which must be a rompointer).
0852B	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.
0DB30	ROM-WORD?	(ob → flag)

2.11.2 Libraries

07EE2	TOSRRP	(# →) Attaches library to HOME directory. -- <REF>TEXT:Libraries
07E87	OFFSRRP	(# →) Detaches library from HOME directory. -- <REF>TEXT:Libraries
07F66	ONSRRP?	(# → flag) Returns TRUE if library is attached to HOME directory.
3F1CF	XEQSETLIB	(% →) Internal ATTACH.

2.11.3 Backup Objects

```
08AEC      BAKNAME                ( bak → id T )
                                  Returns backup's name
09DA5      BAK>OB                 ( bak → ob )
                                  Gets backup object.
```

3 General SysRPL Entries

3.1 Stack Operations

03188	DUP	(ob → ob ob)
0F593	DUPDUP	(ob → ob ob ob)
15337	NDUPN	(ob #n → ob..ob #n)
		(ob #0 → #0)
0F892	DUPROT	(1 2 → 2 2 1)
0D0D8	DUPUNROT	(1 2 → 2 1 2)
		aka: SWAPOVER
0F9D2	DUPROLL	(1..n #n → 1 3..n #n 2)
0F9BE	DUPPICK	(n..1 #n → n..1 #n n-1)
031AC	2DUP	(1 2 → 1 2 1 2)
104BB	2DUP5ROLL	(1 2 3 → 2 3 2 3 1)
031D9	NDUP	(1..n #n → 1..n 1..n)
03244	DROP	(1 →)
0E322	DROPDUP	(1 2 → 1 1)
10821	DROPNDROP	(1..n #n ob →)
0E287	DROPSWAP	(1 2 3 → 2 1)
0F8A6	DROPROT	(1 2 3 4 → 2 3 1)
0F90A	DROPOVER	(1 2 3 → 1 2 1)
03258	2DROP	(1 2 →)
0CD2B	3DROP	(1 2 3 →)
		aka: XYZ>
0CD30	4DROP	(1..4 →)
		aka: XYZW>
0CD35	5DROP	(1..5 →)
0CD41	6DROP	(1..6 →)
0CD4D	7DROP	(1..7 →)
0326E	NDROP	(1..n #n →)
0F84F	#1+NDROP	(ob 1..n #n →)
		aka: N+1DROP
0314C	DEPTH	(1..n → 1..n #n)
15194	reversym	(1..n #n → n..1 #n)
03223	SWAP	(1 2 → 2 1)
0E2C2	SWAPDUP	(1 2 → 2 1 1)
1930B	SWAP2DUP	(1 2 → 2 1 2 1)
0CD73	SWAPDROP	(1 2 → 2)
		aka: XY>Y

0E3AB	SWAPDROPDUP	(1 2 → 2 2)
0CD13	SWAPROT	(1 2 3 → 3 2 1)
		aka: UNROTSWAP, XYZ>ZYX
104A7	SWAP4ROLL	(1 2 3 4 → 2 4 3 1)
		aka: XYZW>YWZX
104CF	SWAP3PICK	(1 2 3 → 1 3 2 1)
0DB7B	2SWAP	(1 2 3 4 → 3 4 1 2)
03295	ROT	(1 2 3 → 2 3 1)
0E2F0	ROTDUP	(1 2 3 → 2 3 1 1)
0F557	ROT2DUP	(1 2 3 → 2 3 1 3 1)
0CD01	ROTDROP	(1 2 3 → 2 3)
		aka: XYZ>YZ
0E2A1	ROT2DROP	(1 2 3 → 2)
		aka: DROPSWAPDROP, XYZ>Y
0CCEE	ROTDROPSWAP	(1 2 3 → 3 2)
		aka: XYZ>ZY
0CCC7	ROTSWAP	(1 2 3 → 2 1 3)
		aka: XYZ>YXZ
0CF16	ROTROT2DROP	(1 2 3 → 3)
		aka: UNROT2DROP, XYZ>Z
0F57F	ROTOVER	(1 2 3 → 2 3 1 3)
0CD93	4ROLL	(1 2 3 4 → 2 3 4 1)
		aka: FOURROLL, XYZW>YZWX
0F7A5	4ROLLSWAP	(1 2 3 4 → 2 3 1 4)
0F8E2	4ROLLROT	(1 2 3 4 → 2 4 1 3)
		aka: FOURROLLROT
0F982	4ROLLOVER	(1 2 3 4 → 2 3 4 1 4)
0CDB0	5ROLL	(1 2 3 4 5 → 2 3 4 5 1)
		aka: FIVEROLL
0CDDA	6ROLL	(1..6 → 2..6 1)
		aka: SIXROLL
0CE43	7ROLL	(1..7 → 2..7 1)
		aka: SEVENROLL
0CE14	8ROLL	(1..8 → 2..8 1)
		aka: EIGHTROLL
0CE71	9ROLL	(1..9 → 2..9 1)
03325	ROLL	(1..n #n → 2..n 1)
0F863	ROLLDROP	(1..n #n → 2..n)
0F61F	ROLLSWAP	(1..n #n → 2..n-1 1 n)
0D04B	#1+ROLL	(ob 1..n #n → 1..n ob)
0D070	#2+ROLL	(a b 1..n #n → b 1..n a)
0D036	#+ROLL	(1..n+m #n #m → 2..n+m 1)
0D024	#-ROLL	(1..n-m #n #m → 2..n-m 1)

0CD84	UNROT	(1 2 3 → 3 1 2) aka: 3UNROLL, XYZ>ZXY
0F5CF	UNROTDUP	(1 2 3 → 3 1 2 1)
0E3C6	UNROTDROP	(1 2 3 → 3 1) aka: SWAPDROPSWAP, XYZ>ZX
0F96E	UNROTOVER	(1 2 3 → 3 1 2 1)
0CD84	3UNROLL	(1 2 3 → 3 1 2) aka: UNROT, XYZ>ZXY
0CE8A	4UNROLL	(1 2 3 4 → 4 1 2 3) aka: FOURUNROLL, XYZW>WXYZ
0F5E3	4UNROLLDUP	(1 2 3 4 → 4 1 2 3 3)
0CF28	4UNROLL3DROP	(1 2 3 4 → 4) aka: XYZW>W
0F8F6	4UNROLLROT	(1 2 3 4 → 4 3 2 1)
0CEB0	5UNROLL	(1 2 3 4 5 → 5 1 2 3 4) aka: FIVEUNROLL
0CEE6	6UNROLL	(1..6 → 6 1..5) aka: SIXUNROLL
0F49E	7UNROLL	(1..7 → 7 1..6)
0F9FA	8UNROLL	(1..8 → 8 1..7)
0FA0E	10UNROLL	(1..10 → 10 1..9)
0339E	UNROLL	(1..n #n → n 1..n-1)
0D0AB	#1+UNROLL	(ob 1..n #n → n ob 1..n-1)
0D0BD	#2+UNROLL	(a b 1..n #n → n a b 1..n-1)
0D096	#+UNROLL	(1..n+m #n #m → n+m 1..n+m-1)
0D084	#-UNROLL	(1..n-m #n #m → n-m 1..n+m-1)
032C2	OVER	(1 2 → 1 2 1)
0F5A7	OVERDUP	(1 2 → 1 2 1 1)
0F60B	OVERSWAP	(1 2 → 1 1 2) aka: OVERUNROT
0F60B	OVERUNROT	(1 2 → 1 1 2) aka: OVERSWAP
1050B	OVER5PICK	(1 2 3 4 → 1 2 3 4 3 1)
10835	2OVER	(1 2 3 4 → 1 2 3 4 1 2)
0CFDE	3PICK	(1 2 3 → 1 2 3 1)
0F7B9	3PICKSWAP	(1 2 3 → 1 2 1 3)
0F996	3PICKOVER	(1 2 3 → 1 2 3 1 3)
104E3	3PICK3PICK	(1 2 3 → 1 2 3 1 2)
0CFE3	4PICK	(1 2 3 4 → 1 2 3 4 1)
0F7CD	4PICKSWAP	(1 2 3 4 → 1 2 3 1 4)
104F7	SWAP4PICK	(1 2 3 4 → 1 2 4 3 1)
0F9AA	4PICKOVER	(1 2 3 4 → 1 2 3 4 1 4)

0CFE8	5PICK	(1 2 3 4 5 → 1 2 3 4 5 1)
0CFED	6PICK	(1..6 → 1..6 1)
0CFF2	7PICK	(1..7 → 1..7 1)
0CFF7	8PICK	(1..8 → 1..8 1)
0CFFC	9PICK	(1..9 → 1..9 1)
0D001	10PICK	(1..10 → 1..10 1)
032E2	PICK	(1..n #n → 1..n 1)
0CF8F	#1+PICK	(1..n #n-1 → 1..n 1)
0CFAA	#2+PICK	(1..n #n-2 → 1..n 1)
0CFBE	#3+PICK	(1..n #n-3 → 1..n 1)
0CFCD	#4+PICK	(1..n #n-4 → 1..n 1)
0CF70	#+PICK	(1..n+m #n #m → 1..n+m 1)
0CF5E	#-PICK	(1..n-m #n #m → 1..n-m 1)

3.2 Temporary Environments

3.2.1 Built-in IDs and LAMs

20AFF	NULLID	(→ id)
		Null (empty) identifier.
109FB	NULLLAM	(→ lam)
		Puts NULLLAM in the stack.

3.2.2 Conversion

062CE	$>ID	($ → ID)
0FB7D	DUP$>ID	($ → $ ID)

3.2.3 Temporary Environments Words

07CA9	BIND	(obn..ob1 {lamn..lam1} →)
		Binds n objects to n differently named lams.
07CBD	DOBIND	(obn..ob1 lamn..lam1 #n →)
		Binds n objects to n differently named lams.
0FDC7	1LAMBIND	(ob →)
		Binds one object to a null named lam.
0FDC2	DUP1LAMBIND	(ob → ob)
		Does DUP then <REF>1LAMBIND.
102A9	dvarlsBIND	(ob →)
		Binds ob to LAM 'dvar.

07C70	ABND	(→)
		Abandons topmost temporary environment.
0D863	CACHE	(obn..ob1 #n lam →)
		Binds all objects under the same name. 1LAM has the count.
0DA21	DUMP	(NULLLAM → ob1..obn #n)
		Inverse of CACHE. Always does garbage collection.
080E7	@LAM	(lam → ob T)
		(lam → F)
		Tries recalling object from lam. If successful, returns object and TRUE, otherwise returns just FALSE.
08638	STOLAM	(ob lam →)
		Tries storing object in lam. Generates "Undefined Local Name" error if lam is not found.
07D7E	GETLAM	(#n → ob)
		Gets contents of nth topmost lam.
0D16F	1GETLAM	(→ ob)
0D179	2GETLAM	(→ ob)
0D183	3GETLAM	(→ ob)
0D18D	4GETLAM	(→ ob)
0D197	5GETLAM	(→ ob)
0D1A1	6GETLAM	(→ ob)
0D1AB	7GETLAM	(→ ob)
0D1B5	8GETLAM	(→ ob)
0D1BF	9GETLAM	(→ ob)
0D1C9	10GETLAM	(→ ob)
0D1D3	11GETLAM	(→ ob)
0D1DD	12GETLAM	(→ ob)
0D1E7	13GETLAM	(→ ob)
0D1F1	14GETLAM	(→ ob)
0D1FB	15GETLAM	(→ ob)
0D205	16GETLAM	(→ ob)
0D20F	17GETLAM	(→ ob)
0D219	18GETLAM	(→ ob)
0D223	19GETLAM	(→ ob)
0D22D	20GETLAM	(→ ob)
0D237	21GETLAM	(→ ob)
0D241	22GETLAM	(→ ob)
07DC2	PUTLAM	(ob #n →)
		Stores new contents to nth topmost lam.
0D16A	1PUTLAM	(ob →)

0D174	2PUTLAM	(ob →)
0D17E	3PUTLAM	(ob →)
0D188	4PUTLAM	(ob →)
0D192	5PUTLAM	(ob →)
0D19C	6PUTLAM	(ob →)
0D1A6	7PUTLAM	(ob →)
0D1B0	8PUTLAM	(ob →)
0D1BA	9PUTLAM	(ob →)
0D1C4	10PUTLAM	(ob →)
0D1CE	11PUTLAM	(ob →)
0D1D8	12PUTLAM	(ob →)
0D1E2	13PUTLAM	(ob →)
0D1EC	14PUTLAM	(ob →)
0D1F6	15PUTLAM	(ob →)
0D200	16PUTLAM	(ob →)
0D20A	17PUTLAM	(ob →)
0D214	18PUTLAM	(ob →)
0D21E	19PUTLAM	(ob →)
0D228	20PUTLAM	(ob →)
0D232	21PUTLAM	(ob →)
0D23C	22PUTLAM	(ob →)
0FDAE	1GETABND	(→ 1lamob)
		Does <REF>1GETLAM then <REF>ABND .
0F68D	1ABNDSWAP	(ob → 1lamob ob)
		Does <REF>1GETABND then SWAP.
0F7E1	1GETSWAP	(ob → 1lamob ob)
		Does <REF>1GETLAM then SWAP.
0FBCD	2GETEVAL	(→ ?)
		Does <REF>2GETLAM then <REF>EVAL .
0D3A1	GETLAMPAIR	(#n → #n ob lam F)
		(#n → #n T)
		Gets lam contents and name (10 = 1lam, 20 = 2lam, etc.)
109F6	1NULLLAM{}	(→ {})
		Puts a list with one NULLLAM in the stack.

3.3 Error Handling

3.3.1 General Words

20782	ERRBEEP	(→)
		Beeps.
054F8	ERROR@	(→ #)
		Returns current error number.
05520	ERRORSTO	(# →)
		Stores new error number.
100D3	ERROROUT	(# →)
		Stores new error number and calls ERRJMP.
05545	ERRORCLR	(→)
		Stores zero as new error number.
056CF	ERRJMP	(→)
		Invokes error handling sub-system.
05605	GETEXITMSG	(→ $)
		Gets EXITMSG (user defined error message).
05635	EXITMSGSTO	($ →)
		Stores $ as EXITMSG.
209AE	DO#EXIT	(# →)
		Stores new error number, does <REF>AtUserStack and then <REF>ERRJMP.
209C7	DO$EXIT	($ →)
		Stores string as EXITMSG, #70000 as error number, does <REF>AtUserStack and then <REF>ERRJMP .
056A2	ABORT	(→)
		Does <REF>ERRORCLR and <REF>ERRJMP .
0565C	ERRSET	(→)
		Sets new error trap.
056B6	ERRTRAP	(→)
		Error trap marker. If no error happens, still removes all temporary environments created since ERRSET.
05599	JstGetTHEMESG	(# → $)
		Fetches message from message table. To get a message from a library, use the formula:
		libnum*#100+msgnum.
		--
		<REF>TEXT:Libraries aka: JstGETTHEMSG
05576	GETTHEMESG	(# → $)
		If #70000 then does <REF>GETEXITMSG, else does <REF>JstGetTHEMESG .
		--
		<REF>TEXT:Libraries
15BE9	?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

0576F	SETMEMERR	Error 001h
		Generates "Insufficient Memory" error.
057CF	SETROMPERR	Error 004h
		Generates "Undefined XLIB Name" error.
05763	SETLBERR	Error 006h
		Generates "Power Lost" error.
1321C	SETSTACKERR	Error 201h
		Generates "Too Few Arguments" error.
1320C	SETTYPEERR	Error 202h
		Generates "Bad Argument Type" error.
131FC	SETSIZEERR	Error 203h
		Generates "Bad Argument Value" error.
131EC	SETNONEXTERR	Error 204h
		Generates "Undefined Name" error.

3.4 Conditionals

3.4.1 Boolean Flags

1B6F0	%0<>	(% → flag)
		Can be used to change a user flag into a system flag.
03A81	TRUE	(→ T)
0C476	TrueTrue	(→ T T)
0FDEF	TrueFalse	(→ T F)
		aka: TRUEFALSE
03AC0	FALSE	(→ F)
0FE03	FalseTrue	(→ F T)
		aka: FALSETRUE
3F090	FalseFalse	(→ F F)
0C48A	failed	(→ F T)
0DDD4	DROPTRUE	(ob → T)
0DDDD	DROPFALSE	(ob → F)
0F44E	2DROPFALSE	(ob1 ob2 → F)
0F791	XYZ>ZTRUE	(ob1 ob2 ob3 → ob3 T)
03AF2	NOT	(flag → flag')
		Returns FALSE if the input is TRUE, and vice-versa.
03B46	AND	(flag1 flag2 → flag)
		Returns TRUE if both flags are TRUE.
03B75	OR	(flag1 flag2 → flag)
		Returns TRUE if either flag is TRUE.

03ADA	XOR	(flag1 flag2 → flag)
		Returns `TRUE` if flags are different.
0FEA8	ORNOT	(flag1 flag2 → flag)
		Returns `FALSE` if either flag is `TRUE`.
0F52F	NOTAND	(flag1 flag2 → flag)
		Returns `TRUE` if flag1 is `TRUE` and flag2 is `FALSE`.
0F56B	ROTAND	(flag1 ob flag2 → ob flag)
		Returns `TRUE` if either flag is `TRUE`.

3.4.2 General Tests

03B2E	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.
0FED0	2DUPEQ	(ob1 ob2 → ob1 ob2 flag)
		Does 2DUP then `EQ`.
0FEE4	EQOR	(flag ob1 ob2 → flag')
		Does `EQ` then `OR`.
0F91E	EQOVER	(ob3 ob1 ob2 → ob3 flag ob3)
		Does `EQ` then `OVER`.
03B97	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.
0FEBC	EQUALNOT	(ob1 ob2 → flag)
		Returns `TRUE` if the objects are different.
0FEF8	EQUALOR	(flag ob1 ob2 → flag')
		Does `EQUAL` then `OR`.

3.4.3 True/False Tests

0D604	?SEMI	(T → :: ;)
		(F → :: <ob1> <rest> ;)
0D5F5	NOT?SEMI	(T → :: <ob1> <rest> ;)
		(F → :: ;)
10164	?SEMIDROP	(ob T → :: ob ;)
		(ob F → :: <ob1> <rest> ;)
0F7F5	?SWAP	(ob1 ob2 T → :: ob2 ob1 <ob1> <rest> ;)
		(ob1 ob2 F → :: ob1 ob2 <ob1> <rest> ;)
0F679	?SKIPSWAP	(ob1 ob2 T → :: ob1 ob2 <ob1> <rest> ;)
		(ob1 ob2 F → :: ob2 ob1 <ob1> <rest> ;)
0F836	?SWAPDROP	(ob1 ob2 T → :: ob1 <ob1> <rest> ;)
		(ob1 ob2 F → :: ob2 <ob1> <rest> ;)

0F81D	NOT?SWAPDROP	(ob1 ob2 T → :: ob2 <ob1> <rest> ;)
		(ob1 ob2 F → :: ob1 <ob1> <rest> ;)
078D6	RPIT	(T ob → :: ob <ob1> <rest> ;)
		(F ob → :: <ob1> <rest> ;)
		ob is actually executed, and not pushed in the stack.
0789C	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.
0D657	COLARPITE	(T ob1 ob2 → :: ob1 ;)
		(F ob1 ob2 → :: ob2 ;)
		ob1 or ob2 is actually executed, and not pushed in the stack.
0D6B2	2'RCOLARPITE	Return to composite and ITE there.
0D585	IT	(T → :: <ob1> <rest> ;)
		(F → :: <ob2> <rest> ;)
07903	?SKIP	(T → :: <ob2> <rest> ;)
		(F → :: <ob1> <rest> ;)
		aka: NOT_IT
0D6A1	ITE	(T → :: <ob1> <ob3> <rest> ;)
		(F → :: <ob2> <rest> ;)
0D621	ITE_DROP	(ob T → :: <ob2> <rest> ;)
		(ob F → :: ob <ob1> <rest> ;)
106DC	ANDITE	(f1 f2 → :: <ob1> <ob3> <rest> ;)
		(f1 f2 → :: <ob2> <rest> ;)
0D55C	case	(T → :: <ob1> ;)
		(F → :: <ob2> <rest> ;)
0D576	NOTcase	(T → :: <ob2> <rest> ;)
		(F → :: <ob1> ;)
10565	ANDcase	(f1 f2 → :: <ob1> ;)
		(f1 f2 → :: <ob2> <rest> ;)
1065A	ANDNOTcase	(f1 f2 → :: <ob1> ;)
		(f1 f2 → :: <ob2> <rest> ;)
0E537	ORcase	(f1 f2 → :: <ob1> ;)
		(f1 f2 → :: <ob2> <rest> ;)
0D4C0	casedrop	(ob T → :: <ob1> ;)
		(ob F → :: ob <ob2> <rest> ;)
0D4B1	NOTcasedrop	(ob T → :: ob <ob2> <rest> ;)
		(ob F → :: <ob1> ;)
0D4E8	case2drop	(ob1 ob2 T → :: <ob1> ;)
		(ob1 ob2 F → :: ob1 ob2 <ob2> <rest> ;)
0D4D9	NOTcase2drop	(ob1 ob2 T → :: ob1 ob2 <ob2> <rest> ;)
		(ob1 ob2 F → :: <ob1> ;)
0D514	caseDROP	(ob T → :: ;)
		(ob F → :: ob <ob1> <rest> ;)

0D529	NOTcaseDROP	(ob T → :: ob <ob1> <rest> ;)
		(ob F → :: ;)
10119	casedrptru	(ob T → T)
		(ob F → :: ob <ob1> <rest> ;)
		Note: should be called `caseDRPTRU`.
0FE62	casedrpfls	(ob T → F)
		(ob F → :: ob <ob1> <rest> ;)
		Note: should be called `caseDRPFLS`.
10380	NOTcsdrpfls	(ob T → :: ob <ob1> <rest> ;)
		(ob F → F)
		Note: should be called `NOTcaseDRPFLS`.
0D539	case2DROP	(ob1 ob2 T → :: ;)
		(ob1 ob2 F → :: ob1 ob2 <ob1> <rest> ;)
0D54D	NOTcase2DROP	(ob1 ob2 T → :: ob1 ob2 <ob1> <rest> ;)
		(ob1 ob2 F → :: ;)
0FE7B	case2drpfls	(ob1 ob2 T → F)
		(ob1 ob2 F → :: ob1 ob2 <ob1> <rest> ;)
		Note: should be called `case2DRPFLS`.
0FDDB	caseTRUE	(T → T)
		(F → :: <ob1> <rest> ;)
10132	NOTcaseTRUE	(T → :: <ob1> <rest> ;)
		(F → T)
0FE94	caseFALSE	(T → F)
		(F → :: <ob1> <rest> ;)
1014B	NOTcaseFALSE	(T → :: <ob1> <rest> ;)
		(F → F)
0E501	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

0FC64	#=?SKIP	(#m #n → :: <ob2> <rest> ;)
		(#m #n → :: <ob1> <rest> ;)
0FC91	#>?SKIP	(#m #n → :: <ob1> <rest> ;)
		(#m #n → :: <ob2> <rest> ;)
0F507	#=ITE	(#m #n → :: <ob1> <ob3> <rest> ;)
		(#m #n → :: <ob2> <rest> ;)
10718	#<ITE	(#m #n → :: <ob1> <ob3> <rest> ;)
		(#m #n → :: <ob2> <rest> ;)
1072C	#>ITE	(#m #n → :: <ob2> <rest> ;)
		(#m #n → :: <ob1> <ob3> <rest> ;)
0D435	#=case	(#m #n → :: <ob1> ;)
		(#m #n → :: <ob2> <rest> ;)
0D445	OVER#=case	(#m #n → :: #m <ob1> ;)
		(#m #n → :: #m <ob2> <rest> ;)

0D49C	#=casedrop	(#m #n → :: <ob1> ;)
		(#m #n → :: #m <ob2> <rest> ;)
		Note: should be called OVER#=casedrop.
0FE3F	#=casedrpfls	(#m #n → F)
		(#m #n → :: #m <ob1> <rest> ;)
		Note: should be called OVER#=caseDRPFLS.
105B5	#<>case	(#m #n → :: <ob2> <rest> ;)
		(#m #n → :: <ob1> ;)
1058D	#<case	(#m #n → :: <ob1> ;)
		(#m #n → :: <ob2> <rest> ;)
105E2	#>case	(#m #n → :: <ob2> <rest> ;)
		(#m #n → :: <ob1> ;)
0D5E1	#0=?SEMI	(#0 → :: ;)
		(# → :: <ob1> <rest> ;)
0FC22	#0=?SKIP	(#0 → :: <ob2> <rest> ;)
		(# → :: <ob1> <rest> ;)
10704	#0=ITE	(#0 → :: <ob1> <ob3> <rest> ;)
		(# → :: <ob2> <rest>)
106C3	DUP#0=IT	(#0 → :: #0 <ob1> <rest> ;)
		(# → :: # <ob2> <rest> ;)
10740	DUP#0=ITE	(#0 → :: #0 <ob1> <ob3> <rest> ;)
		(# → :: # <ob2> <rest> ;)
0D45F	#0=case	(#0 → :: <ob1> ;)
		(# → :: <ob2> <rest> ;)
0D45A	DUP#0=case	(#0 → :: #0 <ob1> ;)
		(# → :: # <ob2> <rest> ;)
0D471	DUP#0=csedrp	(#0 → :: <ob1> ;)
		(# → :: # <ob2> <rest> ;)
10538	DUP#0=csDROP	(#0 → :: ;)
		(# → :: # <ob1> <rest> ;)
105A1	#1=case	(#1 → :: <ob1> ;)
		(# → :: <ob2> <rest> ;)
0FC4B	#1=?SKIP	(#1 → :: <ob2> <rest> ;)
		(# → :: <ob1> <rest> ;)
105C9	#>2case	(#0/#1/#2 → :: <ob2> <rest> ;)
		(# → :: <ob1> ;)
24008	?CaseKeyDef	(# #' → :: ' ob1 T ;)
		(# #' → :: <ob2> <rest> ;)
		Compares two bints. If equal, quotes the next object from the runsream and returns it along with TRUE.
2403A	?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

105F6	j%0=case	(%0 → :: <ob1> ;) (ob → :: <ob2> <rest> ;)

3.4.6 General Object Tests

106AA	EQIT	(ob1 ob1 → :: <ob1> <rest> ;) (ob1 ob2 → :: <ob2> <rest> ;)
106F0	EQITE	(ob1 ob1 → :: <ob1> <ob3> <rest> ;) (ob1 ob2 → :: <ob2> <rest> ;)
10551	jEQcase	(ob1 ob1 → :: <ob1> ;) (ob1 ob2 → :: <ob2> <rest> ;)
0D4FC	EQcase	(ob1 ob1 → :: ob1 <ob1> ;) (ob1 ob2 → :: ob1 <ob2> <rest> ;) Note: Should be called OVEREQcase.
0D483	EQcasedrop	(ob1 ob1 → :: <ob1> ;) (ob1 ob2 → :: ob1 <ob2> <rest> ;) Note: should be called OVEREQcasedrop.
10579	EQUALcase	(ob1 ob1 → :: <ob1> ;) (ob1 ob2 → :: <ob2> <rest> ;)
1066E	EQUALNOTcase	(ob1 ob1 → :: <ob2> <rest> ;) (ob1 ob2 → :: <ob1> ;)
1051F	EQUALcasedrp	(ob ob1 ob2 → :: <ob1> ;) (ob ob1 ob2 → :: ob <ob2> <rest> ;)
10682	dIDNTNcase	(id → :: id <ob2> <rest> ;) (ob → :: ob <ob1> ;)
1060A	REALcase	(% → :: <ob1> ;) (ob → :: <ob2> <rest> ;)
10696	dREALNcase	(% → :: % <ob2> <rest> ;) (ob → :: ob <ob1> ;)
1061E	dARRYcase	([] → :: [] <ob1> ;) (ob → :: ob <ob2> <rest> ;)
10632	dLISTcase	({} → :: {} ob1 ;) (ob → :: ob <ob2> <rest> ;)

3.4.7 Miscellaneous

10754	UserITE	(#set → :: <ob1> <ob3> <rest> ;) (#clr → :: <ob2> <rest> ;)
10768	SysITE	(#set → :: <ob1> <ob3> <rest> ;) (#clr → :: <ob2> <rest> ;)

10466	caseDoBadKey	(T → :: DoBadKey ;)
		(F → :: <ob1> <rest> ;)
		aka: caseDEADKEY
1044D	caseDrpBadKy	(ob T → :: DoBadKey ;)
		(ob F → :: ob <ob1> <rest> ;)
0FA51	caseERRJMP	(T → :: ERRJMP ;)
		(F → :: <ob> <rest> ;)
10399	caseSIZEERR	(T → :: SIZEERR ;)
		(F → :: <ob> <rest> ;)
103AD	NcaseSIZEERR	(T → :: <ob> <rest> ;)
		(F → :: SIZEERR ;)
103C1	NcaseTYPEERR	(T → :: <ob1> <rest> ;)
		(F → :: TYPEERR ;)
10646	EditExstCase	(→ :: <ob1> <rest> ;)
		(→ :: <rest> ;)
		Tests if there is an edit line active.

3.5 Runstream Control

0765C	NOP	(→)
		Does nothing.
3AD12	xNEGNEG	(→)
		Does nothing, decompiles to
		:: CK1&Dispatch BINT0 NOP ;
		There like NOP, but requires an argument.
076B9	'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.
07734	'REVAL	(→ ?)
		Does <REF>'R then <REF>EVAL.
1025E	'R'R	(→ ob1 ob2)
		Does <REF>'R twice.
0D752	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.
1027C	'RRDROP	(→ ob)
		Does <REF>'R , then <REF>RDROP.
07778	>R	(:: →)
		Pushes :: to top of return stack (skips prolog, i.e., the composite will be executed automatically).

077F8	R>	(→ ::)
		Creates and pops a secondary from top return stack body to stack.
077EB	R@	(→ ::)
		Like <REF>R>, but the return stack is not popped.
0775C	EVAL	(ob →)
		Evaluates object.
1340B	COMPEVAL	(comp →)
		EVAL just pushes a list back, this one executes it.
20898	RDUP	(→)
		Duplicates top return stack level.
07790	RDROP	(→)
		Pops the return stack.
0CF3A	2RDROP	(→)
		Pops two return stack levels.
0CF4C	3RDROP	(→)
		Pops three return stack levels.
0FBE1	DROPRDROP	(ob →)
		Does DROP then <REF>RDROP .
0E4D3	RDROPCOLA	(→)
		Does <REF>RDROP then <REF>COLA .
0CC9D	RSWAP	(→)
		Swap in the return stack.
208ED	RROLL	(#n →)
		Rolls nth return stack level to top of return stack.
100E7	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

07665	'	(→ nob (nextob))
		Pushes next object in the stream to the stack (skipping it).
3EBFB	xSILENT'	(→ nextob)
		Put the next ob in the runstream on the stack. Quoter used in UserRPL.
101A5	DUP'	(ob → ob nob)
		Does DUP then '.
101CD	DROP'	(ob → nob)
		Does DROP then '.
101B9	SWAP'	(ob1 ob2 → ob2 ob1 nob)
		Does SWAP then '.

101E1	OVER'	(ob1 ob2 → ob1 ob2 ob1 nob)
		Does OVER then '.
101F5	STO'	(ob id/lam → nob)
		Does STO then '.
10209	TRUE'	(→ T nob)
		Pushes TRUE and the next object to the stack.
10236	FALSE'	(→ F nob)
		Pushes FALSE and the next object to the stack.
1021D	ONEFALSE'	(→ #1 F nob)
		Pushes ONE, FALSE and the next object to the stack.
1024A	#1+'	(# → #+1 nob)
		Does #1+ then '.
0FBA5	'NOP	(→ NOP)
		Pushes NOP to the stack.
0FA3D	'ERRJMP	(→ ERRJMP)
		Pushes ERRJMP to the stack.
26E33	'DROPFALSE	(→ DROPFALSE)
		Pushes DROPFALSE to the stack.
23F4A	'DoBadKey	(→ DoBadKey)
		Pushes DoBadKey to the stack.
23F5E	'DoBadKeyT	(→ DoBadKey T)
		Pushes <REF>DoBadKey and TRUE to the stack.
103E9	'xDER	(→ xDER)
		Pushes xDER (User word ∂) to the stack.
14D89	'Rapndit	(meta ob1...ob4 → meta&ob ob1...ob4)
		Takes ob from runstream and appends it to the meta
		starting in level 5.
102D6	'xDEREQ	(ob → flag)
		Is ob eq to user command xDER?

3.5.2 Skipping Objects

077AA	COLA	Evals next obj and
		drops rest of this stream.
10295	ONECOLA	Does ONE, then COLA.
0FBFA	SWAPCOLA	Does SWAP, then COLA.
0FC0E	XYZ>ZCOLA	Does UNROT2DROP, then COLA.
0D636	COLA_EVAL	Returns and evals first obj
		in previous stream.
0E4E8	COLACOLA	Drops rest of current stream
		does COLA in the above one.
07926	SKIP	Skips 1 obj in the runstream.
0FCAA	COLASKIP	Drops rest of current stream
		and skips one obj in above stream.

3.6 Loops

3.6.1 Indefinite Loops

0797B	BEGIN	(→) Pushes interpreter pointer into the return stack.
07984	AGAIN	(→) Sets the interpreter pointer to the topmost value in the return stack, without popping it.
079BE	REPEAT	(→) Sets the interpreter pointer to the topmost value in the return stack, without popping it.
079A1	UNTIL	(flag →) If FALSE then <REF>AGAIN, otherwise <REF>RDROP .
0FCBE	NOT_UNTIL	(flag →) NOT then <REF>UNTIL .
0F4B2	#0=UNTIL	(# → #) Actually, should be called DUP#0=UNTIL.
079C7	WHILE	(flag →) If TRUE does nothing, otherwise <REF>RDROP then <REF>2SKIP .
0FCD7	NOT_WHILE	(flag →) NOT then <REF>WHILE .
0FCF0	DUP#0<>WHILE	(# →) Try to guess what it does.

3.6.2 Definite Loops

07BD0	DO	(#stop #start →)
07B9C	ZERO_DO	(#stop →)
0FD77	DUP#0_DO	(#stop → #stop)
07BA7	ONE_DO	(#stop →)
07BB4	#1+_ONE_DO	(#stop →)
0FD90	toLEN_DO	({} → {}) From ONE to #elements.
07B0D	LOOP	(→)
07B7E	+LOOP	(# →) Increments index by specified number.
0FD5E	DROPLOOP	(ob →)
0FD45	SWAPLOOP	(ob1 ob2 → ob2 ob1)
079FA	INDEX@	(→ #) Recalls topmost loop counter value.

0FD09	DUPINDEX@	(ob → ob #)
0FD1D	SWAPINDEX@	(ob1 ob2 → ob2 ob1 #)
0FD31	OVERINDEX@	(ob1 ob2 → ob1 ob2 ob1 #)
1006F	INDEX@#-	(# → #')
07A49	INDEXSTO	(# →)
		Stores new topmost loop counter value.
07A22	ISTOP@	(→ #)
		Recalls topmost loop stop value.
07A6E	ISTOPSTO	(# →)
		Stores new topmost loop stop value.
07A31	JINDEX@	(→ #)
		Recalls second topmost loop counter value.
07A86	JINDEXSTO	(# →)
		Stores new second topmost loop counter value.
07A3D	JSTOP@	(→ #)
		Recalls second topmost loop stop value.
07A9B	JSTOPSTO	(# →)
		Stores new second topmost loop stop value.
1088A	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

081D3	@	(id/lam → ob T)
		(id/lam → F)
		Basic recalling function.
0F4DF	DUP@	(id/lam → id/lam ob T)
		(id/lam → id/lam F)
		Does DUP then <REF>@.
08644	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.
08ED9	REPLACE	(newob oldob → newob)
		Replaces oldob (in memory) with newob.
0952D	PURGE	(id →)
		Purges variable. Does no type check first.

08F9C	CREATE	(ob id →)

Creates a variable in the current directory. Errors if id is or contains current directory. Assumes id is not a pco.

102BD	'LAMLNAMESTO	(ob →)

`STO` to LAM `LAMLNAME`.

3.7.2 Directories

07FBD	MAKERRP	(#libnum → rrp)

Creates an empty directory.

22F3B	CREATEDIR	(id →)

Creates an empty directory. Calls <REF>?PURGE_HERE first to delete the original.

08C35	LASTRAM-WORD	(rrp → ob T)
		(rrp → F)

Recalls first object in directory.

08C85	PREVRAM-WORD	(ob → ob' T)
		(ob → F)

Recalls next object in directory.

08BF2	RAM-WORDNAME	(ob → id)

Recalls name of object in current directory.

096DA	SYSRRP?	(rrp → flag)

Is `rrp` HOME?

09660	CONTEXT@	(→ rrp)

Recalls current directory.

0960E	CONTEXT!	(rrp →)

Sets new current directory.

096DA	SYSRRP?	(rrp → flag)

Is `rrp` HOME?

09698	HOMEDIR	(→)

Sets HOME as current directory. aka: `SYSCONTEXT`

3.7.3 Temporary Memory

06E2F	TOTEMPOB	(ob → ob')

Copies object to `TEMPOB` and returns pointer to the new copy.

0F543	TOTEMPSWAP	(ob1 ob2 → ob2' ob1)

Does `TOTEMPOB` then `SWAP`.

16451	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.

107F9	SWAPCKREF	(ob1 ob2 → ob2 ob1')
		Does SWAP then <REF>CKREF.
0731C	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

25186	SLOW	(→)
		15 millisecond delay.
251A1	VERYSLOW	(→)
		300 millisecond delay.
251B1	VERYVERYSLOW	(→)
		3 second delay.
3A7DB	dowait	(%secs →)
		Waits specified number of seconds.
1B594	%>HMS	(% → %hms)
		Converts from decimal to H.MMSS format.
1B5AD	%HMS>	(%hms → %)
		Converts from H.MMSS format to decimal.
1B5C1	%HMS+	(%hms1 %hms2 → %hms)
		Adds time in hms format.
1B5E9	%HMS-	(%hms1 %hms2 → %hms)
		Subtracts time in hms format.
3F7A7	CLKTICKS	(→ hxs)
		Returns tick count. aka: SysTime

3.9 System Functions

3.9.1 User and System Flags

193A7	SetSysFlag	(# →)
		Sets the system flag with number #. <REF>TEXT:Flags
193D7	ClrSysFlag	(# →)
		Clears the system flag with number #. <REF>TEXT:Flags
193FA	TestSysFlag	(# → flag)
		Returns TRUE if system flag is set. <REF>TEXT:Flags

1939B	SetUserFlag	(# →)
		Set the user flag with number #. <REF>TEXT:Flags
193CB	ClrUserFlag	(# →)
		Clear the user flag with number #. <REF>TEXT:Flags
193EE	TestUserFlag	(# → flag)
		Returns TRUE if user flag is set. <REF>TEXT:Flags

3.9.2 General Functions

2074F	setbeep	(#ms #Hz →)
		Also beeps.
04A53	TurnOff	(→)
		Internal OFF.
06742	GARBAGE	(→)
		Forces garbage collection.
06761	MEM	(→ #)
		Returns amount of free memory in nibbles. Does not do garbage collection. (The user word does.)
060BB	OSIZE	(ob → #)
		Returns object size in nibbles. Forces garbage collection.
060FD	OCRC	(ob → #nib hxs)
		Returns size in nibbles and checksum as hxs.
140B7	INHARDROM?	(ob → ob flag)
		Is object address < #80000h?
0626C	CHANGETYPE	(ob #prolog → ob')
		Changes prolog of object, does TOTEMPOB.

3.10 Kermit

3F293	DOFINISH	(→)
		Internal FINISH.
3F28E	CLOSEUART	(→)
		Internal CLOSEIO.
3F040	OpenIO	
3F2B3	docr	
170BB	CloseUart	
3F2A4	OpenUart?Clr	
3F2A5	PRINT	
3F2AD	SetEcma94	
3F291	AllowPrlcdCl	

```
3F2AF      UARTBUFLEN
3F2A6      PUTSERIAL
3F29E      GETSERIAL
```

4 Input and Output

4.1 Checking for Arguments

4.1.1 Number and Type of Arguments

12F1E	CK0	(→)

Saves current command to LASTCKCMD. Marks stack below level 1 to STACKMARK.

13573	CK&DISPATCH0	(→)

Dispatches on stack argument. Does not convert ZINTs to REAL.

--

<REF>CK&DISPATCH1 <REF>CK&DISPATCH2 <REF>TEXT:Dispatch_Types

1352A	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

13582	CK&DISPATCH2	(→)

Equivalent to <REF>CK&DISPATCH1.

--

<REF>CK&DISPATCH0
<REF>TEXT:Dispatch_Types

1341F	CK1&Dispatch	(→)

Combines <REF>CK1 with <REF>CK&DISPATCH1.

--

<REF>TEXT:Dispatch_Types

13430	CK2&Dispatch	(→)

Combines <REF>CK2 with <REF>CK&DISPATCH1.

--

<REF>TEXT:Dispatch_Types

13441	CK3&Dispatch	(→)

Combines <REF>CK3 with <REF>CK&DISPATCH1.

--

<REF>TEXT:Dispatch_Types

13452	CK4&Dispatch	(→)
		Combines <REF>CK4 with <REF>CK&DISPATCH1.
		--
		<REF>TEXT:Dispatch_Types
13463	CK5&Dispatch	(→)
		Combines <REF>CK5 with <REF>CK&DISPATCH1.
		--
		<REF>TEXT:Dispatch_Types
232A7	0LASTOWDOB!	(→)
		Clears command save by last CK<n> command. <REF>CK0 aka: 0LASTOWDOB!, 0LastRomWrd!

4.1.2 Type Checking

0E767	CKREAL	(% → %)
		(Z → %)
		Checks for real. If a ZINT, convert to real. Else SETTYPEERR.
03C64	TYPE	(ob → #prolog)
		Returns address of prolog of object.
0DC71	TYPEREAL?	(ob → flag)
0DC6C	DUPTYPEREAL?	(ob → ob flag)
		aka: DTYPEREAL?
0DC80	TYPECMP?	(ob → flag)
0DC7B	DUPTYPECMP?	(ob → ob flag)
0DC62	TYPECSTR?	(ob → flag)
0DC5D	DUPTYPECSTR?	(ob → ob flag)
		aka: DTYPECSTR?
0DC8A	DUPTYPEARRY?	(ob → ob flag)
		aka: DTYPEARRY?
0DC8F	TYPEARRY?	(ob → flag ???)
0DDE6	TYPERARRY?	(ob → flag)
0DE01	TYPECARRY?	(ob → flag)
0DCE9	TYPELIST?	(ob → flag)
0DCE4	DUPTYPELIST?	(ob → ob flag)
		aka: DTYPELIST?
0DBAE	TYPEIDNT?	(ob → flag)
0DBA9	DUPTYPEIDNT?	(ob → ob flag)
0DC35	TYPELAM?	(ob → flag)
0DC30	DUPTYPELAM?	(ob → ob flag)
0DCBC	TYPESYMB?	(ob → flag)

0DCB7	DUPTYPESYMB?	(ob → ob flag)
0DC53	TYPEHSTR?	(ob → flag)
0DC4E	DUPTYPEHSTR?	(ob → ob flag)
0DCDA	TYPEGROB?	(ob → flag)
0DCD5	DUPTYPEGROB?	(ob → ob flag)
0DCF3	DUPTYPETAG?	(ob → ob flag)
0DD07	TYPEEXT?	(ob → flag) Is ob a unit object?
0DD02	DUPTYPEEXT?	(ob → ob flag) Is ob a unit object?
0DC9E	TYPEROMP?	(ob → flag)
0DC99	DUPTYPEROMP?	(ob → ob flag)
0DC44	TYPEBINT?	(ob → flag)
0DC3F	DUPTYPEBINT?	(ob → ob flag)
0DCAD	TYPERRP?	(ob → flag)
0DCA8	DUPTYPERRP?	(ob → ob flag)
0DB9F	TYPECHAR?	(ob → flag)
0DB9A	DUPTYPECHAR?	(ob → ob flag)
0DCCB	TYPECOL?	(ob → flag) Is on a secondary?
0DCC6	DUPTYPECOL?	(ob → ob flag) Is ob a secondary? aka: DTYPECOL?

4.2 Keyboard Control

4.2.1 Converting Keycodes

25F78	Ck&DecKeyLoc	(%rc.p → #kc #p) Converts from user key representation format to system. Does handle shift-hold keys.
26068	CodePl>%rc.p	(#kc #p → %rc.p) Converts from system key representation format to user. Does handle shift-hold keys.
23F90	H/W>KeyCode	(# → #') Converts the keycode offset for shift keys to the keycode of the shift key, i.e. 80h->32d, 40h->37d, C0h->42d
23FC7	ModifierKey?	(#kc #pl → flag) Is the key any of the three modifiers right-shift, left-shift, or alpha?

4.2.2 Waiting for Keys

00C3A	FLUSHKEYS	(→) Flushes the key buffer. aka: FLUSH
04F92	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
04F9E	GETTOUCH	(→ #kc T) (→ F) Pops next key from key buffer (if there is one). Does handle shift-hold keys. -- <REF>TEXT:Keycodes
2661F	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
264C6	GETKEY*	(→ #kc T) (→ F F) (→ {Alrmlist} T F) Get a single keypress from the keybuffer, waits if necessary. The key is returned along with TRUE. If an exception happens (error or alarm), the exceptions is handled and the entry returns FALSE. Does handle shift-hold keys. -- <REF>TEXT:Keycodes
05051	REPKEY?	(#kc → flag) Returns TRUE if the key is being pressed. -- <REF>TEXT:Keycodes
1747B	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

1747B	REPEATERCH	(→)
		Same as `REPEATER`, but slower, so more appropriate for scrolling and cursor motions.
		--
		<REF>TEXT:Keycodes
26943	KEYINBUFFER?	(→ `flag`)
		Returns `TRUE` if there is at least a key in the key buffer.
262DB	WaitForKey	(→ `#kc #flag`)
		Returns next full key press. Does *not* handle shift-hold keys.
		--
		<REF>TEXT:Keycodes

4.2.3 The ATTN Flag

26793	ATTN?	(→ `flag`)
		Returns `TRUE` if (CANCEL) has been pressed.
269B1	?ATTNQUIT	(→)
		If (CANCEL) has been pressed, ABORTs program. aka: `?ATTN_QUIT`
05821	ATTNFLGCLR	(→)
		Clears (CANCEL) key counter. Does not affect the key buffer.

4.2.4 Bad Keys

| 23E15 | DoBadKey | (→) |
| | | Beeps. |

4.2.5 User Keys

| 23C2B | 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

234B0   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

24448   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).
29B79   NullMenuKey   (→)
                                     A placeholder for an empty menu key when defining
                                     menu lists.
2523D   InitMenu   (menu →)
                                     menu is {} or :: settings {} ; Settings override the
                                     default settings installed by InitMenu.

4.3.3 Menu Display

10E0B   ?DispMenu   (→)
                                     Redisplay the menu now if no key is waiting in the
                                     buffer. Even better is this:
                                     :: DA3OK?NOTIT ?DispMenu ;
10E3D   DispMenu.1   (→)
                                     Displays the menu immediately.
10E29   DispMenu   (→)
                                     :: DispMenu.1 SetDAsValid ;

4.3.4 General Entries

26E61   SetSomeRow   (#n →)
                                     with $\mathrm{Mod}(n,\mathrm{FFFFFh}) = 0$.

4.4 InputLine and Inputforms

4.4.1 Inputform

```
15A41        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)

```
35577        ParOuterLoop             ( Disp Keys NonAppKeys? DoStdKeys? menu #row
                                      suspendOK? ExitCond AppErr → )
355AE        POLSaveUI                ( Disp Keys NonAppKeys? DoStdKeys? menu #row
                                      suspendOK? ExitCond AppErr → )
                                      Saves current UI to LAMSavedUI.
3565D        POLSetUI                 <see>ParOuterLoop
                                      Sets new UI, same arguments as to ParOuterLoop.
35743        POLKeyUI                 ( → )
                                      Displays, reads and evaluates keys according to set
                                      UI.
357E3        POLRestoreUI             ( → )
                                      Restores saved UI from LAMSavedUI.
357CA        POLResUI&Err             ( → )
                                      Restores saved UI and executes ERRJMP.
```

4.7 The Display

4.7.1 Display Organization

```
1F4A8        TOADISP                  ( → )
                                      Sets the text display as the active.
1F490        TOGDISP                  ( → )
                                      Sets the graphic display as the active.
1E96F        ABUFF                    ( → textgrob )
                                      Returns the text grob to the stack.
1E97F        GBUFF                    ( → graphgrob )
                                      Returns the graphic grob to the stack. The HP49
                                      extable address for ExitAction! is the same, but this
                                      must be a bug.
```

1E94F	HARDBUFF	(→ dispgrob)

Returns the current grob to the stack.

1E95F	HARDBUFF2	(→ menugrob)

Returns the menu grob to the stack.

1EE86	HARDHEIGHT	(→ #height)

Returns the height of HARDBUFF.

4.7.2 Preparing the Display

1F3C6	RECLAIMDISP	(→)

Activates the text grob, clears it and sets the default size.

3F392	MENUOFF?	(→ flag)

Returns TRUE if the menu grob is off.

3F3D5	TURNMENUOFF	(→)

Turns off menu display, enlarges ABUFF to fill screen.

3F3D6	TURNMENUON	(→)

Turns menu grob on.

1F0EB	HEIGHTENGROB	(grob #rows →)

Heightens graph or text grob.

1F37B	KILLGDISP	(→)

Clears graph display by setting it to NULLGROB. See DOERASE.

3F336	DOERASE	(→)

Erases the graphics display grob without changing its size.

4.7.3 Controlling Display Refresh

3624D	ClrDA1OK	(→)
36266	ClrDA2aOK	(→)
3627F	ClrDA2bOK	(→)
36298	ClrDA2OK	(→)
362AC	ClrDA3OK	(→)
35F55	DA1OK?	(→ flag)
35FA5	DA2aOK?	(→ flag)
35FF5	DA2bOK?	(→ flag)
36045	DA2OK?	(→ flag)
3605E	DA3OK?	(→ flag)
360AE	DAsOK?	(→ flag)
36162	DA2aLess1OK?	(→ flag)

3611C	DA3OK?NOTIT	(→)
		Does `DA3OK?`, `NOT` then `IT`.
361D5	SetDA1Temp	(→)
361EE	SetDA2aTemp	(→)
3621B	SetDA3Temp	(→)
3639C	SetDA12Temp	(→)
363B0	SetDAsTemp	(→)
3617B	SetDA1Valid	(→)
36194	SetDA2aValid	(→)
361A8	SetDA2bValid	(→)
362DE	SetDA2Valid	(→)
361C1	SetDA3Valid	(→)
36482	SetDA3ValidF	(→)
365FC	SetDA1Bad	(→)
3660A	ClrDA1Bad	(→)
365EE	DA1Bad?	(→ flag)
36626	SetDA2aBad	(→)
36634	ClrDA2aBad	(→)
36618	DA2aBad?	(→ flag)
36650	SetDA2bBad	(→)
3665E	ClrDA2bBad	(→)
3667A	SetDA3Bad	(→)
36554	SetDA1NoCh	(→)
3657E	SetDA2aNoCh	(→)
365A8	SetDA2bNoCh	(→)
365B6	ClrDA2bNoCh	(→)
3659A	DA2bNoCh?	(→ flag)
3630B	SetDA2NoCh	(→)
3631F	SetDA12NoCh	(→)
365D2	SetDA3NoCh	(→)
36347	SetDA13NoCh	(→)
36333	SetDA23NoCh	(→)
3635B	SetDA12a3NCh	(→)
		aka: `SetDA12a3NoCh`
3636F	SetDAsNoCh	(→)
3622F	SetDA2aEcho	(→)
366A4	SetDA1IsStat	(→)
366FB	SetNoRollDA2	(→)

4.7.4 Clearing the Display

1E9F9	BLANKIT	(#startrow #rows →) Clears #rows from HARDBUFF, starting at #startrow.
01F83	CLCD10	(→) Clears status and stack areas.
3F334	DOCLLCD	(→) Like user word <REF>CLLCD.

4.7.5 Annunciator and Modes Control

25023	LockAlpha	(→) Sets alpha mode, annunciators, etc.
25037	UnLockAlpha	(→) Clears alpha mode, annunciators, etc.

4.7.6 Window Coordinates

1FC49	WINDOWCORNER	(→ #y #x) Gets coordinates of corner of window. Note the order of #x and #y.
1FB06	WINDOWXY	(#y #x →) Sets corner coordinates. The name really should be WINDOWYX

4.7.7 Scrolling the Display

1F523	WINDOWUP	(→) Moves display one pixel up.
1F57B	WINDOWDOWN	(→) Moves display one pixel down.
1F83F	WINDOWLEFT	(→) Moves display one pixel left.
1FA0C	WINDOWRIGHT	(→) Moves display one pixel right.
3F3BA	SCROLLUP	(→) Moves display one pixel up, checks for corresponding key being pressed.
3F3B9	SCROLLDOWN	(→) Moves display one pixel down, checks for corresponding key being pressed.

3F672	SCROLLLEFT	(→)
		Moves display one pixel left, checks for corresponding key being pressed.
3F673	SCROLLRIGHT	(→)
		Moves display one pixel right, checks for corresponding key being pressed.
3F64F	JUMPTOP	(→)
		Jumps to top of display.
3F64C	JUMPBOT	(→)
		Jumps to bottom of display.
3F64D	JUMPLEFT	(→)
		Jumps to left of display.
3F64E	JUMPRIGHT	(→)
		Jumps to right of display.
3F68F	WINDOWTOP?	(→ flag)
		Is window at the top?
3F68C	WINDOWBOT?	(→ flag)
		Is window at the bottom?
3F68D	WINDOWLEFT?	(→ flag)
		Is window at the left?
3F68E	WINDOWRIGHT?	(→ flag)
		Is window at the right?

4.7.8 Displaying Text

20652	DODISP	(ob %row →)
		Displays any object in specified row.
1E735	DISPROW1	($ →)
		aka: DISP@01, BIGDISPROW1
1E745	DISPROW2	($ →)
		aka: DISP@09, BIGDISPROW2
1E755	DISPROW3	($ →)
		aka: DISP@17, BIGDISPROW3
1E765	DISPROW4	($ →)
		aka: DISP@25, BIGDISPROW4
1E775	DISPROW5	($ →)
1E785	DISPROW6	($ →)
1E795	DISPROW7	($ →)
1E7A5	DISPROW8	($ →)
		May not be possible depending on the size of the font and whether the menu is on or off.
1E701	DISPN	($ #row →)
		aka: BIGDISPN

3F6FA	Disp5x7	($ #start #max →)

Displays string on multiple lines, starting at #start and no using more than #max rows. New lines must be manually specified. Segments longer than 22 characters are truncated and appended with "...".

4.7.9 Messages and Boxes

4.8 Graphics

4.8.1 Built-in Grobs

4.8.2 Dimensions

150A4	GROBDIM	(grob → #height #width)
1047F	GROBDIMw	(grob → #width)
3F320	CKGROBFITS	(g1 g2 #n #m → g1 g2' #n #m)

Shrinks g2 if it does not fit in g1.

3F5F1	CHECKHEIGHT	(grob #height →)

Forces grob (ABUFF/GBUFF) to be at least 64 rows high.

4.8.3 Grob Handling

1D704	GROB!	(grob1 grob2 #x #y →)

Stores grob1 into grob2. Bang type.

3F632	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.

1DAF8	GROB!ZERO	(grob #x1 #y1 #x2 #y2 → grob')

Blanks a rectangular region of the grob. Bang type.

10105	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.

1D9BA	SUBGROB	(grob #x1 #y1 #x2 #y2 → grob')

Returns specified portion of grob.

1EBCA	XYGROBDISP	(#x #y grob →)

Stores grob in HARDBUFF with upper left corner at (#x,#y). HARDBUFF is expanded if necessary.

1F2AE	GROB>GDISP	(grob →)
		Stores new graph grob.
1D622	MAKEGROB	(#height #width → grob)
		Creates a blank grob.
1E5BD	INVGROB	(grob → grob')
		Inverts grob data bits. Bang type.
1FCDE	PIXON	(#x #y →)
		Sets pixel in text grob.
1FCCF	PIXOFF	(#x #y →)
		Clears pixel in text grob.
1FE26	PIXON?	(#x #y → flag)
		Is pixel in text grob on?
1FCB9	PIXON3	(#x #y →)
		Sets pixel in graph grob.
1FCA3	PIXOFF3	(#x #y →)
		Clears pixel in graph grob.
1FE1A	PIXON?3	(#x #y → flag)
		Is pixel in graph grob on?
3F38A	LINEON	(#x1 #y1 #x2 #y2 →)
		Draws a line in text grob.
3F388	LINEOFF	(#x1 #y1 #x2 #y2 →)
		Clears a line in text grob.
3F3D0	TOGLINE	(#x1 #y1 #x2 #y2 →)
		Toggles a line in text grob.
3F38B	LINEON3	(#x1 #y1 #x2 #y2 →)
		Draws a line in graph grob.
3F389	LINEOFF3	(#x1 #y1 #x2 #y2 →)
		Clears a line in graph grob.
3F689	TOGLINE3	(#x1 #y1 #x2 #y2 →)
		Toggles a line in graph grob.
3F338	DOLCD>	(→ grob)
		Returns current display.

4.8.4 Creating Menu Label Grobs

10F64	MakeStdLabel	($ → grob)
		Makes standard menu label.
10FC6	MakeBoxLabel	($ → grob)
		Makes label with a box.
11028	MakeDirLabel	($ → grob)
		Makes directory label.
1108A	MakeInvLabel	($ → grob)
		Makes inverse label.
234C4	Box/StdLabel	($ flag → grob)
		If TRUE makes box label, otherwise makes standard label.

23537	Std/BoxLabel	($ flag → grob)

If `TRUE` makes standard label, otherwise makes box label.

4.8.5 Converting Strings to Grobs

1DD7E	$>BIGGROB	($ → grob)

Makes grob of the string using the large font (5x9).

1DD8B	$>GROB	($ → grob)

Makes grob of the string using the system font. Linefeed does *not* make new line.

1DFAC	$>grob	($ → grob)

Makes grob of the string using the minifont. Linefeed does *not* make new line.

4.9 Plotting

3F31F	CHECKPICT	(→)

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

3F6A3	GETXMIN	(→ %)

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

3F69F	PUTXMIN	(% →)

Sets a new value for XMIN. PPAR is created if necessary.
--
<REF>TEXT:Reserved|PPAR

3F6A1	GETXMAX	(→ %)

Recalls XMAX from the PPAR list if existent. If not, the default PPAR is created in the current directory.
--
<REF>TEXT:Reserved|PPAR

16EB1	PUTXMAX	(% →)

Sets a new value for XMAX. PPAR is created if necessary.
--
<REF>TEXT:Reserved|PPAR

3F6A2	GETYMIN	(→ %)

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

3F69E	PUTYMIN	(% →)
		Sets a new value for YMIN. PPAR is created if necessary.
		--
		<REF>TEXT:Reserved|PPAR
3F6A0	GETYMAX	(→ %)
		Recalls YMAX from the PPAR list if existent. If not, the default PPAR is created in the current directory.
		--
		<REF>TEXT:Reserved|PPAR
16ECA	PUTYMAX	(% →)
		Sets a new value for YMAX. PPAR is created if necessary.
		--
		<REF>TEXT:Reserved|PPAR
3F40D	GETXPOS	
3F40E	GETYPOS	
3F62F	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.

35A6F	TopOuterLoop	
35B58	TOLSaveUI	
35BA3	TOLSetTopicUI	
35C0C	TOLSetViewUI	
35CD9	TOLKeyUI	
35D2E	TOLErrorTrap	
35D60	TOLResUI&Err	
35D79	TOLRestoreUI	
35E0F	?ExitThisTop	
12696	CALCCXT!	(ob $\rightarrow$)
126A6	CALCCXT@	($\rightarrow$ ob)
126B6	PGMCXT!	(ob $\rightarrow$)
126C6	PGMCXT@	($\rightarrow$ ob)
126D6	NOTESCXT!	(ob $\rightarrow$)
126E6	NOTESCXT@	($\rightarrow$ ob)
126F6	apletPTR!	(ob $\rightarrow$)
12706	apletPTR@	($\rightarrow$ ob)
12716	funcPTR!	(ob $\rightarrow$)
12726	funcPTR@	($\rightarrow$ ob)
12736	polarPTR!	(ob $\rightarrow$)
12746	polarPTR@	($\rightarrow$ ob)
12756	paramPTR!	(ob $\rightarrow$)
12766	paramPTR@	($\rightarrow$ ob)
12776	seqPTR!	(ob $\rightarrow$)
12786	seqPTR@	($\rightarrow$ ob)
12796	statPTR!	(ob $\rightarrow$)
127A6	statPTR@	($\rightarrow$ ob)
127B6	solvePTR!	(ob $\rightarrow$)
127C6	solvePTR@	($\rightarrow$ ob)
127D6	otherPTR!	(ob $\rightarrow$)

```
127E6      otherPTR@              ( → ob )
3873F      TopicVarN!
3871B      TopicVarN@
12CA1      TopicVar1!             ( ob → )
12CFC      TopicVar1@             ( → ob )
12CA2      TopicVar2!             ( ob → )
12CFD      TopicVar2@             ( → ob )
12CA3      TopicVar3!             ( ob → )
12CFE      TopicVar3@             ( → ob )
12CA4      TopicVar4!             ( ob → )
12CFF      TopicVar4@             ( → ob )
12CA5      TopicVar5!             ( ob → )
12D00      TopicVar5@             ( → ob )
12CA6      TopicVar6!             ( ob → )
12D01      TopicVar6@             ( → ob )
12CA7      TopicVar7!             ( ob → )
12D02      TopicVar7@             ( → ob )
12CA8      TopicVar8!             ( ob → )
12D03      TopicVar8@             ( → ob )
12CA9      TopicVar9!             ( ob → )
12D04      TopicVar9@             ( → ob )
12CAA      TopicVar10!            ( ob → )
12D05      TopicVar10@            ( → ob )
12CAB      TopicVar11!            ( ob → )
12D06      TopicVar11@            ( → ob )
12CAC      TopicVar12!            ( ob → )
12D07      TopicVar12@            ( → ob )
12CAD      TopicVar13!            ( ob → )
12D08      TopicVar13@            ( → ob )
12CAE      TopicVar14!            ( ob → )
12D09      TopicVar14@            ( → ob )
12CAF      TopicVar15!            ( ob → )
12D0A      TopicVar15@            ( → ob )
12CB0      TopicVar16!            ( ob → )
12D0B      TopicVar16@            ( → ob )
12CB1      TopicVar17!            ( ob → )
12D0C      TopicVar17@            ( → ob )
12CB2      TopicVar18!            ( ob → )
```

```
12D0D      TopicVar18@              ( → ob )
12CB3      TopicVar19!              ( ob → )
12D0E      TopicVar19@              ( → ob )
12CB4      TopicVar20!              ( ob → )
12D0F      TopicVar20@              ( → ob )
12CB5      TopicVar21!              ( ob → )
12D10      TopicVar21@              ( → ob )
12CB6      TopicVar22!              ( ob → )
12D11      TopicVar22@              ( → ob )
12CB7      TopicVar23!              ( ob → )
12D12      TopicVar23@              ( → ob )
12CB8      TopicVar24!              ( ob → )
12D13      TopicVar24@              ( → ob )
12CB9      TopicVar25!              ( ob → )
12D14      TopicVar25@              ( → ob )
12CBA      TopicVar26!              ( ob → )
12D15      TopicVar26@              ( → ob )
12CBB      TopicVar27!              ( ob → )
12D16      TopicVar27@              ( → ob )
12CBC      TopicVar28!              ( ob → )
12D17      TopicVar28@              ( → ob )
12CBD      TopicVar29!              ( ob → )
12D18      TopicVar29@              ( → ob )
12CBE      TopicVar30!              ( ob → )
12D19      TopicVar30@              ( → ob )
12CBF      TopicVar31!              ( ob → )
12D1A      TopicVar31@              ( → ob )
12CC0      TopicVar32!              ( ob → )
12D1B      TopicVar32@              ( → ob )
12CC1      TopicVar33!              ( ob → )
12D1C      TopicVar33@              ( → ob )
12CC2      TopicVar34!              ( ob → )
12D1D      TopicVar34@              ( → ob )
12CC3      TopicVar35!              ( ob → )
12D1E      TopicVar35@              ( → ob )
12CC4      TopicVar36!              ( ob → )
12D1F      TopicVar36@              ( → ob )
12CC5      TopicVar37!              ( ob → )
```

```
12D20       TopicVar37@              ( → ob )
12CC6       TopicVar38!              ( ob → )
12D21       TopicVar38@              ( → ob )
12CC7       TopicVar39!              ( ob → )
12D22       TopicVar39@              ( → ob )
12CC8       TopicVar40!              ( ob → )
12D23       TopicVar40@              ( → ob )
12CC9       TopicVar41!              ( ob → )
12D24       TopicVar41@              ( → ob )
12CCA       TopicVar42!              ( ob → )
12D25       TopicVar42@              ( → ob )
12CCB       TopicVar43!              ( ob → )
12D26       TopicVar43@              ( → ob )
12CCC       TopicVar44!              ( ob → )
12D27       TopicVar44@              ( → ob )
12CCD       TopicVar45!              ( ob → )
12D28       TopicVar45@              ( → ob )
12CCE       TopicVar46!              ( ob → )
12D29       TopicVar46@              ( → ob )
12CCF       TopicVar47!              ( ob → )
12D2A       TopicVar47@              ( → ob )
12CD0       TopicVar48!              ( ob → )
12D2B       TopicVar48@              ( → ob )
12CD1       TopicVar49!              ( ob → )
12D2C       TopicVar49@              ( → ob )
12CD2       TopicVar50!              ( ob → )
12D2D       TopicVar50@              ( → ob )
12CD3       TopicVar51!              ( ob → )
12D2E       TopicVar51@              ( → ob )
12D2F       TopicVar52@              ( ob → )
12CD4       TopicVar52!              ( → ob )
12D30       TopicVar53@              ( ob → )
12CD5       TopicVar53!              ( → ob )
12D31       TopicVar54@              ( ob → )
12CD6       TopicVar54!              ( → ob )
12D32       TopicVar55@              ( ob → )
12CD7       TopicVar55!              ( → ob )
12D33       TopicVar56@              ( ob → )
```

12CD8	TopicVar56!	(→ ob)
12D34	TopicVar57@	(ob →)
12CD9	TopicVar57!	(→ ob)
12D35	TopicVar58@	(ob →)
12CDA	TopicVar58!	(→ ob)
12D36	TopicVar59@	(ob →)
12CDB	TopicVar59!	(→ ob)
12D37	TopicVar60@	(ob →)
12CDC	TopicVar60!	(→ ob)
12D38	TopicVar61@	(ob →)
12CDD	TopicVar61!	(→ ob)
12D39	TopicVar62@	(ob →)
12CDE	TopicVar62!	(→ ob)
12D3A	TopicVar63@	(ob →)
12CDF	TopicVar63!	(→ ob)
12D3B	TopicVar64@	(ob →)
12CE0	TopicVar64!	(→ ob)
12D3C	TopicVar65@	(ob →)
12CE1	TopicVar65!	(→ ob)
12D3D	TopicVar66@	(ob →)
12CE2	TopicVar66!	(→ ob)
12D3E	TopicVar67@	(ob →)
12CE3	TopicVar67!	(→ ob)
12D3F	TopicVar68@	(ob →)
12CE4	TopicVar68!	(→ ob)
12D40	TopicVar69@	(ob →)
12CE5	TopicVar69!	(→ ob)
12D41	TopicVar70@	(ob →)
12CE6	TopicVar70!	(→ ob)
12D42	TopicVar71@	(ob →)
12CE7	TopicVar71!	(→ ob)
12D43	TopicVar72@	(ob →)
12CE8	TopicVar72!	(→ ob)
12D44	TopicVar73@	(ob →)
12CE9	TopicVar73!	(→ ob)
12D45	TopicVar74@	(ob →)
12CEA	TopicVar74!	(→ ob)
12D46	TopicVar75@	(ob →)

```
12CEB      TopicVar75!              ( → ob )
12D47      TopicVar76@              ( ob → )
12CEC      TopicVar76!              ( → ob )
12D48      TopicVar77@              ( ob → )
12CED      TopicVar77!              ( → ob )
12D49      TopicVar78@              ( ob → )
12CEE      TopicVar78!              ( → ob )
12D4A      TopicVar79@              ( ob → )
12CEF      TopicVar79!              ( → ob )
12D4B      TopicVar80@              ( ob → )
12CF0      TopicVar80!              ( → ob )
12D4C      TopicVar81@              ( ob → )
12CF1      TopicVar81!              ( → ob )
12D4D      TopicVar82@              ( ob → )
12CF2      TopicVar82!              ( → ob )
12D4E      TopicVar83@              ( ob → )
12CF3      TopicVar83!              ( → ob )
12D4F      TopicVar84@              ( ob → )
12CF4      TopicVar84!              ( → ob )
12D50      TopicVar85@              ( ob → )
12CF5      TopicVar85!              ( → ob )
12D51      TopicVar86@              ( ob → )
12CF6      TopicVar86!              ( → ob )
12D52      TopicVar87@              ( ob → )
12CF7      TopicVar87!              ( → ob )
12D53      TopicVar88@              ( ob → )
12CF8      TopicVar88!              ( → ob )
12D54      TopicVar89@              ( ob → )
12CF9      TopicVar89!              ( → ob )
12D55      TopicVar90@              ( ob → )
12CFA      TopicVar90!              ( → ob )
12CFB      TopicVar91!              ( ob → )
12D56      TopicVar91@              ( → ob )
12D57      TOLVar1!                 ( ob → )
12E2F      TOLVar1@                 ( → ob )
12D58      TOLVar2!                 ( ob → )
12E30      TOLVar2@                 ( → ob )
12D59      TOLVar3!                 ( ob → )
```

```
12E31       TOLVar3@                ( → ob )
12D5A       TOLVar4!                ( ob → )
12E32       TOLVar4@                ( → ob )
12D5B       TOLVar5!                ( ob → )
12E33       TOLVar5@                ( → ob )
12D5C       TOLVar6!                ( ob → )
12E34       TOLVar6@                ( → ob )
12D5D       TOLVar7!                ( ob → )
12E35       TOLVar7@                ( → ob )
12D5E       TOLVar8!                ( ob → )
12E36       TOLVar8@                ( → ob )
12D5F       TOLVar9!                ( ob → )
12E37       TOLVar9@                ( → ob )
12D60       TOLVar10!               ( ob → )
12E38       TOLVar10@               ( → ob )
12D61       TOLVar11!               ( ob → )
12E39       TOLVar11@               ( → ob )
12D62       TOLVar12!               ( ob → )
12E3A       TOLVar12@               ( → ob )
12D63       TOLVar13!               ( ob → )
12E3B       TOLVar13@               ( → ob )
12D64       TOLVar14!               ( ob → )
12E3C       TOLVar14@               ( → ob )
12D65       TOLVar15!               ( ob → )
12E3D       TOLVar15@               ( → ob )
12D66       TOLVar16!               ( ob → )
12E3E       TOLVar16@               ( → ob )
12D67       TOLVar17!               ( ob → )
12E3F       TOLVar17@               ( → ob )
12D68       TOLVar18!               ( ob → )
12E40       TOLVar18@               ( → ob )
12D69       TOLVar19!               ( ob → )
12E41       TOLVar19@               ( → ob )
12D6A       TOLVar20!               ( ob → )
12E42       TOLVar20@               ( → ob )
12D6B       TOLVar21!               ( ob → )
12E43       TOLVar21@               ( → ob )
12D6C       TOLVar22!               ( ob → )
```

```
12E44       TOLVar22@                   ( → ob )
12D6D       TOLVar23!                   ( ob → )
12E45       TOLVar23@                   ( → ob )
12D6E       TOLVar24!                   ( ob → )
12E46       TOLVar24@                   ( → ob )
12D6F       TOLVar25!                   ( ob → )
12E47       TOLVar25@                   ( → ob )
12D70       TOLVar26!                   ( ob → )
12E48       TOLVar26@                   ( → ob )
12D71       TOLVar27!                   ( ob → )
12E49       TOLVar27@                   ( → ob )
12D72       TOLVar28!                   ( ob → )
12E4A       TOLVar28@                   ( → ob )
12D73       TOLVar29!                   ( ob → )
12E4B       TOLVar29@                   ( → ob )
12D74       TOLVar30!                   ( ob → )
12E4C       TOLVar30@                   ( → ob )
12D75       TOLVar31!                   ( ob → )
12E4D       TOLVar31@                   ( → ob )
12D76       TOLVar32!                   ( ob → )
12E4E       TOLVar32@                   ( → ob )
12D77       TOLVar33!                   ( ob → )
12E4F       TOLVar33@                   ( → ob )
12D78       TOLVar34!                   ( ob → )
12E50       TOLVar34@                   ( → ob )
12D79       TOLVar35!                   ( ob → )
12E51       TOLVar35@                   ( → ob )
12D7A       TOLVar36!                   ( ob → )
12E52       TOLVar36@                   ( → ob )
12D7B       TOLVar37!                   ( ob → )
12E53       TOLVar37@                   ( → ob )
12D7C       TOLVar38!                   ( ob → )
12E54       TOLVar38@                   ( → ob )
12D7D       TOLVar39!                   ( ob → )
12E55       TOLVar39@                   ( → ob )
12D7E       TOLVar40!                   ( ob → )
12E56       TOLVar40@                   ( → ob )
12D7F       TOLVar41!                   ( ob → )
```

```
12E57      TOLVar41@              ( → ob )
12D80      TOLVar42!             ( ob → )
12E58      TOLVar42@              ( → ob )
12D81      TOLVar43!             ( ob → )
12E59      TOLVar43@              ( → ob )
12D82      TOLVar44!             ( ob → )
12E5A      TOLVar44@              ( → ob )
12D83      TOLVar45!             ( ob → )
12E5B      TOLVar45@              ( → ob )
12D84      TOLVar46!             ( ob → )
12E5C      TOLVar46@              ( → ob )
12D85      TOLVar47!             ( ob → )
12E5D      TOLVar47@              ( → ob )
12D86      TOLVar48!             ( ob → )
12E5E      TOLVar48@              ( → ob )
12D87      TOLVar49!             ( ob → )
12E5F      TOLVar49@              ( → ob )
12D88      TOLVar50!             ( ob → )
12E60      TOLVar50@              ( → ob )
12D89      TOLVar51!             ( ob → )
12E61      TOLVar51@              ( → ob )
12D8A      TOLVar52!             ( ob → )
12E62      TOLVar52@              ( → ob )
12D8B      TOLVar53!             ( ob → )
12E63      TOLVar53@              ( → ob )
12D8C      TOLVar54!             ( ob → )
12E64      TOLVar54@              ( → ob )
12D8D      TOLVar55!             ( ob → )
12E65      TOLVar55@              ( → ob )
12D8E      TOLVar56!             ( ob → )
12E66      TOLVar56@              ( → ob )
12D8F      TOLVar57!             ( ob → )
12E67      TOLVar57@              ( → ob )
12D90      TOLVar58!             ( ob → )
12E68      TOLVar58@              ( → ob )
12D91      TOLVar59!             ( ob → )
12E69      TOLVar59@              ( → ob )
12D92      TOLVar60!             ( ob → )
```

12E6A	TOLVar60@	(→ ob)
12D93	TOLVar61!	(ob →)
12E6B	TOLVar61@	(→ ob)
12D94	TOLVar62!	(ob →)
12E6C	TOLVar62@	(→ ob)
12D95	TOLVar63!	(ob →)
12E6D	TOLVar63@	(→ ob)
12D96	TOLVar64!	(ob →)
12E6E	TOLVar64@	(→ ob)
12D97	TOLVar65!	(ob →)
12E6F	TOLVar65@	(→ ob)
12D98	TOLVar66!	(ob →)
12E70	TOLVar66@	(→ ob)
12D99	TOLVar67!	(ob →)
12E71	TOLVar67@	(→ ob)
12D9A	TOLVar68!	(ob →)
12E72	TOLVar68@	(→ ob)
12D9B	TOLVar69!	(ob →)
12E73	TOLVar69@	(→ ob)
12D9C	TOLVar70!	(ob →)
12E74	TOLVar70@	(→ ob)
12D9D	TOLVar71!	(ob →)
12E75	TOLVar71@	(→ ob)
12D9E	TOLVar72!	(ob →)
12E76	TOLVar72@	(→ ob)
12D9F	TOLVar73!	(ob →)
12E77	TOLVar73@	(→ ob)
12DA0	TOLVar74!	(ob →)
12E78	TOLVar74@	(→ ob)
12DA1	TOLVar75!	(ob →)
12E79	TOLVar75@	(→ ob)
12DA2	TOLVar76!	(ob →)
12E7A	TOLVar76@	(→ ob)
12DA3	TOLVar77!	(ob →)
12E7B	TOLVar77@	(→ ob)
12DA4	TOLVar78!	(ob →)
12E7C	TOLVar78@	(→ ob)
12DA5	TOLVar79!	(ob →)

```
12E7D       TOLVar79@                    ( → ob )
12DA6       TOLVar80!                    ( ob → )
12E7E       TOLVar80@                    ( → ob )
12DA7       TOLVar81!                    ( ob → )
12E7F       TOLVar81@                    ( → ob )
12DA8       TOLVar82!                    ( ob → )
12E80       TOLVar82@                    ( → ob )
12DA9       TOLVar83!                    ( ob → )
12E81       TOLVar83@                    ( → ob )
12DAA       TOLVar84!                    ( ob → )
12E82       TOLVar84@                    ( → ob )
12DAB       TOLVar85!                    ( ob → )
12E83       TOLVar85@                    ( → ob )
12DAC       TOLVar86!                    ( ob → )
12E84       TOLVar86@                    ( → ob )
12DAD       TOLVar87!                    ( ob → )
12E85       TOLVar87@                    ( → ob )
12DAE       TOLVar88!                    ( ob → )
12E86       TOLVar88@                    ( → ob )
12DAF       TOLVar89!                    ( ob → )
12E87       TOLVar89@                    ( → ob )
12DB0       TOLVar90!                    ( ob → )
12E88       TOLVar90@                    ( → ob )
12DB1       TOLVar91!                    ( ob → )
12E89       TOLVar91@                    ( → ob )
12DB2       TOLVar92!                    ( ob → )
12E8A       TOLVar92@                    ( → ob )
12DB3       TOLVar93!                    ( ob → )
12E8B       TOLVar93@                    ( → ob )
12DB4       TOLVar94!                    ( ob → )
12E8C       TOLVar94@                    ( → ob )
12DB5       TOLVar95!                    ( ob → )
12E8D       TOLVar95@                    ( → ob )
12DB6       TOLVar96!                    ( ob → )
12E8E       TOLVar96@                    ( → ob )
12DB7       TOLVar97!                    ( ob → )
12E8F       TOLVar97@                    ( → ob )
12DB8       TOLVar98!                    ( ob → )
```

12E90	TOLVar98@	(→ ob)
12DB9	TOLVar99!	(ob →)
12E91	TOLVar99@	(→ ob)
12DBA	TOLVar100!	(ob →)
12E92	TOLVar100@	(→ ob)
12DBB	TOLVar101!	(ob →)
12E93	TOLVar101@	(→ ob)
12DBC	TOLVar102!	(ob →)
12E94	TOLVar102@	(→ ob)
12DBD	TOLVar103!	(ob →)
12E95	TOLVar103@	(→ ob)
12DBE	TOLVar104!	(ob →)
12E96	TOLVar104@	(→ ob)
12DBF	TOLVar105!	(ob →)
12E97	TOLVar105@	(→ ob)
12DC0	TOLVar106!	(ob →)
12E98	TOLVar106@	(→ ob)
12DC1	TOLVar107!	(ob →)
12E99	TOLVar107@	(→ ob)
12DC2	TOLVar108!	(ob →)
12E9A	TOLVar108@	(→ ob)
12DC3	TOLVar109!	(ob →)
12E9B	TOLVar109@	(→ ob)
12DC4	TOLVar110!	(ob →)
12E9C	TOLVar110@	(→ ob)
12DC5	TOLVar111!	(ob →)
12E9D	TOLVar111@	(→ ob)
12DC6	TOLVar112!	(ob →)
12E9E	TOLVar112@	(→ ob)
12DC7	TOLVar113!	(ob →)
12E9F	TOLVar113@	(→ ob)
12DC8	TOLVar114!	(ob →)
12EA0	TOLVar114@	(→ ob)
12DC9	TOLVar115!	(ob →)
12EA1	TOLVar115@	(→ ob)
12DCA	TOLVar116!	(ob →)
12EA2	TOLVar116@	(→ ob)
12DCB	TOLVar117!	(ob →)

```
12EA3      TOLVar117@              ( → ob )
12DCC      TOLVar118!             ( ob → )
12EA4      TOLVar118@              ( → ob )
12DCD      TOLVar119!             ( ob → )
12EA5      TOLVar119@              ( → ob )
12DCE      TOLVar120!             ( ob → )
12EA6      TOLVar120@              ( → ob )
12DCF      TOLVar121!             ( ob → )
12EA7      TOLVar121@              ( → ob )
12DD0      TOLVar122!             ( ob → )
12EA8      TOLVar122@              ( → ob )
12DD1      TOLVar123!             ( ob → )
12EA9      TOLVar123@              ( → ob )
12DD2      TOLVar124!             ( ob → )
12EAA      TOLVar124@              ( → ob )
12DD3      TOLVar125!             ( ob → )
12EAB      TOLVar125@              ( → ob )
12DD4      TOLVar126!             ( ob → )
12EAC      TOLVar126@              ( → ob )
12DD5      TOLVar127!             ( ob → )
12EAD      TOLVar127@              ( → ob )
12DD6      TOLVar128!             ( ob → )
12EAE      TOLVar128@              ( → ob )
12DD7      TOLVar129!             ( ob → )
12EAF      TOLVar129@              ( → ob )
12DD8      TOLVar130!             ( ob → )
12EB0      TOLVar130@              ( → ob )
12DD9      TOLVar131!             ( ob → )
12EB1      TOLVar131@              ( → ob )
12DDA      TOLVar132!             ( ob → )
12EB2      TOLVar132@              ( → ob )
12DDB      TOLVar133!             ( ob → )
12EB3      TOLVar133@              ( → ob )
12DDC      TOLVar134!             ( ob → )
12EB4      TOLVar134@              ( → ob )
12DDD      TOLVar135!             ( ob → )
12EB5      TOLVar135@              ( → ob )
12DDE      TOLVar136!             ( ob → )
```

```
12EB6      TOLVar136@              ( → ob )
12DDF      TOLVar137!             ( ob → )
12EB7      TOLVar137@              ( → ob )
12DE0      TOLVar138!             ( ob → )
12EB8      TOLVar138@              ( → ob )
12DE1      TOLVar139!             ( ob → )
12EB9      TOLVar139@              ( → ob )
12DE2      TOLVar140!             ( ob → )
12EBA      TOLVar140@              ( → ob )
12DE3      TOLVar141!             ( ob → )
12EBB      TOLVar141@              ( → ob )
12DE4      TOLVar142!             ( ob → )
12EBC      TOLVar142@              ( → ob )
12DE5      TOLVar143!             ( ob → )
12EBD      TOLVar143@              ( → ob )
12DE6      TOLVar144!             ( ob → )
12EBE      TOLVar144@              ( → ob )
12DE7      TOLVar145!             ( ob → )
12EBF      TOLVar145@              ( → ob )
12DE8      TOLVar146!             ( ob → )
12EC0      TOLVar146@              ( → ob )
12DE9      TOLVar147!             ( ob → )
12EC1      TOLVar147@              ( → ob )
12DEA      TOLVar148!             ( ob → )
12EC2      TOLVar148@              ( → ob )
12DEB      TOLVar149!             ( ob → )
12EC3      TOLVar149@              ( → ob )
12DEC      TOLVar150!             ( ob → )
12EC4      TOLVar150@              ( → ob )
12DED      TOLVar151!             ( ob → )
12EC5      TOLVar151@              ( → ob )
12DEE      TOLVar152!             ( ob → )
12EC6      TOLVar152@              ( → ob )
12DEF      TOLVar153!             ( ob → )
12EC7      TOLVar153@              ( → ob )
12DF0      TOLVar154!             ( ob → )
12EC8      TOLVar154@              ( → ob )
12DF1      TOLVar155!             ( ob → )
```

```
12EC9      TOLVar155@              ( → ob )
12DF2      TOLVar156!             ( ob → )
12ECA      TOLVar156@              ( → ob )
12DF3      TOLVar157!             ( ob → )
12ECB      TOLVar157@              ( → ob )
12DF4      TOLVar158!             ( ob → )
12ECC      TOLVar158@              ( → ob )
12DF5      TOLVar159!             ( ob → )
12ECD      TOLVar159@              ( → ob )
12DF6      TOLVar160!             ( ob → )
12ECE      TOLVar160@              ( → ob )
12DF7      TOLVar161!             ( ob → )
12ECF      TOLVar161@              ( → ob )
12DF8      TOLVar162!             ( ob → )
12ED0      TOLVar162@              ( → ob )
12DF9      TOLVar163!             ( ob → )
12ED1      TOLVar163@              ( → ob )
12DFA      TOLVar164!             ( ob → )
12ED2      TOLVar164@              ( → ob )
12DFB      TOLVar165!             ( ob → )
12ED3      TOLVar165@              ( → ob )
12DFC      TOLVar166!             ( ob → )
12ED4      TOLVar166@              ( → ob )
12DFD      TOLVar167!             ( ob → )
12ED5      TOLVar167@              ( → ob )
12DFE      TOLVar168!             ( ob → )
12ED6      TOLVar168@              ( → ob )
12DFF      TOLVar169!             ( ob → )
12ED7      TOLVar169@              ( → ob )
12E00      TOLVar170!             ( ob → )
12ED8      TOLVar170@              ( → ob )
12E01      TOLVar171!             ( ob → )
12ED9      TOLVar171@              ( → ob )
12E02      TOLVar172!             ( ob → )
12EDA      TOLVar172@              ( → ob )
12E03      TOLVar173!             ( ob → )
12EDB      TOLVar173@              ( → ob )
12E04      TOLVar174!             ( ob → )
```

```
12EDC      TOLVar174@                ( → ob )
12E05      TOLVar175!                ( ob → )
12EDD      TOLVar175@                ( → ob )
12E06      TOLVar176!                ( ob → )
12EDE      TOLVar176@                ( → ob )
12E07      TOLVar177!                ( ob → )
12EDF      TOLVar177@                ( → ob )
12E08      TOLVar178!                ( ob → )
12EE0      TOLVar178@                ( → ob )
12E09      TOLVar179!                ( ob → )
12EE1      TOLVar179@                ( → ob )
12E0A      TOLVar180!                ( ob → )
12EE2      TOLVar180@                ( → ob )
12E0B      TOLVar181!                ( ob → )
12EE3      TOLVar181@                ( → ob )
12E0C      TOLVar182!                ( ob → )
12EE4      TOLVar182@                ( → ob )
12E0D      TOLVar183!                ( ob → )
12EE5      TOLVar183@                ( → ob )
12E0E      TOLVar184!                ( ob → )
12EE6      TOLVar184@                ( → ob )
12E0F      TOLVar185!                ( ob → )
12EE7      TOLVar185@                ( → ob )
12E10      TOLVar186!                ( ob → )
12EE8      TOLVar186@                ( → ob )
12E11      TOLVar187!                ( ob → )
12EE9      TOLVar187@                ( → ob )
12E12      TOLVar188!                ( ob → )
12EEA      TOLVar188@                ( → ob )
12E13      TOLVar189!                ( ob → )
12EEB      TOLVar189@                ( → ob )
12E14      TOLVar190!                ( ob → )
12EEC      TOLVar190@                ( → ob )
12E15      TOLVar191!                ( ob → )
12EED      TOLVar191@                ( → ob )
12E16      TOLVar192!                ( ob → )
12EEE      TOLVar192@                ( → ob )
12E17      TOLVar193!                ( ob → )
```

```
12EEF     TOLVar193@              ( → ob )
12E18     TOLVar194!             ( ob → )
12EF0     TOLVar194@              ( → ob )
12E19     TOLVar195!             ( ob → )
12EF1     TOLVar195@              ( → ob )
12E1A     TOLVar196!             ( ob → )
12EF2     TOLVar196@              ( → ob )
12E1B     TOLVar197!             ( ob → )
12EF3     TOLVar197@              ( → ob )
12E1C     TOLVar198!             ( ob → )
12EF4     TOLVar198@              ( → ob )
12E1D     TOLVar199!             ( ob → )
12EF5     TOLVar199@              ( → ob )
12E1E     TOLVar200!             ( ob → )
12EF6     TOLVar200@              ( → ob )
12E1F     TOLVar201!             ( ob → )
12EF7     TOLVar201@              ( → ob )
12E20     TOLVar202!             ( ob → )
12EF8     TOLVar202@              ( → ob )
12E21     TOLVar203!             ( ob → )
12EF9     TOLVar203@              ( → ob )
12E22     TOLVar204!             ( ob → )
12EFA     TOLVar204@              ( → ob )
12E23     TOLVar205!             ( ob → )
12EFB     TOLVar205@              ( → ob )
12E24     TOLVar206!             ( ob → )
12EFC     TOLVar206@              ( → ob )
12E25     TOLVar207!             ( ob → )
12EFD     TOLVar207@              ( → ob )
12E26     TOLVar208!             ( ob → )
12EFE     TOLVar208@              ( → ob )
12E27     TOLVar209!             ( ob → )
12EFF     TOLVar209@              ( → ob )
12E28     TOLVar210!             ( ob → )
12F00     TOLVar210@              ( → ob )
12E29     TOLVar211!             ( ob → )
12F01     TOLVar211@              ( → ob )
12E2A     TOLVar212!             ( ob → )
```

12F02	TOLVar212@	(→ ob)
12E2B	TOLVar213!	(ob →)
12F03	TOLVar213@	(→ ob)
12E2C	TOLVar214!	(ob →)
12F04	TOLVar214@	(→ ob)
12E2D	TOLVar215!	(ob →)
12F05	TOLVar215@	(→ ob)
12E2E	TOLVar216!	(ob →)
12F06	TOLVar216@	(→ ob)
1281D	TOLVarN!	(ob →)
12840	TOLVarN@	(→ ob)
12863	ClrAllTVars	
12899	ClrAllTOLVs	
128B5	%0AllTopicVs	
128F0	%0AllTOLVars	
1292B	TOLVarSet!	
129F8	%0TOLVarSet	
12A48	1getcxt!	
12A5C	DoInCxt	
12AB6	DoInCalcCxt	
12ACF	DoInAppCxt	
12AE8	DoInFuncCxt	
12B01	DoInPolarCxt	
12B1A	DoInParamCxt	
12B33	DoInSeqCxt	
12B4C	DoInStatCxt	
12B65	DoInSolveCxt	
12B7E	DoInOtherCxt	
12C4B	otherNG?	
12C7E	GET@tTYPER	

5.2 Special Variables

26F25	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

```
0820F2    ~xa                        ( → id )
                                     The id corresponding to the real variable A.
0830F2    ~xb                        ( → id )
                                     The id corresponding to the real variable B.
0840F2    ~xc                        ( → id )
                                     The id corresponding to the real variable C.
0850F2    ~xd                        ( → id )
                                     The id corresponding to the real variable D.
0870F2    ~xf                        ( → id )
                                     The id corresponding to the real variable F.
0880F2    ~xg                        ( → id )
                                     The id corresponding to the real variable G.
0890F2    ~xh                        ( → id )
                                     The id corresponding to the real variable H.
08B0F2    ~xj                        ( → id )
                                     The id corresponding to the real variable J.
08C0F2    ~xk                        ( → id )
                                     The id corresponding to the real variable K.
08D0F2    ~xl                        ( → id )
                                     The id corresponding to the real variable L.
08E0F2    ~xm                        ( → id )
                                     The id corresponding to the real variable M.
08F0F2    ~xn                        ( → id )
                                     The id corresponding to the real variable N.
0900F2    ~xo                        ( → id )
                                     The id corresponding to the real variable O.
0910F2    ~xp                        ( → id )
                                     The id corresponding to the real variable P.
0920F2    ~xq                        ( → id )
                                     The id corresponding to the real variable Q.
0930F2    ~xr                        ( → id )
                                     The id corresponding to the real variable R.
0940F2    ~xs                        ( → id )
                                     The id corresponding to the real variable S.
0950F2    ~xt                        ( → id )
                                     The id corresponding to the real variable T.
0960F2    ~xu                        ( → id )
                                     The id corresponding to the real variable U.
0970F2    ~xv                        ( → id )
                                     The id corresponding to the real variable V.
0980F2    ~xw                        ( → id )
                                     The id corresponding to the real variable W.
```

```
0990F2    ~xx                     ( → id )
```
The id corresponding to the real variable X.
```
09A0F2    ~xy                     ( → id )
```
The id corresponding to the real variable Y.
```
09B0F2    ~xz                     ( → id )
```
The id corresponding to the real variable Z.
```
09C0F2    ~xtheta                 ( → id )
```
The id corresponding to the real variable θ.

5.2.2 Complex HOME variableS

```
0B30F2    ~xz0                    ( → id )
```
The id corresponding to the complex variable Z0.
```
0B40F2    ~xz1                    ( → id )
```
The id corresponding to the complex variable Z1.
```
0B50F2    ~xz2                    ( → id )
```
The id corresponding to the complex variable Z2.
```
0B60F2    ~xz3                    ( → id )
```
The id corresponding to the complex variable Z3.
```
0B70F2    ~xz4                    ( → id )
```
The id corresponding to the complex variable Z4.
```
0B80F2    ~xz5                    ( → id )
```
The id corresponding to the complex variable Z5.
```
0B90F2    ~xz6                    ( → id )
```
The id corresponding to the complex variable Z6.
```
0BA0F2    ~xz7                    ( → id )
```
The id corresponding to the complex variable Z7.
```
0BB0F2    ~xz8                    ( → id )
```
The id corresponding to the complex variable Z8.
```
0BC0F2    ~xz9                    ( → id )
```
The id corresponding to the complex variable Z9.

5.2.3 Matrix HOME variables

```
0BD0F2    ~xm0                    ( → id )
```
The id corresponding to the matrix variable M0.
```
0BE0F2    ~xm1                    ( → id )
```
The id corresponding to the matrix variable M1.
```
0BF0F2    ~xm2                    ( → id )
```
The id corresponding to the matrix variable M2.
```
0C00F2    ~xm3                    ( → id )
```
The id corresponding to the matrix variable M3.
```
0C10F2    ~xm4                    ( → id )
```
The id corresponding to the matrix variable M4.

0C20F2	~xm5	(→ id)
		The id corresponding to the matrix variable M5.
0C30F2	~xm6	(→ id)
		The id corresponding to the matrix variable M6.
0C40F2	~xm7	(→ id)
		The id corresponding to the matrix variable M7.
0C50F2	~xm8	(→ id)
		The id corresponding to the matrix variable M8.
0C60F2	~xm9	(→ id)
		The id corresponding to the matrix variable M9.

5.2.4 Graphical HOME variables

0D10F2	~xg0	(→ id)
		The id corresponding to the graphics variable G0.
0D20F2	~xg1	(→ id)
		The id corresponding to the graphics variable G1.
0D30F2	~xg2	(→ id)
		The id corresponding to the graphics variable G2.
0D40F2	~xg3	(→ id)
		The id corresponding to the graphics variable G3.
0D50F2	~xg4	(→ id)
		The id corresponding to the graphics variable G4.
0D60F2	~xg5	(→ id)
		The id corresponding to the graphics variable G5.
0D70F2	~xg6	(→ id)
		The id corresponding to the graphics variable G6.
0D80F2	~xg7	(→ id)
		The id corresponding to the graphics variable G7.
0D90F2	~xg8	(→ id)
		The id corresponding to the graphics variable G8.
0DA0F2	~xg9	(→ id)
		The id corresponding to the graphics variable G9.

5.2.5 List HOME variables

0C70F2	~xl0	(→ id)
		The id corresponding to the list variable L0.
0C80F2	~xl1	(→ id)
		The id corresponding to the list variable L1.
0C90F2	~xl2	(→ id)
		The id corresponding to the list variable L2.
0CA0F2	~xl3	(→ id)
		The id corresponding to the list variable L3.

0CB0F2	~xl4	(→ id)
		The id corresponding to the list variable L4.
0CC0F2	~xl5	(→ id)
		The id corresponding to the list variable L5.
0CD0F2	~xl6	(→ id)
		The id corresponding to the list variable L6.
0CE0F2	~xl7	(→ id)
		The id corresponding to the list variable L7.
0CF0F2	~xl8	(→ id)
		The id corresponding to the list variable L8.
0D00F2	~xl9	(→ id)
		The id corresponding to the list variable L9.
3F604	ClearList0	(→)
		Clears list 0.
3F605	ClearList1	(→)
		Clears list 1.
3F606	ClearList2	(→)
		Clears list 2.
3F607	ClearList3	(→)
		Clears list 3.
3F608	ClearList4	(→)
		Clears list 4.
3F609	ClearList5	(→)
		Clears list 5.
3F60A	ClearList6	(→)
		Clears list 6.
3F60B	ClearList7	(→)
		Clears list 7.
3F60C	ClearList8	(→)
		Clears list 8.
3F60D	ClearList9	(→)
		Clears list 9.
3F60E	ClrListUtil	
3F5C0	ClrInAplet	

5.2.6 FUNCTION applet

0260F2	~xF0	(→ id)
		The id corresponding to the FUNCTION applet variable F0.
0140F2	~xF1	(→ id)
		The id corresponding to the FUNCTION applet variable F1.
0160F2	~xF2	(→ id)
		The id corresponding to the FUNCTION applet variable F2.

```
0180F2    ~xF3                        ( → id )
```
The id corresponding to the FUNCTION applet variable F3.
```
01A0F2    ~xF4                        ( → id )
```
The id corresponding to the FUNCTION applet variable F4.
```
01C0F2    ~xF5                        ( → id )
```
The id corresponding to the FUNCTION applet variable F5.
```
01E0F2    ~xF6                        ( → id )
```
The id corresponding to the FUNCTION applet variable F6.
```
0200F2    ~xF7                        ( → id )
```
The id corresponding to the FUNCTION applet variable F7.
```
0220F2    ~xF8                        ( → id )
```
The id corresponding to the FUNCTION applet variable F8.
```
0240F2    ~xF9                        ( → id )
```
The id corresponding to the FUNCTION applet variable F9.
```
0270F2    ~idF0                       ( → id )
```
The id corresponding to the FUNCTION applet variable F0.
```
0150F2    ~idF1                       ( → id )
```
The id corresponding to the FUNCTION applet variable F1.
```
0170F2    ~idF2                       ( → id )
```
The id corresponding to the FUNCTION applet variable F2.
```
0190F2    ~idF3                       ( → id )
```
The id corresponding to the FUNCTION applet variable F3.
```
01B0F2    ~idF4                       ( → id )
```
The id corresponding to the FUNCTION applet variable F4.
```
01D0F2    ~idF5                       ( → id )
```
The id corresponding to the FUNCTION applet variable F5.
```
01F0F2    ~idF6                       ( → id )
```
The id corresponding to the FUNCTION applet variable F6.
```
0210F2    ~idF7                       ( → id )
```
The id corresponding to the FUNCTION applet variable F7.
```
0230F2    ~idF8                       ( → id )
```
The id corresponding to the FUNCTION applet variable F8.

```
0250F2    ~idF9                    ( → id )
```
The id corresponding to the `FUNCTION` applet variable F9.

5.2.7 PARAMETRIC applet

```
04C0F2    ~xX0                     ( → id )
```
The id corresponding to the `PARAMETRIC` applet variable X0.
```
0280F2    ~xX1                     ( → id )
```
The id corresponding to the `PARAMETRIC` applet variable X1.
```
02C0F2    ~xX2                     ( → id )
```
The id corresponding to the `PARAMETRIC` applet variable X2.
```
0300F2    ~xX3                     ( → id )
```
The id corresponding to the `PARAMETRIC` applet variable X3.
```
0340F2    ~xX4                     ( → id )
```
The id corresponding to the `PARAMETRIC` applet variable X4.
```
0380F2    ~xX5                     ( → id )
```
The id corresponding to the `PARAMETRIC` applet variable X5.
```
03C0F2    ~xX6                     ( → id )
```
The id corresponding to the `PARAMETRIC` applet variable X6.
```
0400F2    ~xX7                     ( → id )
```
The id corresponding to the `PARAMETRIC` applet variable X7.
```
0440F2    ~xX8                     ( → id )
```
The id corresponding to the `PARAMETRIC` applet variable X8.
```
0480F2    ~xX9                     ( → id )
```
The id corresponding to the `PARAMETRIC` applet variable X9.
```
04E0F2    ~xY0                     ( → id )
```
The id corresponding to the `PARAMETRIC` applet variable Y0.
```
02A0F2    ~xY1                     ( → id )
```
The id corresponding to the `PARAMETRIC` applet variable Y1.
```
02E0F2    ~xY2                     ( → id )
```
The id corresponding to the `PARAMETRIC` applet variable Y2.
```
0320F2    ~xY3                     ( → id )
```
The id corresponding to the `PARAMETRIC` applet variable Y3.

```
0360F2     ˜xY4                          ( → id )
```
The id corresponding to the PARAMETRIC applet variable Y4.
```
03A0F2     ˜xY5                          ( → id )
```
The id corresponding to the PARAMETRIC applet variable Y5.
```
03E0F2     ˜xY6                          ( → id )
```
The id corresponding to the PARAMETRIC applet variable Y6.
```
0420F2     ˜xY7                          ( → id )
```
The id corresponding to the PARAMETRIC applet variable Y7.
```
0460F2     ˜xY8                          ( → id )
```
The id corresponding to the PARAMETRIC applet variable Y8.
```
04A0F2     ˜xY9                          ( → id )
```
The id corresponding to the PARAMETRIC applet variable Y9.
```
04D0F2     ˜idX0                         ( → id )
```
The id corresponding to the PARAMETRIC applet variable X0.
```
0290F2     ˜idX1                         ( → id )
```
The id corresponding to the PARAMETRIC applet variable X1.
```
02D0F2     ˜idX2                         ( → id )
```
The id corresponding to the PARAMETRIC applet variable X2.
```
0310F2     ˜idX3                         ( → id )
```
The id corresponding to the PARAMETRIC applet variable X3.
```
0350F2     ˜idX4                         ( → id )
```
The id corresponding to the PARAMETRIC applet variable X4.
```
0390F2     ˜idX5                         ( → id )
```
The id corresponding to the PARAMETRIC applet variable X5.
```
03D0F2     ˜idX6                         ( → id )
```
The id corresponding to the PARAMETRIC applet variable X6.
```
0410F2     ˜idX7                         ( → id )
```
The id corresponding to the PARAMETRIC applet variable X7.
```
0450F2     ˜idX8                         ( → id )
```
The id corresponding to the PARAMETRIC applet variable X8.
```
0490F2     ˜idX9                         ( → id )
```
The id corresponding to the PARAMETRIC applet variable X9.

```
04F0F2    ~idY0                      ( → id )
```
The id corresponding to the `PARAMETRIC` applet variable Y0.
```
02B0F2    ~idY1                      ( → id )
```
The id corresponding to the `PARAMETRIC` applet variable Y1.
```
02F0F2    ~idY2                      ( → id )
```
The id corresponding to the `PARAMETRIC` applet variable Y2.
```
0330F2    ~idY3                      ( → id )
```
The id corresponding to the `PARAMETRIC` applet variable Y3.
```
0370F2    ~idY4                      ( → id )
```
The id corresponding to the `PARAMETRIC` applet variable Y4.
```
03B0F2    ~idY5                      ( → id )
```
The id corresponding to the `PARAMETRIC` applet variable Y5.
```
03F0F2    ~idY6                      ( → id )
```
The id corresponding to the `PARAMETRIC` applet variable Y6.
```
0430F2    ~idY7                      ( → id )
```
The id corresponding to the `PARAMETRIC` applet variable Y7.
```
0470F2    ~idY8                      ( → id )
```
The id corresponding to the `PARAMETRIC` applet variable Y8.
```
04B0F2    ~idY9                      ( → id )
```
The id corresponding to the `PARAMETRIC` applet variable Y9.

5.2.8 POLAR applet

```
0620F2    ~xR0                       ( → id )
```
The id corresponding to the `POLAR` applet variable R0.
```
0500F2    ~xR1                       ( → id )
```
The id corresponding to the `POLAR` applet variable R1.
```
0520F2    ~xR2                       ( → id )
```
The id corresponding to the `POLAR` applet variable R2.
```
0540F2    ~xR3                       ( → id )
```
The id corresponding to the `POLAR` applet variable R3.
```
0560F2    ~xR4                       ( → id )
```
The id corresponding to the `POLAR` applet variable R4.

```
0580F2    ~xR5                        ( → id )
```
The id corresponding to the `POLAR` applet variable R5.
```
05A0F2    ~xR6                        ( → id )
```
The id corresponding to the `POLAR` applet variable R6.
```
05C0F2    ~xR7                        ( → id )
```
The id corresponding to the `POLAR` applet variable R7.
```
05E0F2    ~xR8                        ( → id )
```
The id corresponding to the `POLAR` applet variable R8.
```
0600F2    ~xR9                        ( → id )
```
The id corresponding to the `POLAR` applet variable R9.
```
0630F2    ~idR0                       ( → id )
```
The id corresponding to the `POLAR` applet variable R0.
```
0510F2    ~idR1                       ( → id )
```
The id corresponding to the `POLAR` applet variable R1.
```
0530F2    ~idR2                       ( → id )
```
The id corresponding to the `POLAR` applet variable R2.
```
0550F2    ~idR3                       ( → id )
```
The id corresponding to the `POLAR` applet variable R3.
```
0570F2    ~idR4                       ( → id )
```
The id corresponding to the `POLAR` applet variable R4.
```
0590F2    ~idR5                       ( → id )
```
The id corresponding to the `POLAR` applet variable R5.
```
05B0F2    ~idR6                       ( → id )
```
The id corresponding to the `POLAR` applet variable R6.
```
05D0F2    ~idR7                       ( → id )
```
The id corresponding to the `POLAR` applet variable R7.
```
05F0F2    ~idR8                       ( → id )
```
The id corresponding to the `POLAR` applet variable R8.
```
0610F2    ~idR9                       ( → id )
```
The id corresponding to the `POLAR` applet variable R9.

5.2.9 SEQUENCE applet

```
0120F2    ~xU0                    ( → id )
```
The id corresponding to the SEQUENCE applet variable U0.
```
0000F2    ~xU1                    ( → id )
```
The id corresponding to the SEQUENCE applet variable U1.
```
0020F2    ~xU2                    ( → id )
```
The id corresponding to the SEQUENCE applet variable U2.
```
0040F2    ~xU3                    ( → id )
```
The id corresponding to the SEQUENCE applet variable U3.
```
0060F2    ~xU4                    ( → id )
```
The id corresponding to the SEQUENCE applet variable U4.
```
0080F2    ~xU5                    ( → id )
```
The id corresponding to the SEQUENCE applet variable U5.
```
00A0F2    ~xU6                    ( → id )
```
The id corresponding to the SEQUENCE applet variable U6.
```
00C0F2    ~xU7                    ( → id )
```
The id corresponding to the SEQUENCE applet variable U7.
```
00E0F2    ~xU8                    ( → id )
```
The id corresponding to the SEQUENCE applet variable U8.
```
0100F2    ~xU9                    ( → id )
```
The id corresponding to the SEQUENCE applet variable U9.
```
0130F2    ~idU0                   ( → id )
```
The id corresponding to the SEQUENCE applet variable U0.
```
0010F2    ~idU1                   ( → id )
```
The id corresponding to the SEQUENCE applet variable U1.
```
0030F2    ~idU2                   ( → id )
```
The id corresponding to the SEQUENCE applet variable U2.
```
0050F2    ~idU3                   ( → id )
```
The id corresponding to the SEQUENCE applet variable U3.
```
0070F2    ~idU4                   ( → id )
```
The id corresponding to the SEQUENCE applet variable U4.
```
0090F2    ~idU5                   ( → id )
```
The id corresponding to the SEQUENCE applet variable U5.

00B0F2	~idU6	(→ id)
		The id corresponding to the SEQUENCE applet variable U6.
00D0F2	~idU7	(→ id)
		The id corresponding to the SEQUENCE applet variable U7.
00F0F2	~idU8	(→ id)
		The id corresponding to the SEQUENCE applet variable U8.
0110F2	~idU9	(→ id)
		The id corresponding to the SEQUENCE applet variable U9.

5.2.10 SOLVE applet

1100F2	~idE0	(→ id)
		The id corresponding to the SOLVE applet variable E0.
1070F2	~idE1	(→ id)
		The id corresponding to the SOLVE applet variable E1.
1080F2	~idE2	(→ id)
		The id corresponding to the SOLVE applet variable E2.
1090F2	~idE3	(→ id)
		The id corresponding to the SOLVE applet variable E3.
10A0F2	~idE4	(→ id)
		The id corresponding to the SOLVE applet variable E4.
10B0F2	~idE5	(→ id)
		The id corresponding to the SOLVE applet variable E5.
10C0F2	~idE6	(→ id)
		The id corresponding to the SOLVE applet variable E6.
10D0F2	~idE7	(→ id)
		The id corresponding to the SOLVE applet variable E7.
10E0F2	~idE8	(→ id)
		The id corresponding to the SOLVE applet variable E8.
10F0F2	~idE9	(→ id)
		The id corresponding to the SOLVE applet variable E9.

5.2.11 STATISTICS applet

```
2B9AB      xC0                        ( → id )
                                      The id corresponding to the STATISTICS applet
                                      variable C0.
2B8DC      xC1                        ( → id )
                                      The id corresponding to the STATISTICS applet
                                      variable C1.
2B8F3      xC2                        ( → id )
                                      The id corresponding to the STATISTICS applet
                                      variable C2.
2B90A      xC3                        ( → id )
                                      The id corresponding to the STATISTICS applet
                                      variable C3.
2B921      xC4                        ( → id )
                                      The id corresponding to the STATISTICS applet
                                      variable C4.
2B938      xC5                        ( → id )
                                      The id corresponding to the STATISTICS applet
                                      variable C5.
2B94F      xC6                        ( → id )
                                      The id corresponding to the STATISTICS applet
                                      variable C6.
2B966      xC7                        ( → id )
                                      The id corresponding to the STATISTICS applet
                                      variable C7.
2B97D      xC8                        ( → id )
                                      The id corresponding to the STATISTICS applet
                                      variable C8.
2B994      xC9                        ( → id )
                                      The id corresponding to the STATISTICS applet
                                      variable C9.
```

5.2.12 Unknown

```
0780F2     ~xs1
0790F2     ~xs2
07A0F2     ~xs3
07B0F2     ~xs4
07C0F2     ~xs5
07D0F2     ~xn1
07E0F2     ~xn2
07F0F2     ~xn3
0800F2     ~xn4
```

```
0810F2    ~xn5
06D0F2    ~xD0
0640F2    ~xD1
0650F2    ~xD2
0660F2    ~xD3
0670F2    ~xD4
0680F2    ~xD5
0690F2    ~xD6
06A0F2    ~xD7
06B0F2    ~xD8
06C0F2    ~xD9
0860F2    ~xE
06E0F2    ~xH1
06F0F2    ~xH2
0700F2    ~xH3
0710F2    ~xH4
0720F2    ~xH5
08A0F2    ~xI
0A70F2    ~xQ1
0A80F2    ~xQ3
0730F2    ~xS1
1290F2    ~xS1fit
1240F2    ~xS1mark
0740F2    ~xS2
12A0F2    ~xS2fit
1250F2    ~xS2mark
0750F2    ~xS3
12B0F2    ~xS3fit
1260F2    ~xS3mark
0760F2    ~xS4
12C0F2    ~xS4fit
1270F2    ~xS4mark
0770F2    ~xS5
12D0F2    ~xS5fit
1280F2    ~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.

```
00B0A1    ~NSKey3.2              ???
                                 The unshifted key 3.2.
00C0A1    ~NSKey3.3              ???
                                 The unshifted key 3.3.
0100A1    ~NSKey4.1              ???
                                 The unshifted key 4.1.
0110A1    ~NSKey4.2              ???
                                 The unshifted key 4.2.
0120A1    ~NSKey4.3              ???
                                 The unshifted key 4.3.
0130A1    ~NSKey4.4              ???
                                 The unshifted key 4.4.
0140A1    ~NSKey4.5              ???
                                 The unshifted key 4.5.
0160A1    ~NSKey5.2              ???
                                 The unshifted key 5.2.
0170A1    ~NSKey5.3              ???
                                 The unshifted key 5.3.
0180A1    ~NSKey5.4              ???
                                 The unshifted key 5.4.
0190A1    ~NSKey5.5              ???
                                 The unshifted key 5.5.
01A0A1    ~NSKey6.1              ???
                                 The unshifted key 6.1.
01B0A1    ~NSKey6.2              ???
                                 The unshifted key 6.2.
```

```
01C0A1    ~NSKey6.3              ???
                                 The unshifted key 6.3.
01D0A1    ~NSKey6.4              ???
                                 The unshifted key 6.4.
01E0A1    ~NSKey6.5              ???
                                 The unshifted key 6.5.
01F0A1    ~NSKey7.1              ???
                                 The unshifted key 7.1.
0200A1    ~NSKey7.2              ???
                                 The unshifted key 7.2.
0210A1    ~NSKey7.3              ???
                                 The unshifted key 7.3.
0220A1    ~NSKey7.4              ???
                                 The unshifted key 7.4.
0230A1    ~NSKey7.5              ???
                                 The unshifted key 7.5.
0240A1    ~NSKey8.1              ???
                                 The unshifted key 8.1.
0250A1    ~NSKey8.2              ???
                                 The unshifted key 8.2.
0260A1    ~NSKey8.3              ???
                                 The unshifted key 8.3.
0270A1    ~NSKey8.4              ???
                                 The unshifted key 8.4.
0280A1    ~NSKey8.5              ???
                                 The unshifted key 8.5.
02A0A1    ~NSKey9.2              ???
                                 The unshifted key 9.2.
02B0A1    ~NSKey9.3              ???
                                 The unshifted key 9.3.
02C0A1    ~NSKey9.4              ???
                                 The unshifted key 9.4.
02D0A1    ~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.
00A0A2    ~LSKey3.1              ???
                                 The shifted key 3.1.
00B0A2    ~LSKey3.2              ???
                                 The shifted key 3.2.
00C0A2    ~LSKey3.3              ???
                                 The shifted key 3.3.
0100A2    ~LSKey4.1              ???
                                 The shifted key 4.1.
0110A2    ~LSKey4.2              ???
                                 The shifted key 4.2.
0120A2    ~LSKey4.3              ???
                                 The shifted key 4.3.
0130A2    ~LSKey4.4              ???
                                 The shifted key 4.4.
0140A2    ~LSKey4.5              ???
                                 The shifted key 4.5.
0150A2    ~LSKey5.1              ???
                                 The shifted key 5.1.
0160A2    ~LSKey5.2              ???
                                 The shifted key 5.2.
0180A2    ~LSKey5.4              ???
                                 The shifted key 5.4.
0190A2    ~LSKey5.5              ???
                                 The shifted key 5.5.
01A0A2    ~LSKey6.1              ???
                                 The shifted key 6.1.
01B0A2    ~LSKey6.2              ???
                                 The shifted key 6.2.
01C0A2    ~LSKey6.3              ???
                                 The shifted key 6.3.
01D0A2    ~LSKey6.4              ???
                                 The shifted key 6.4.
01E0A2    ~LSKey6.5              ???
                                 The shifted key 6.5.
01F0A2    ~LSKey7.1              ???
                                 The shifted key 7.1.
0200A2    ~LSKey7.2              ???
                                 The shifted key 7.2.
0210A2    ~LSKey7.3              ???
                                 The shifted key 7.3.
```

```
0220A2    ~LSKey7.4              ???
                                 The shifted key 7.4.
0230A2    ~LSKey7.5              ???
                                 The shifted key 7.5.
0240A2    ~LSKey8.1              ???
                                 The shifted key 8.1.
0250A2    ~LSKey8.2              ???
                                 The shifted key 8.2.
0260A2    ~LSKey8.3              ???
                                 The shifted key 8.3.
0270A2    ~LSKey8.4              ???
                                 The shifted key 8.4.
0280A2    ~LSKey8.5              ???
                                 The shifted key 8.5.
0290A2    ~LSKey9.1              ???
                                 The shifted key 9.1.
02A0A2    ~LSKey9.2              ???
                                 The shifted key 9.2.
02B0A2    ~LSKey9.3              ???
                                 The shifted key 9.3.
02C0A2    ~LSKey9.4              ???
                                 The shifted key 9.4.
02D0A2    ~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    ~ANSKey3.1             ???
                                 The key 3.1 in alpha mode.
```

```
00B0A4    ~ANSKey3.2              ???
                                  The key 3.2 in alpha mode.
00C0A4    ~ANSKey3.3              ???
                                  The key 3.3 in alpha mode.
00D0A4    ~ANSKey3.4              ???
                                  The key 3.4 in alpha mode.
00E0A4    ~ANSKey3.5              ???
                                  The key 3.5 in alpha mode.
00F0A4    ~ANSKey3.6              ???
                                  The key 3.6 in alpha mode.
0100A4    ~ANSKey4.1              ???
                                  The key 4.1 in alpha mode.
0110A4    ~ANSKey4.2              ???
                                  The key 4.2 in alpha mode.
0120A4    ~ANSKey4.3              ???
                                  The key 4.3 in alpha mode.
0130A4    ~ANSKey4.4              ???
                                  The key 4.4 in alpha mode.
0140A4    ~ANSKey4.5              ???
                                  The key 4.5 in alpha mode.
0150A4    ~ANSKey5.1              ???
                                  The key 5.1 in alpha mode.
0160A4    ~ANSKey5.2              ???
                                  The key 5.2 in alpha mode.
0170A4    ~ANSKey5.3              ???
                                  The key 5.3 in alpha mode.
0180A4    ~ANSKey5.4              ???
                                  The key 5.4 in alpha mode.
0190A4    ~ANSKey5.5              ???
                                  The key 5.5 in alpha mode.
01A0A4    ~ANSKey6.1              ???
                                  The key 6.1 in alpha mode.
01B0A4    ~ANSKey6.2              ???
                                  The key 6.2 in alpha mode.
01C0A4    ~ANSKey6.3              ???
                                  The key 6.3 in alpha mode.
01D0A4    ~ANSKey6.4              ???
                                  The key 6.4 in alpha mode.
01E0A4    ~ANSKey6.5              ???
                                  The key 6.5 in alpha mode.
01F0A4    ~ANSKey7.1              ???
                                  The key 7.1 in alpha mode.
0200A4    ~ANSKey7.2              ???
                                  The key 7.2 in alpha mode.
0210A4    ~ANSKey7.3              ???
                                  The key 7.3 in alpha mode.
```

```
0220A4    ~ANSKey7.4            ???
                                The key 7.4 in alpha mode.
0230A4    ~ANSKey7.5            ???
                                The key 7.5 in alpha mode.
0240A4    ~ANSKey8.1            ???
                                The key 8.1 in alpha mode.
0250A4    ~ANSKey8.2            ???
                                The key 8.2 in alpha mode.
0260A4    ~ANSKey8.3            ???
                                The key 8.3 in alpha mode.
0270A4    ~ANSKey8.4            ???
                                The key 8.4 in alpha mode.
0280A4    ~ANSKey8.5            ???
                                The key 8.5 in alpha mode.
0290A4    ~ANSKey9.1            ???
                                The key 9.1 in alpha mode.
02A0A4    ~ANSKey9.2            ???
                                The key 9.2 in alpha mode.
02B0A4    ~ANSKey9.3            ???
                                The key 9.3 in alpha mode.
02C0A4    ~ANSKey9.4            ???
                                The key 9.4 in alpha mode.
02D0A4    ~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    ~ALSKey3.1            ???
                                The shifted 3.1 key in alpha mode.
```

00B0A5	~ALSKey3.2	???
		The shifted 3.2 key in alpha mode.
00C0A5	~ALSKey3.3	???
		The shifted 3.3 key in alpha mode.
00D0A5	~ALSKey3.4	???
		The shifted 3.4 key in alpha mode.
00E0A5	~ALSKey3.5	???
		The shifted 3.5 key in alpha mode.
00F0A5	~ALSKey3.6	???
		The shifted 3.6 key in alpha mode.
0100A5	~ALSKey4.1	???
		The shifted 4.1 key in alpha mode.
0110A5	~ALSKey4.2	???
		The shifted 4.2 key in alpha mode.
0120A5	~ALSKey4.3	???
		The shifted 4.3 key in alpha mode.
0130A5	~ALSKey4.4	???
		The shifted 4.4 key in alpha mode.
0140A5	~ALSKey4.5	???
		The shifted 4.5 key in alpha mode.
0150A5	~ALSKey5.1	???
		The shifted 5.1 key in alpha mode.
0160A5	~ALSKey5.2	???
		The shifted 5.2 key in alpha mode.
0170A5	~ALSKey5.3	???
		The shifted 5.3 key in alpha mode.
0180A5	~ALSKey5.4	???
		The shifted 5.4 key in alpha mode.
0190A5	~ALSKey5.5	???
		The shifted 5.5 key in alpha mode.
01A0A5	~ALSKey6.1	???
		The shifted 6.1 key in alpha mode.
01B0A5	~ALSKey6.2	???
		The shifted 6.2 key in alpha mode.
01C0A5	~ALSKey6.3	???
		The shifted 6.3 key in alpha mode.
01D0A5	~ALSKey6.4	???
		The shifted 6.4 key in alpha mode.
01E0A5	~ALSKey6.5	???
		The shifted 6.5 key in alpha mode.
01F0A5	~ALSKey7.1	???
		The shifted 7.1 key in alpha mode.
0200A5	~ALSKey7.2	???
		The shifted 7.2 key in alpha mode.
0210A5	~ALSKey7.3	???
		The shifted 7.3 key in alpha mode.

0220A5	~ALSKey7.4	???
		The shifted 7.4 key in alpha mode.
0230A5	~ALSKey7.5	???
		The shifted 7.5 key in alpha mode.
0240A5	~ALSKey8.1	???
		The shifted 8.1 key in alpha mode.
0250A5	~ALSKey8.2	???
		The shifted 8.2 key in alpha mode.
0260A5	~ALSKey8.3	???
		The shifted 8.3 key in alpha mode.
0270A5	~ALSKey8.4	???
		The shifted 8.4 key in alpha mode.
0280A5	~ALSKey8.5	???
		The shifted 8.5 key in alpha mode.
0290A5	~ALSKey9.1	???
		The shifted 9.1 key in alpha mode.
02A0A5	~ALSKey9.2	???
		The shifted 9.2 key in alpha mode.
02B0A5	~ALSKey9.3	???
		The shifted 9.3 key in alpha mode.
02C0A5	~ALSKey9.4	???
		The shifted 9.4 key in alpha mode.
02D0A5	~ALSKey9.5	???
		The shifted 9.5 key in alpha mode.

5.5 Labels

38641	INT_00
38646	INT_01
3864B	INT_02
38650	INT_03
38655	INT_04
3865A	INT_05
3865F	INT_06
38664	INT_07
38669	INT_08
3866E	INT_09
38673	INT_0A
38678	INT_0B
3867D	INT_0C
38682	INT_0D
38687	INT_0E

```
3868C       INT_0F
38691       INT_10
38696       INT_11
3869B       INT_12
386A0       INT_13
386A5       INT_14
386AA       INT_15
386AF       INT_16
386B4       INT_17
386B9       INT_18
386BE       INT_19
386C3       INT_1A
386C8       INT_1B
386CD       INT_1C
386D2       INT_1D
386D7       INT_1E
386DC       INT_1F
385D4       INT_NN
```

5.6 LastBut

```
28918       LastBut0
2891F       LastBut1
28926       LastBut2
2892D       LastBut3
28934       LastBut4
2893B       LastBut5
28942       LastBut6
28949       LastBut7
28950       LastBut8
28957       LastBut9
2895E       LastBut10
28965       LastBut11
2896C       LastBut12
28973       LastBut13
2897A       LastBut14
28981       LastBut15
28988       LastBut16
```

```
2898F      LastBut17
28996      LastBut18
2899D      LastBut19
289A4      LastBut20
289AB      LastBut21
289B2      LastBut22
289B9      LastBut23
289C0      LastBut24
289C7      LastBut25
289CE      LastBut26
289D5      LastBut27
289DC      LastBut28
289E3      LastBut29
289EA      LastBut30
289F1      LastBut31
289F8      LastBut32
```

5.7 x

```
3A648      x->NUM
3C685      x<?
3C7B9      x=?
3C729      x>?
0E40F2     ~xAngle
0DB0F2     ~xAns
1140F2     ~xArea
0E50F2     ~xAxes
3A9B8       xCHS
0E70F2     ~xConnect
0EE0F2     ~xCoord
0E30F2     ~xDate
1030F2     ~xDigits
3AB39       xEPS
3A3D2       xEXIT
3BD5B       xEXPM1
1130F2     ~xExtremum
0DC0F2     ~xFIT
058002     ~xFmList
```

```
059002    ~xFmMat
1020F2    ~xFormat
0E60F2    ~xGrid
0DD0F2    ~xHAngle
0DF0F2    ~xHDigits
0DE0F2    ~xHFormat
0F60F2    ~xHTick
0E80F2    ~xHighRes
1210F2    ~xHisWidth
1230F2    ~xHmax
1220F2    ~xHmin
0F80F2    ~xHzoom
3AB17      xINF
3B4C3      xINVERSE
0E10F2    ~xIerr
0F30F2    ~xIndep
0EB0F2    ~xInvCursor
1120F2    ~xIsect
0EC0F2    ~xLabels
0A50F2    ~xMAXS
09D0F2    ~xMEANS
0A90F2    ~xMEANX
0AC0F2    ~xMEANY
0A60F2    ~xMEDIAN
0A40F2    ~xMINS
0A30F2    ~xNS
11D0F2    ~xNmax
11C0F2    ~xNmin
1040F2    ~xNoteText
1000F2    ~xNumCol
1010F2    ~xNumFont
0FD0F2    ~xNumIndep
0FF0F2    ~xNumRow
0FA0F2    ~xNumStart
0FB0F2    ~xNumStep
0FC0F2    ~xNumType
0FE0F2    ~xNumZoom
3C935      xP\8DR
```

```
1050F2    ~xPage
1060F2    ~xPageNum
0B20F2    ~xRELERR
3C885      xR\8DP
0E00F2    ~xRadixMark
0EA0F2    ~xRecenter
1110F2    ~xRoot
0A10F2    ~xSSDEV
09F0F2    ~xSVARS
0AA0F2    ~xSX
0AB0F2    ~xSX2
0AF0F2    ~xSXY
0AD0F2    ~xSY
0AE0F2    ~xSY2
11E0F2    ~xSeqPlot
0E90F2    ~xSimult
1150F2    ~xSlope
11F0F2    ~xStatMode
1200F2    ~xStatPlot
09E0F2    ~xTOTS
1180F2    ~xTStep
11A0F2    ~xThetaMax
1190F2    ~xThetaMin
11B0F2    ~xThetaStep
0E20F2    ~xTime
1170F2    ~xTmax
1160F2    ~xTmin
0ED0F2    ~xTracing
0F70F2    ~xVTick
0F90F2    ~xVzoom
0F40F2    ~xXcross
0F00F2    ~xXmax
0EF0F2    ~xXmin
0F50F2    ~xYcross
0F20F2    ~xYmax
0F10F2    ~xYmin
3CAD2      xint
03F0AB    ~x>COL
```

```
0410AB    ~x>DIAG
061004    ~x>DISPLAY
063004    ~x>PLOT
03D0AB    ~x>ROW
0350AB    ~xACOT
0360AB    ~xACSC
0370AB    ~xASEC
064004    ~xBLANKGROB
0250AB    ~xBOXW
03D701    ~xCHECK
0230AB    ~xCOBWEB
0400AB    ~xCOL>
0170AB    ~xCONCAT
0320AB    ~xCOT
0340AB    ~xCSC
0480AB    ~xCSWP
02B0AB    ~xCUBICFIT
0320F1    ~xDFLTNOTE
0330F1    ~xDFLTPICT
0420AB    ~xDIAG>
060004    ~xDISPLAY>
050004    ~xERASEA
0210AB    ~xFRACTION
02B0F1    ~xFUNCSYMB
0080F1    ~xFUNCTAB
0240AB    ~xHIST
02C0AB    ~xLOGISFIT
0310AB    ~xMKMAT
02D0F1    ~xPARAMSYMB
00A0F1    ~xPARAMTAB
062004    ~xPLOT>
02C0F1    ~xPOLARSYMB
0090F1    ~xPOLARTAB
0390AB    ~xPOLYFORM
0290AB    ~xPOWERFIT
05A0AB    ~xPoly
02A0AB    ~xQUADFIT
03C0F1    ~xREADNOTE
```

```
03D0F1    ~xREADPICT
0380AB    ~xRECURSE
03E0AB    ~xROW>
0330AB    ~xSEC
02E0F1    ~xSEQSYMB
00B0F1    ~xSEQTAB
0310F1    ~xSOLVESYMB
00E0F1    ~xSOLVETAB
0220AB    ~xSTAIRSTEP
0300F1    ~xSTAT2SYMB
00D0F1    ~xSTAT2TAB
02F0F1    ~xSTATSYMB
00C0F1    ~xSTATTAB
02E0AB    ~xStat1Var
02F0AB    ~xStat2Var
03E701    ~xUNCHECK
02D0AB    ~xUSERFIT
01E0F1    ~xUndefined
0150AB    ~x\85LIST
0140AB    ~x\9BLIST
0160AB    ~x\9CLIST
014701    ~xx>DISPLAY
017701    ~xx>GROB
016701    ~xx>PLOT
018701    ~xxARC
003701    ~xxBEEP
00F701    ~xxBOX
01D701    ~xxBREAK
041701    ~xxCHOOSE
023701    ~xxCOL+
022701    ~xxCOL-
025701    ~xxCSWP
01A0AB    ~xxDEG
038701    ~xxDEMO
002701    ~xxDISP
013701    ~xxDISPLAY>
031701    ~xxDO1VSTATS
034701    ~xxDO2VSTATS
```

```
009701    ~xxERASE
00A701    ~xxERASEPLOT
01F701    ~xxFREEZE
042701    ~xxGETKEY
010701    ~xxGOR
01C0AB    ~xxGRAD
012701    ~xxGROBNOT
011701    ~xxGXOR
043701    ~xxHELP
01C701    ~xxINPUT
0300AB    ~xxITERATE
02D701    ~xxLIBEVAL
00D701    ~xxLINE
02A701    ~xxMATEDIT
039701    ~xxMEM
02B701    ~xxMKGROB
040701    ~xxMSGBOX
00C701    ~xxPIXOFF
00B701    ~xxPIXON
015701    ~xxPLOT>
001004    ~xxPOS
01B701    ~xxPRDISPLAY
019701    ~xxPRSTC
01A701    ~xxPRVAR
01B0AB    ~xxRAD
036701    ~xxRANM
026701    ~xxRCI
027701    ~xxRCIJ
008701    ~xxRDM
005701    ~xxRDZ
007701    ~xxREPL
021701    ~xxROW+
020701    ~xxROW-
024701    ~xxRSWP
03A701    ~xxRULES
029701    ~xxRUNPGM
035701    ~xxSELECT
033701    ~xxSETDEPEND
```

```
030701    ~xxSETFREQ
032701    ~xxSETINDEP
02F701    ~xxSETSAMPLE
03C701    ~xxSETVIEWS
000004    ~xxSIZE
01E701    ~xxSTOP
006701    ~xxSUB
02E701    ~xxSYSEVAL
00E701    ~xxTLINE
028701    ~xxTO
001701    ~xxTSTR
037701    ~xxVERSION
004701    ~xxWAIT
000701    ~xxWSLOG
02C701    ~xxZEROGROB
```

5.8 Rest

```
13AF2     !>ARRAY
3F804     #+#2-
15B08     #=Lookup
3709C     #DIV
11C67     #Error:
370C3     #MOD
37087     #NEG
03EB1     #NOT
37116     #ODD
03ED0     #OR
11C21     #ObTypeBase
11C5D     #Warning:NL
03EF2     #XOR
1663B     %%-1
3F07E     &&
1089E     1GETLAM#0=
3F28F     1GETapndcpl
3F5F2     2CDispList
3F5F3     2CKeyOK
14F3E     2Col?Case2Col
```

3F5A9	2ColChoose
04247	2STR
3F5F4	?AdjFocusPos
3F5F5	?DispMoreU/D
3F5F6	?FixFieldKeys
3F5F7	?GetFObTypes
24067	?NoTaskSwDef
13236	ABSCOERCE
042DD	ARRYCMP
042FB	ARRYLIST
0432D	ARRYSYM
3F5F8	AddEq$
2ECD8	AlDrawMenu
3F290	AllowPRLCD
27083	Angle@
3F5F9	AngleField
3F5FA	AngleLabel
363C9	AnyDABad?
3F5FB	ApName$>Id
3F5FC	ApNameId>$
3F5FD	ApNameId>Id
28879	AppDir+Offs
2738F	Area@
0ED6E	AttnPOSCOMP
09FB0	BAK>HOME
3F292	BREAK
271EB	BTRow1@
23460	BadIfEdit
38763	BinLookup
3F5FE	BlankClient
3F630	BlankHelp
3F6F1	Bubble
386E1	CAND
0260F1	~CLRTOPICLAM
08530	COMPROMID
3870B	COR
28862	CalcDir+Offs
28154	CatNot

```
281C7      CatPgm
2815E      CatThisCxt
3F603      Choose&DoTask
3F5CA      ChooseVEntry
3F5CB      ChooseVExit
3F5CC      ChooseViewUI
28890      Contxt+Offs
2E5CB      CopyRegCOb
10A07      DOINDIR
3F3FD      DOLCD>g0
3F78F      DOROOT
3F60F      DOSHOWIT1
192D4      DOSHOWIT1C
3A7C7      DOWAIT
3F610      DecompNoNL
3F5E7      DecompOb
2EC6F      DefauPtXit
2F763      DefauTrcInit
36A17      DefaultHint
3F8D0      DemoFrames
27213      Digits@
3F349      DispMsgBox
30473      DispXFunc
2FCDB      DispYFunc
3F34C      DoAlert&Query
3F611      DoApletLib
3F401      DoAs2Col
3F34E      DoCApNoteV
3F34F      DoCApPlotSV
3F612      DoCApPlotV
3F350      DoCApSketchV
3F351      DoCApSymbSV
3F613      DoCApSymbV
3F352      DoCApTableSV
3F614      DoCApTableV
3F615      DoCurrAplet
3F8D1      DoDemo
3F616      DoDispBorder
```

3F5C1	DoDispField	
3F617	DoDispList	
3F618	DoDispPrompt	
3F6FD	DoEditLCancel	
3F6FF	DoEditLOK	
3F701	DoEditLine	
3F294	DoGetObFrSto	
3F295	DoIOErrAlert	
3F296	DoIOStatusBox	
17236	DoInApLbCxt	
28037	DoInNotCxt	
27B91	DoInPgmCxt	
3F619	DoKeyCheck	
3F61B	DoKeyChoos/Ck	
3F61A	DoKeyChoose	
3F702	DoMatEdit	
3F363	DoQueryBox	

3F02B   DoRecv/GetOb   (id →)

Receive an object. FIXME: Stack diagram not confirmed.

3F297	DoRecvObFrEls	
3F6B2	DoRestCovWin	
3F80D	DoResultTab	
24376	DoRomPtrKey	
3F6B1	DoSaveCovWin	

3F02C   DoSendOb   (ob →)

Send an object. FIXME: Stack diagram not confirmed.

3F298	DoSendObToCDi	
3F299	DoSendObToDir	
3F2D8	DoSendObToEls	
3F29A	DoSendObToSto	
3F61E	DoSpecAlert	
1924D	Docrunchc	
16C90	Done?CkNoNul:	
352C0	DropJunk	
38304	DummyMenuErr	
28A40	DummyVar	
3F29B	ElsieGet	

```
3F29C      ElsiePkt
3F29D      ElsieSend
3F36A      EmptyList?
3F61F      EmptyRList?
38381      EnsureMenuOff
380FE      EnterGraphView
380CC      EnterTextView
3F5F0      EraseGraph
3F620      EraseGrob
11C0D      Err#Chr00
17CC5      EvalPart1
17CD9      EvalPart2
17CED      EvalPart3
273A3      Extremum@
3F631      FAreaBad?
3F08F      FOURpsh
04A49      FSTFLOATROM#
3F623      FTypeCheck?
3F622      FTypeFixedL?
3F624      FTypeList?
3F625      FTypeText?
3F626      FUNCPLOT
3F36F      FitLeftSmF
3F627      FitRightSmF
271FF      Format@
3F628      FuncPLoop
3F629      FuncPSetup1
16D7E      FuncSplitViewers
1F4C2      GDISP?
3F633      GROBBUFF!
3F634      GROBVIEW
3F5AB      GetApDirList
3F29F      GetBVars.1
3F635      GetChoiceFmt
3F636      GetChoiceList
3F375      GetDASpecFlag
15C02      GetElNoRomp
3F410      GetLastNotThis
```

```
3F637      GetNextFId
3F2A1      GetObType$
3F2A0      GetObTypePr$
17C89      GetPart1
17C9D      GetPart2
17CB1      GetPart3
3F638      GetPrevFId
3F5EC      GraphTableUI
3F5ED      GraphZoomUI
3F639      GtoField
09DBB      HOME>BAK
27033      HTick@
2705B      HZoom@
2770E      HisWidth@
27736      Hmax@
27722      Hmin@
3F63A      IFECheck
3F63B      IFEChoosByChr
3F63C      IFEChooseNext
3F63D      IFEDispClient
3F63F      IFEDispHelp
3F640      IFEDispLabel
3F641      IFEDispTitle
3F642      IFEUnShowSel
3F643      IFEdLineMenu
3F644      IFMenu
3F6AA      IFSymbViewUI
3F645      Id>ApNameId
3F383      InAplet?
3F646      InApletF?
3F647      Init2ColMets
3F417      InitIndep
3F5AC      InitListMets
3F648      Init_window
3F5C9      InpFormVEntry
3F5C4      InpFormVExit
3F5C5      InpFormViewUI
3F649      InvertField
```

```
3F64A      IsIFMenu?
3F64B      IsNullField?
245E2      IsTaskSwKey?
2737B      Isect@
36F79      KeyFace
28A1A      LASTBUTN
3F650      LEDispBorder
3F654      LESetIDecomp
3F384      LESetItem
3F655      LESetRowWidth
124C4      LF$
3F656      LHighlight
11D61      LSTBIMACROM#
3F658      LUnHighlight
270D3      LastEq@
270BF      LastIndep@
27097      LastX@
270AB      LastY@
380E5      LeaveGraphView
380E5      LeaveTextView
3F659      ListBar
3F65A      ListNames
273DA      ListOfEqs@
3F2A2      M+prep
3F65C      Make1stAplet
3F393      MakeNoteVTtl
3F65D      MakeNumSVTtl
3F65B      MakeNumVTtl
3F65E      MakePlotSVTtl
3F65F      MakePlotVTtl
3F660      MakeSketVTtl
3F661      MakeSymbSVTtl
3F662      MakeSymbVTtl
3F394      MakeTitleBar
3F663      MakeViewTitle
3F712      MemStoAns
38345      MiniVar
3F664      MoreDown?
```

```
3F665      MoreU/D?
3F666      MoreUp?
16EE3      NULLargcase
13C06      NUMEVAL1
3F667      NextApOfType
3F668      NextTextLine
38C6B      NotAndbitmap
3F6A9      NotesViewUI
3F2A3      OB>BAK
0D4FC      OVEREQcase
38C45      Orbitmap
35725      POLSetUIExt
09C3C      PORT0ADDR
15573      PageDnNGrob
1552F      PageDnYGrob
154EB      PageUpNGrob
154A7      PageUpYGrob
24E34      ParseExpr
3F41D      PickedEqs@
3F6B0      PictViewUI
3F669      PlaceFVals
270FB      Plot2Flg@
2710F      PlotCKSUM@
270E7      PlotFlag@
3F62A      PlotOvViewUI
3F6AB      PlotSetViewUI
3F66A      PlotViewUI
3F2AA      Port0Ids
3F66B      PrepCurrAplet
3F3AA      PrepEdLKeyOb
3F2A7      PrintHist
3F2A8      PrintLcd
3F2A9      PrintVar
3F0C4      PrintVarOb
3F66D      PurgeAplet
3F66C      PurgeApletOb
3F20A      Put3x5
38C7E      Put5x7
```

```
3F3AB      PutDASpecFlag
38BC8      Putbitmap
11CD5      REALLISTREAL
11C85      REALSTRID
3F80E      REDIMPREP
2094F      RPICK
3F3B7      RclAplet
3F66E      RclAplet?Err
3F66F      RclFieldVal
3F670      RclResetVal
27786      ReCalcFlag@
3F671      Replace_List
27353      Root@
3F2AB      SEFINISH
3F3BB      SETPLOTENV
16C54      STArryList
17D47      STBigKeyDef
3F805      STDoMedium
3F806      STDoSmall
17D98      STEditKeyDef
3F735      STErrorGrob
3F807      STInitCols
17DC8      STInsKeyDef
3F808      STJump
3F809      STJumpN
26F75      STO_tTYPE
3F80A      STSortCol
17DF6      STStatsKeyDef
3F736      STTableDisp
3F737      STTableExit
3F738      STTableHKeys
3F739      STTableInit
10D10      STypeAnyChr
10CCA      STypeDirChr
10CFB      STypeLibChr
10CED      STypeLstChr
10CF4      STypeMatChr
10CD8      STypeNteChr
```

```
10CDF      STypePgmChr
10D02      STypeTgtChr
10CE6      STypeVarChr
10D09      STypeVecChr
10CD1      STypeZapChr
3F674      SV_?defined
3F675      SV_actual
3F676      SV_calledit
3F677      SV_getbody
16F3F      SV_keycancel
16F67      SV_keycheck
16F7B      SV_keyedit
16F9C      SV_keyeval
16F53      SV_keyok
16FBD      SV_keyshow
3F67B      SV_setgrob19
3F67C      SV_setitem0
276E6      SVarType@
3F73B      SWAPStatFlag
3F73B      SWAPStatFlags
046A7      SYMSTR
3F3C1      SafeCrunch%
3F3C2      SafeDecomp
3F67D      SaveListMets
3F67E      Sel&DispNextF
3F67F      SelNextField
3F680      SelPrevField
3F3C4      SelectAplet
3F6AC      SeqTableViewUI
3F5E8      Seq_eval0
3F5E9      Seq_eval1
3F5EA      Seq_eval2
3F5EB      Seq_evaln
36388      SetDA2Temp
3F3C5      SetDAsBad
3F5C6      SetFAreaBad
3F3C6      SetFAreaOK
3F681      SetFAreasBad
```

```
3F682      SetFAreasOK
3F683      SetInAplet
3F3C9      SetRedrawFlag
3F684      Set_window
27367      Slope@
3F6AE      SolveNumViewUI
27123      Split1CKSUM@
27137      Split2CKSUM@
271D7      SplitRow@
2F8BC      SplitTraceInit
27772      Stat2Flag@
2783A      StatFit@
2779A      StatFont@
2774A      StatMark@
277A4      StatMisc[]@
2775E      StatModel@
276FA      StatPlot@
3F6AD      StatTableViewUI
276CD      StatType!
276D2      StatType@
0230F1     ~StdApEntry
3F686      StdDecompNoNL
3F2AE      StdIOProc
3F685      StdTableViewUI
3F687      StndXYCoord?
156DA      StoAns@
3F741      StoAns@Drp
3F3CB      StoAplet
3F3CE      StripTicks
3F5AD      SwapL/RMets
3F6AF      SymbSetViewUI
3F688      SymbViewUI
3F80F      TBColL3x5
3F810      TBColL5x7
3F811      TBColR3x5
3F812      TBColR5x7
3F813      TBDrawF3x5
3F814      TBDrawF5x7
```

```
3F742      TBEdSfKeys
3F743      TBErrorGrob
3F744      TBFormat
3F745      TBFuncDecomp
3F747      TBInv3x5C0
3F748      TBInv3x5C1
3F749      TBInv3x5C2
3F74A      TBInv3x5C3
3F815      TBInv5x7C0
3F816      TBInv5x7C1
3F817      TBInv5x7C2
3F818      TBRollD3x5
3F819      TBRollD5x7
3F81A      TBRollU3x5
3F81B      TBRollU5x7
3F752      TBStdZooms
18D1D      TBStndSplitInfo
271C3      TCol1@
3F81C      TDDat3x5C1
3F81D      TDDat3x5C2
3F81E      TDDat3x5C3
3F81F      TDDat3x5C4
3F820      TDDat3x5C5
3F821      TDDatD3x5
3F822      TDDatL3x5
3F823      TDDatR3x5
3F824      TDDatU3x5
3F825      TDDataF3x5
27187      TFlags@
041B1      THIRTYSIX
04251      THREEFOUR
2714B      TStart@
2715F      TStep@
27173      TZoom@
3F68A      TablSetViewUI
271AF      Tcol@
```

3F68B	TempMenuBuff	

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

2FF8E	TraceY
2719B	Trow@
3F8D2	UnpackGrob
3F3E9	Update_tTYPE
3F3EA	UserSto
11CAD	VLMAlarmMsg
11CA3	VLMUserKeys
11CC1	VLMcmdlmsg
11C99	VLMhistmsg
11CB7	VLMlastargs
11C7B	VLMmsg
11C71	VLMpurgemsg
11C85	VLMstkmsg
11C8F	VLMundomsg
27047	VTick@
2706F	VZoom@
3F690	WidthSmF
09D31	XPURGEp
09D6D	XPURGEp0
09BBF	XRCLp
09C19	XRCLp0
099D6	XSTOp
09BA1	XSTOp0
26FA2	Xmax!
26FF7	Xmax2@
26FA7	Xmax@
26F8E	Xmin!
26FE3	Xmin2@
26F93	Xmin@
38C58	Xorbitmap
2701F	Ymax2@
26FCF	Ymax@
2700B	Ymin2@
26FBB	Ymin@
3F62B	ZoomAutoUI

```
3A8DD      alg=
3F148      apndit
3F149      argswap
3F2B0      argswap&&
3F2B1      argswapnext
3F826      blackbox
3F691      check_xrange
3F692      check_yrange
3F2B2      chooselst
0C476      completed
37205      d*
371A9      d<
371D0      d<=
37135      d>
374C0      d>%
371BB      d>=
372DF      dDIV
37303      dMOD
0B964      dirstrucchk
37325      dmuldiv
12802      doptr!
127F2      doptr@
3F063      dvbind
3F2B4      elsiename
3F151      filename
27416      funcCache@
3F5CE      grobCheck
3F5CF      grobCheckX
3F5D0      grobInvChk
3F5D1      grobInvChkX
3F5D4      grobInvNoChk
3F5D3      grobInvUnChkX
3F693      grobMoreDown
3F694      grobMoreUp
3F695      grobNoMore
3F699      grobPOBox2
3F69A      grobPOBox3
3F69B      grobPOBox4
```

```
3F69C      grobPOBox5
3F696      grobPOBoxP2
3F697      grobPOBoxP3
3F698      grobPOBoxP4
3F5D7      grobQueryIcon
3F5D8      grobTitleBar
3F5D2      grobUnCheckX
04409      idntany
11C0D      illnameerr
044A9      lamany
045DF      library
0B9C9      loopdirck
1538F      metaDUP
3F2B5      metainsert
3F2B6      newsymbPA
3F1D0      numargs
3F2B7      oneexpr
3F16C      onestring
3F2B8      optfilename
3F2B9      prx1
3F06A      psh1
3F2BB      psharg
3F2BC      pshmonop
123DC      rbrac
3F176      realPA
3F177      resolved
3F1D3      roll3RD
3F2BE      roll4TH
3F2BF      rollNTH
0421F      rrp
3F69D      s1>
13EA3      solverTTT
3F2C0      startSQFORM
3F2C1      syminner&
3F075      syminner&N
3F2C2      syminnertwo
26F7F      tTYPE@
3F2C3      threeexprs
```

```
3F1A2      tok;triand
3F2C4      toktriand
381B5      topic_install
382B4      topic_uninstall
3F2C5      twoexprs
3F2C6      twostrings
3F2C7      unroll3RD
3F2C8      unroll4TH
3F2C9      unrollNTH
3F079      unsyminner
3F2CA      unsymone
3F07A      verysyminner
3F07B      veryunsymin
3F2CB      vunsymfcn
3F8FB      {NoteText}
3F903      {NumVars}
3F900      {SketchSet}
0640AB   ~*ord
0630AB   ~*spec
049004   ~*xH
04A004   ~*xW
0060E8   ~::args
0130E8   ~?NULLSETDIM
02E0A1   ~AttentionKe
0830F0   ~CHECKEXISTS
08F0F0   ~COLCTDER
0940F0   ~COLCTFCNAP
0920F0   ~COLCTIFTE
0900F0   ~COLCTINTG
0930F0   ~COLCTQUOTE
0910F0   ~COLCTSUM
083002   ~COMPLEXDUMM
0580AB   ~ChkDaList
00E0A1   ~DArrow
0210F1   ~DefauStat2T
0200F1   ~DefauStatTy
0330A2   ~DoExponent
0000A1   ~DoMenuKey1N
```

```
0010A1    ~DoMenuKey2N
0020A1    ~DoMenuKey3N
0030A1    ~DoMenuKey4N
0040A1    ~DoMenuKey5N
0050A1    ~DoMenuKey6N
01F0E8    ~DoNumeric:
0320A1    ~Enter/Again
00E0A2    ~FarDArrow
00D0A2    ~FarLArrow
00F0A2    ~FarRArrow
0090A2    ~FarUArrow
0270F1    ~GETPLTLABLS
01F0F1    ~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
00D0A1    ~LArrow
0300E8    ~LIXRecv
0310E8    ~LIXSend
0330C2    ~MAKEEPIDN
0000B9    ~MiscIFMsg
0210E8    ~NotHidden
0280F1    ~POINTERR
0290F1    ~POINTEXIT
084002    ~POLARDUMMY
0000BE    ~PlotIFMsg
0330A1    ~PlotViewKey
0080C0    ~PolyNSymbF
0070C0    ~PolyNSymbP
00F0A1    ~RArrow
0620F0    ~REIM[]*
0250F1    ~SETTOPICLAM
0240F1    ~SolveApEntr
0000BF    ~SolveIFMsg
0000BB    ~StatIFMsg
0340A1    ~SymbViewKey
0350A1    ~TableViewKe
0360A1    ~TopicLibKey
0090A1    ~UpArrow
0200E8    ~UseHidden{}
0780F0    ~WHEREDER
0770F0    ~WHEREFCNAPP
0790F0    ~WHEREIFTE
07C0F0    ~WHEREINTG
07D0F0    ~WHERESUM
07A0F0    ~WHEREWHERE
04B0AB    ~XEQDIAG>L
04C0AB    ~XEQDIAG>R
04D0AB    ~XEQRANM
03B0AB    ~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
0690AB    ~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.

05D0AB	~dopcoeff
05C0AB	~dopolydiv
05B0AB	~dopolyz
00A0E8	~doptrseqn
06B0AB	~dopval
0010E8	~dosecntuple
0080E8	~dosecseqn
0000E8	~doseq
00B0E8	~doseqn
00C0E8	~doseqn#
01A0E8	~dosort
0100E8	~elsielists?
02F0E8	~etorc
0250C2	~lauserQR
0160E8	~lolatorre
0150E8	~lslatorre
05E0AB	~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
0220F1    ~nPTYPE>PINF
0040F1    ~nSCATTER
0120F1    ~nSEQTOPIC
0060F1    ~nSEQUENCE
0010F1    ~nSOLVE
0140F1    ~nSOLVETOPIC
0130F1    ~nSTATTOPIC
0660AB    ~ns*spec
0170E8    ~ollatorre
0180E8    ~ollatorre+
0040C0    ~pcoef
0050C0    ~pcoefacc1
0060C0    ~pcoefacc2
06C0AB    ~pdiv
0030C0    ~peval
0620AB    ~poly*
0600AB    ~poly+
06E0AB    ~poly/
0680AB    ~poly^
06A0AB    ~poly^#
```

```
0610AB    ~polyneg
05F0AB    ~polyprep
0010C0    ~proot
0000C0    ~proot_c
0020C0    ~proot_r
0070E8    ~ptrargs
0310C2    ~rMAKEPCOPY
0320C2    ~rPACKARRYD
06D0AB    ~resymb
081002    ~rpnAPPLY
07E002    ~rpnDER
07F002    ~rpnINTG
080002    ~rpnWHERE
02A0F1    ~runalias?
0050E8    ~seqid
00E0E8    ~seqnargs
0140E8    ~sllatorre
0650AB    ~sn*spec
0670AB    ~ss*spec
03B0F1    ~topic_CAPLE
0340F1    ~topic_NONE
0090E8    ~udfargs
0360F1    ~view01_CAPL
0370F1    ~view23_CAPL
0380F1    ~view45_CAPL
0390F1    ~view6_CAPLE
03A0F1    ~view7_CAPLE
0350F1    ~view_NONE
01D0E8    ~{$}>{id}
01E0E8    ~{$}>{lam}
01B0E8    ~{id}>{$}
01C0E8    ~{lam}>{$}
03B0F1    ~topic_CAPLET
0360F1    ~view01_CAPLET
0370F1    ~view23_CAPLET
0380F1    ~view45_CAPLET
14ED5      EQcaseDROP
```

6 UserRPL Commands

6.1 A-F

3AA49	xABS	(x → x')

Absolute Value Function
--
Returns the absolute value of its argument.
```
x         →  |x|
(x,y)     →  sqrt(x^2+y^2)
x_unit    →  |x|_unit
[ array ] →  || array ||
'sym'     →  'ABS(sym)'
```
--
Flags: -3
--
Related: NEG,SIGN

3B9E1	xACOS	(x → x')

Arc cos fn
--
Returns angle with given cos.
--
```
 z    →  arc cos z
'sym' →  'ACOS(sym)'
```
--
Related: ASIN,ATAN,COS,ISOL,ACOSH

3BAD8	xACOSH	(x → x')

Arc hyp cos fn
--
Returns val with given hyp cos.
--
```
 z    →  arc hyp cos z
'sym' →  'ACOSH(sym)'
```
--
Related: ASINH,ATANH,COSH,ISOL

```
3BCDB      xALOG                     ( x → x' )
                                     Common antilog fn
                                     --
                                     ALOG x = 10^x
                                     --
                                     Flags: -3
                                        numeric result
                                     --
                                     z      → 10^z
                                     'sym' → 'ALOG(sym)'
                                     --
                                     Related: EXP,LN,LOG
3DC94      xAND                      ( x1 x2 → x3 )
                                     And fn
                                     --
                                     Logical AND of 2 args.
                                     --
                                     #n1    #n1     → #n3
                                     "str1" "str2" → "str3"
                                     T/F1   T/F2    → 0/1
                                     T/F    'sym'   → 'T/F AND sym'
                                     'sym'  T/F     → 'sym AND T/F'
                                     'sym1' 'sym2' → 'sym1 AND sym2'
                                     --
                                     Flags: -3 -5
                                     Numeric res       -3
                                     Bin int wordsize -5 → -10
                                     --
                                     Related: NOT,OR,XOR
3D122      xAPPLY                    ( {symb1 .. symbn} f → f(symb1...symbn) )
                                     Apply to args fn
                                     --
                                     Creates expr for specified fn name & args
                                     --
                                     Related: QUOTE,|
3B5A0      xARG                      ( c → θ )
                                     Argument fn
                                     --
                                     Returns angle of a complex number
                                     --
                                     (x,y) → θ
                                     'sym' → 'ARG(sym)'
                                     --
                                     Flags: -17 -18
                                     Ang Mode -17,-18
```

```
3B95B      xASIN              ( x → x' )
                              Arc sin fn
                              --
                              Gives angle whose sin is given
                              --
                              z       → arc sin z
                              'sym' → 'ASIN(sym)'
                              --
                              Flags: -1 -3 -17 -18
                              Principal soln -1
                              Numerical res  -3
                              Angle mode     -17,-18
                              --
                              Related: ACOS,ATAN,ISOL,SIN
3BA93      xASINH             ( x → x' )
                              Arc hyp sin fn
                              --
                              Gives Val whose hyp sin is given
                              --
                              z       → arc hyp sin z
                              'sym' → 'ASINH(sym)'
                              --
                              Flags: -1 -3
                              Principal soln -1
                              Numerical res  -3
                              --
                              Related: ACOSH,ATANH,ISOL,SINH
3BA49      xATAN              ( x → x' )
                              Arc tan fn
                              --
                              Returns the angle having the tan
                              --
                              z       → arc tan z
                              'sym' → 'ATAN(sym)'
                              --
                              Flags: -1 -3 -17 -18
                              Principle soln  -1
                              Numeric results -3
                              Angle mode      -17,-18
                              --
                              Related: ACOS,ASIN,ISOL,TAN
```

```
3BB4A      xATANH              ( x → x' )
                               Arc hyp tan fn
                               --
                               Returns the value with given hyp tan.
                               --
                               z       → arc hyp tan z
                               'sym' → 'ATANH(sym)'
                               --
                               Flags: -1 -3 -22
                               Principle soln  -1
                               Numeric results -3
                               Infinite result exception -22
                               --
                               Related: ACOSH,ASINH,ISOL,TANH
3A62E      xBEEP               ( freq dur → )
                               Beep cmd
                               --
                               Sounds a tone of n Hz for x secs.
                               --
                               Flags: -56
                               Error Beep -56
                               Max Freq = 4400 Hz
                               Max Duration = 1048.575 secs.
                               --
                               Related: HALT,INPUT,PROMPT,WAIT
3EE91      xCASE               ( → )
                               CASE Conditional Structure Cmd
                               --
                               Starts CASE ... END conditional structure.
                               --
                               CASE  →
                               THEN T/F →
                               END   →
                               END   →
                               --
                               Related: END,IF,IFERR,THEN
```

```
3BEB2      xCEIL                ( x → n )
                                Ceiling Func
                                --
                                Returns the smallest integer greater than or equal to
                                the argument.
                                --
                                x      → n
                                x_u    → n_u
                                'sym' → 'CEIL(sym)'
                                --
                                Flags: -3
                                --
                                Related: FLOOR,IP,RND,TRNC
04E0AB     ~xCHOOSE             ( title {elems} pos → ob 1 )
                                ( title {elems} pos → 0 )
                                User-Defined Choose Box Cmd
                                --
                                Creates a user-defined choose box
                                --
                                Related: INFORM,NOVAL
3C53B      x%CH                 ( x1 x2 → x3 )
                                Percent Change Func
                                --
                                Returns the percent change from x (level 2) to y
                                (level 1) as a percentage of x.
                                --
                                x      y       → 100(y-x)/x
                                x      'sym'  → '%CH(x,sym)'
                                'sym'  x      → '%CH(sym,x)'
                                'sym1' 'sym2' → '%CH(sym1,sym2)'
                                x_u    y_u     → 100(y_u-x_u)/x_u
                                x_u    'sym'  → '%CH(x_u,sym)'
                                'sym'  x_u     → '%CH(sym,x_u)'
                                --
                                Flags: -3
                                --
                                Related: %,%T
3A864      xCLLCD               ( → )
                                Clear LCD Cmd
                                --
                                Clears (blanks) the stack display
                                --
                                Related: DISP,FREEZE
```

3C33A	xCNRM	([] → col_norm)

Column Norm Cmd

--

Returns the column norm (onenorm) of the array argument.

--

Related: CROSS,DET,DOT,RNRM

0460AB	~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-

0450AB	~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-

3D9F3	xCOLCT	(symb → symb')

Collect Like Terms Cmd

--

Simplifies an algebraic expression or equation by "collecting" like terms. Does not modify numbers.

--

Related: EXPAN,ISOL,QUAD,SHOW

```
3C5E4     xCOMB                   ( n k → Cn,k )
                                  Combinations Func
                                  --
                                  Returns the number of possible combinations of n
                                  items taken m at a time.
                                  --
                                  n        m        → Cn:m
                                  'symn' m         → 'COMB(symn,m)'
                                  n        'symm' → 'COMB(n,symm)'
                                  'symn' 'symm' → 'COMB(symn,symm)'
                                  --
                                  Related: PERM,!
0010AB    ~xCOND                  ( [[n*n]] → x )
                                  Conditional Number Cmd
                                  --
                                  Returns the 1-norm (column norm) condition num-
                                  ber of a square matrix.
                                  --
                                  Related: SNRM,SRAD,TRACE
3AA9F     xCONJ                   ( x → x' )
                                  Conjugate Analytic Func
                                  --
                                  Conjugates a complex number or a complex array.
                                  --
                                  x               → x
                                  (x,y)           → (x,-y)
                                  [ R-arr ]   → [ R-arr ]
                                  [ C-arr ]1 → [ C-arr ]2
                                  'sym'           → 'CONJ(sym)'
                                  --
                                  Flags: -3
                                  --
                                  Related: ABS,IM,RE,SCONJ,SIGN
3A8C0     xCONT                   ( → )
                                  Continue Program Execution Cmd
                                  --
                                  Resumes execution of a halted program.
                                  --
                                  Related: HALT,KILL,PROMPT
```

2BF46   xCORR   (→ x_correlation)
                         Correlation Cmd
                         --
                         Returns the correlation coefficient of the indepen-
                         dent and dependent data columns in the current
                         statistics matrix (reserved variable ΣDAT).
                         --
                         <REF>TEXT:Reserved|ΣDAT
                         --
                         Related: COLΣ,COV,PREDX,PREDY,XCOL,YCOL

3B7D5   xCOS   (x → x')
                         Cos Func
                         --
                         Returns the cos of the argument.
                         --
                         z   → cos z
                         'sym'   → 'COS(sym)'
                         x_uangular → cos(x_uangular)
                         --
                         Flags: -3 -17 -18
                         --
                         Related: ACOS,SIN,TAN

3B8C7   xCOSH   (x → x')
                         Hyp Cos Func
                         --
                         Returns the hyp cos of the argument.
                         --
                         z   → cosh z
                         'sym' → 'COSH(sym)'
                         --
                         Flags: -3
                         --
                         Related: ACOSH,SINH,TANH

2BF61   xCOV   (→ x_covariance)
                         Covariance Cmd
                         --
                         Returns the sample covariance of the independent
                         and dependent data columns in the current stat
                         matrix (reserved variable ΣDAT).
                         --
                         <REF>TEXT:Reserved|ΣDAT
                         --
                         Related: COLΣ,CORR,PCOV,PREDX,PREDY,
                         XCOL,YCOL

3C3EC	xCROSS	([1] [2] → [3])
		Cross Product Cmd
		--
		CROSS returns the cross product [3] = [1] x [2] of vectors [1] and [2].
		--
		Related: CNRM,DET,DOT,RNRM
0480AB	~xCSWP	
3C36D	xDET	([[]] → x)
		Determinant Func
		--
		Returns the determinant of a square matrix.
		--
		Related: CNRM,CROSS,DOT,RNRM
3EFBF	xDIR	
3A614	xDISP	(obj n_line →)
		Display Cmd
		--
		Displays obj in the nth display line.
		--
		Related: FREEZE,HALT,INPUT,PROMPT
3E695	xDO	(→)
		DO Indefinite Loop Structure Cmd
		--
		Starts DO ... UNTIL ... END indefinite loop structure.
		--
		DO →
		UNTIL →
		END T/F →
		--
		Related: END,UNTIL,WHILE
3A3D2	xDOERR	(n →)
		(#n →)
		($ →)
		(0 →)
		Do Error Cmd
		--
		Executes a "user-specified" error, causing a program to behave exactly as if a normal error had occurred during program execution.
		--
		Related: ERRM,ERRN,ERR0

0590AB	~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
0540AB	~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
3C3A0	xDOT	([1] [2] → x)
		Dot Product Cmd
		--
		Returns the dot product AoB of two arrays A and B, calculated as the sum of the products of the corresponding elements of the two arrays.
		--
		Related: CNRM,CROSS,DET,RNRM

3C223	xD>R	(x → (π/180)x) Degrees to Radians Func -- Converts a real number representing an angle in degrees to its equivalent in radians. -- x → (π/180) x 'sym' → 'D→R(sym)' -- Related: R→D UserRPL: xD→R
3AB5B	xCONSTANTe	(→ e) e Func -- Returns the symbolic constant e or its numerical representation, 2.71828182846. -- Related: EXP,EXPM,i,LN,LNP1,MAXR,MINR,π UserRPL: xe
0070AB	~xEGV	([[]] → [[evect]]' [evals]) Eigenvalues and Eigenvectors Command -- Computes the eigenvalues and right eigenvectors for a square matrix. -- Related: EGVL
0080AB	~xEGVL	([[]] → [egval]) Eigenvalues Cmd -- Computes the eigenvalues of a square matrix. -- Related: EGV
3E54D	xELSE	(→) ELSE Cmd -- Starts false clause in conditional or error-trapping structure. See the IF and IFERR keyword entries for syntax information. -- Related: IF,CASE,DO,ELSE,IFERR,REPEAT, THEN,UNTIL,WHILE

3EDD3	xENDDO	(1/0 →)

END Cmd
--
Ends conditional, error-trapping, and indefinite loop structures. ; See the IF, CASE, IFERR, DO, and WHILE keyword entries for syntax information.
--
Related: IF,CASE,DO,ELSE,IFERR,REPEAT, THEN,UNTIL,WHILE UserRPL: xEND

3E568	xIFEND

END Cmd
--
Ends conditional, error-trapping, and indefinite loop structures.
--
See the IF, CASE, IFERR, DO, and WHILE keyword entries for syntax information.
--
Related: IF,CASE,DO,ELSE,IFERR,REPEAT, THEN,UNTIL,WHILE UserRPL: xEND

3EDB3	xWHILEEND

END Cmd
--
Ends conditional, error-trapping, and indefinite loop structures.
--
See the IF, CASE, IFERR, DO, and WHILE keyword entries for syntax information.
--
Related: IF,CASE,DO,ELSE,IFERR,REPEAT, THEN,UNTIL,WHILE UserRPL: xEND

0560AB	~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

3A400	xERR0	(→)

Clear Last Error Number Cmd
--
Clears the last error number so that a subsequent execution of ERRN returns # 0h, and clears the last error message.
--
Related: DOERR,ERRM,ERRN

3A42A	xERRM	(→ `$msg`)
		Error Message Cmd
		--
		Returns a string containing the error message of the most recent calculator error.
		--
		Related: DOERR,`ERRN`,`ERR0`
3A415	xERRN	(→ `$nerr`)
		Error Number Cmd
		--
		Returns the error number of the most recent calculator error.
		--
		Related: DOERR,`ERRM`,`ERR0`
3A43F	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

```
3BBAD       xEXP            ( x → x' )
                            Exponential Analytic Func
                            --
                            Returns the exponential, or natural antilogarithm,
                            of the argument; that is, e raised to the given power.
                            --
                            z      → ez
                            'sym' → 'EXP(sym)'
                            --
                            Related: ALOG,EXPM,LN,LOG
3DA36       xEXPAN          ( symb1 → symb2 )
                            Expand Products Cmd
                            --
                            Rewrites an algebraic expression or equation by ex-
                            panding products and powers.
                            --
                            Related: COLCT,EXPAND,ISOL,QUAD,SHOW
3BD5B       xEXPM           ( x → x' )
                            Exponential Minus 1 Analytic Func
                            --
                            Returns e^x - 1.
                            --
                            x      → e^x - 1
                            'sym' → 'EXPM(sym)'
                            --
                            Related: EXP,LNP1
3BE77       xFLOOR          ( x → n )
                            Floor Func
                            --
                            Returns the greatest integer that is less than or equal
                            to the argument.
                            --
                            x      → n
                            x_u    → n_u
                            'sym' → 'FLOOR(sym)'
                            --
                            Related: CEIL,IP,RND,TRNC
```

```
3E7A8      xSTARTVAR            ( start finish → )
```
FOR Definite Loop Structure Cmd
--
Starts
```
FOR ... NEXT and
FOR ... STEP
```
definite loop structures.
--
```
FOR  xstart xfinish         →
             NEXT           →
FOR  xstart xfinish         →
    STEP   xincrement       →
    STEP   'symincrement'   →
```
--
Related: NEXT,START,STEP UserRPL: xFOR

```
3BE3C      xFP                  ( x → x' )
```
Fractional part Func
--
Returns the fractional part of an argument.
--
```
x       → y
x_u     → y_u
'sym' → 'FP(sym)'
```
--
Related: IP

```
3DC48      xFUNCTION            ( → )
```
Function Plot Type Cmd
--
Sets the plot type to FUNCTION.
--
Related: BAR,CONIC,DIFFEQ,FASTEQ,FAST3D,
GRIDMAP,HISTOGRAM,PARAMETRIC,PARSURFACE,
PCONTOUR,POLAR,SCATTER,SLOPEFIELD,TRUTH,
WIREFRAME,YSLICE

6.2 G-M

```
0520AB      ~xHEAD               ( {} → ob )
                                 ( $ → $' )
```
First Listed Element Cmd
--
Returns the first element of a list or string.
--
Related: TAIL

```
044701      ~xHELP
```

3C2E5	xHMS-	(hms1 hms2 → hms3)
		Hours-Minutes-Seconds Minus Cmd
		--
		Returns the difference of two real number, where the arguments and the result are interpreted in hours-minutes-seconds format.
		--
		Related: HMS→,→HMS,HMS+
3C2CB	xHMS+	(hms1 hms2 → hms3)
		Hours-Minutes-Seconds Plus Cmd
		--
		Returns the sum of two real number, where the arguments and the result are interpreted in hours-minutes-seconds format.
		--
		Related: HMS→,→HMS,HMS-
3C27B	x>HMS	(x → x')
		Decimal to Hours-Minutes-Seconds Cmd
		--
		Converts a real number representing hours or degrees with a decimal fraction to hours-minutes-seconds format.
		--
		Related: HMS→,HMS+,HMS- UserRPL: x→HMS
3C2A7	xHMS>	(x → x')
		Hours-Min-Sec to Decimal Cmd
		--
		Converts a real number in hours -minutes-seconds format to its decimal form (hours or degrees with a decimal fraction.
		--
		Related: →HMS,HMS+,HMS- UserRPL: xHMS→
3AB7D	xi	(→ i)

```
3E416      xIF                    ( → )
                                  IF Conditional Structure Cmd
                                  --
                                  Starts IF ...  THEN ...  END and IF ...  THEN ...
                                  ELSE ... END conditional structures.
                                  --
                                  IF         →
                                  THEN T/F →
                                  END        →
                                             →
                                  IF         →
                                  THEN T/F →
                                  ELSE       →
                                  END        →
                                  --
                                  Related: CASE,ELSE,END,IFERR,THEN
3EB17      xIFERR                 ( → )
                                  If Error Conditional Struct Cmd
                                  --
                                  Starts IFERR ...  THEN ...  END and IFERR ...
                                  THEN ... ELSE ... END error trapping structures.
                                  --
                                  Related: CASE,ELSE,END,IF,THEN
3A569      xIFT                   ( 0/1 obj → ? )
                                  IF-THEN Cmd
                                  --
                                  Executes obj if T/F is nonzero. Discards obj if T/F
                                  is zero.
                                  --
                                  Related: IFTE
3A475      xIFTE                  ( 0/1 objT objF → ? )
                                  IF-THEN-ELSE Cmd
                                  --
                                  Executes objT if T/F is nonzero. Discards objF if
                                  T/F is zero.
                                  --
                                  Related: IFT
3AB7D      xi                     ( → i )
```

```
3CA86      xIM                        ( (x,y) → y )
                                      ( [] → []' )
                                      Imaginary Part Func
                                      --
                                      Returns the imaginary part of its (complex) argu-
                                      ment.
                                      --
                                      x          → 0
                                      (x,y)      → y
                                      [ R-arr ] → [ R-arr ]
                                      [ C-arr ] → [ R-arr ]
                                      'sym'      → 'IM(sym)'
                                      --
                                      Related: C→R,RE,R→C
3B528      xINV                       ( x → 1/x )
                                      ( [[]] → [[]]' )
                                      Inverse (1/x) Analytic Func
                                      --
                                      Returns the reciprocal or the matrix inverse.
                                      --
                                      Related: SINV,/
3BE01      xIP                        ( x → n )
                                      Integer Part Func
                                      --
                                      Returns the integer part of the argument.
                                      --
                                      x      → n
                                      x_u    → n_u
                                      'sym' → 'IP(sym)'
                                      --
                                      Related: FP
3DAA0      xISOL                      ( symb var → symb' )
                                      Isolate Variable Cmd
                                      --
                                      Returns an algebraic symb' that rearranges symb to
                                      "isolate" the first occurrence of variable var.
                                      --
                                      Related: COLCT,EXPAN,QUAD,SHOW,SOLVE
3D905      xFORMUNIT
                                      UserRPL: x_
3ED7E      x'
3ED98      xENDTIC
                                      UserRPL: x'
3ED4D      x<<
                                      UserRPL: x≪
3ED68      x>>
                                      UserRPL: x≫
```

`3A879`	`xKEY`	(→ `rc 1`)

(→ `0`)

Key Cmd

--

Returns to level 1 a test result, and if a key is pressed, returns to level 2 the row-column location `xn` m of that key.

--

Related: `WAIT`,`KEYEVAL`

`3A3A8`	`xKILL`	(→)

Cancel Halted Programs Cmd

--

Cancels all currently halted programs. (Halted programs are typically canceled by pressing PRG NXT RUN KILL.) If KILL is executed within a program, that program is also canceled.

--

Related: CONT,DOERR,`HALT`,PROMPT

`3A662`	`xLAST`	(→ `ob1 .. obn`)

Last Arguments Cmd

--

Returns copies of the arguments of the most recently executed command. UserRPL: `xLASTARG`

`0000AB`	`~xLININ`	(`symb var` → `0/1`)

Linear Test Func

--

Tests whether an algebraic is structurally linear for a given variable.

`3BBF7`	`xLN`	(`x` → `x'`)

Natural Logarithm Analytic Func

--

Returns the natural (base e) logarithm of the argument.

--

```
z      → ln z
'sym' → 'LN(sym)'
```

--

Related: ALOG,EXP,ISOL,LNP1,LOG

`3BD25`	`xLNP1`	(`x` → `x'`)

Natural Log of x+1 Analytic Func

--

Returns ln (x + 1).

--

```
x      → ln(x+1)
'sym' → 'LNP1(sym)'
```

--

Related: EXPM,LN

```
3BC69      xLOG           ( x → x' )
                          Common Logarithm Analytic Func
                          --
```
Returns the common logarithm (base 10) of the argument.
```
                          --
                          z     → log z
                          'sym' → 'LOG(sym)'
                          --
```
Related: ALOG,EXP,ISOL,LN
```
00D0AB     ~xLQ           ( [[]] → [[L]] [[Q]] [[P]] )
                          LQ Factorization of a Matrix Cmd
                          --
```
Returns the LQ factorization of an nm matrix.
```
                          --
```
Related: LSQ,QR
```
0060AB     ~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,/
```
00B0AB     ~xLU           ( [[]] → [[L]] [[U]] [[P]] )
                          LU Dec of a Sq. Matrix Cmd
                          --
```
Returns the LU decomposition of a square matrix.
```
                          --
```
Related: DET,INV,LSQ,/
```
3C1F2      xMANT          ( x → x' )
                          Mantissa Func
                          --
```
Returns the mantissa of the argument.
```
                          --
                          x     → ymant
                          'sym' → 'MANT(sym)'
                          --
```
Related: SIGN,XPON
```
3D8C6      xMATCHDN       ( symb {spat srepl} → symb' 0/1 )
                          ( symb {spat srepl scond} → symb' 0/1 )
                          Match Pattern Down Cmd
                          --
```
Rewrites an expression.
```
                          --
```
Related: X↑MATCH UserRPL: x↓MATCH

3D898	xMATCHUP	(symb {spat srepl} → symb' 0/1)
		(symb {spat srepl scond} → symb' 0/1)
		Bottom-Up Match and Replace Cmd

--

Rewrites an expression.

--

Related: X∇MATCH UserRPL: x↑MATCH

3BF1E	xMAX	(x y → x')
		Maximum Func

--

Returns the greater (more positive) of the arguments.

--

```
x        y        → max(x, y)
x       'sym'   → 'MAX(x, sym)'
'sym'   x        → 'MAX(sym, x)'
'sym1' 'sym2' → 'MAX(sym1, sym2)'
x_u1    y_u2   → max(x_u1, y_u2)
```

--

Related: MIN

3AB17	xMAXR	(→ MAXR)
		Maximum Real Func

--

Returns the symbolic constant 'MAXR' or its numerical representation, 9.99999999999E499.

--

```
→ 'MAXR'
→ 9.99999999999E499
```

--

Related: E,e,i,MINR,π

3BFF9	xMIN	(x y → x')
		Minumum Func

--

Returns the lesser (more negative) of its two arguments.

--

```
x        y        → min(x, y)
x       'sym'   → 'MIN(x, sym)'
'sym'   x        → 'MIN(sym, x)'
'sym1' 'sym2' → 'MIN(sym1, sym2)'
x_u1    y_u2   → min(x_u1, y_u2)
```

--

Related: MAX

```
3AB39      xMINR                    ( → MINR )
                                    Minimum Real Func
                                    --
                                    Returns the symbolic constant 'MINR' or its numer-
                                    ical representation, 1.00000000000E-499.
                                    --
                                    → 'MAXR'
                                    → 1.00000000000E-499
                                    --
                                    Related: e,i,MAXR,π
3C1A3      xMOD                     ( x y → x' )
                                    Modulo Func
                                    --
                                    Returns a remainder defined by: x mod y = x - y
                                    floor (x/y)
                                    --
                                    x        y        → x mod y
                                    x        'sym'    → 'MOD(x, sym)'
                                    'sym'    x        → 'MOD(sym, x)'
                                    'sym1'   'sym2'   → 'MOD(sym1, sym2)'
                                    --
                                    Related: FLOOR,/
04F0AB     ~xMSGBOX                 ( $ → )
                                    Message Box Cmd
                                    --
                                    Creates a user-defined message box.
                                    --
                                    Related: CHOOSE,INFORM,PROMPT
```

6.3 N-S

```
3A9B8      xNEG                     ( x → x' )
                                    Negate Analytic Func
                                    --
                                    Changes the sign or negates an object.
                                    --
                                    z        → -z
                                    #n1      → #n2
                                    [ arr ]  → [ -arr ]
                                    'sym'    → '-(sym)'
                                    x_u      → -x_u
                                    grob1    → grob2
                                    PICT1    → PICT2
                                    --
                                    Related: ABS,CONJ,NOT,SIGN
```

3E98E	xNEXT	(→)

NEXT Cmd

--

Ends definite loop structures. See the FOR and START command entries for syntax information.

--

Related: FOR,START,STEP

3DDBE	xNOT	(x → x')

`NOT` Cmd

--

Returns the one's complement or the logical inverse of the argument.

--

```
#n1    → #n2
T/F    → 0/1
"str1" → "str2"
'sym'  → 'NOT sym'
```

--

Related: `AND`,`OR`,`XOR`

0550AB	~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

3A648	x>NUM	(x → x')

Evaluate to Number Cmd

--

Evaluates a symbolic argument object and returns the numerical result.

--

```
objsym → z
```

--

Related: →Q,→Qpi UserRPL: x→NUM

3A3BD	xOFF	(→)

Off Cmd

--

Turns off the calculator.

--

Related: CONT,`HALT`,KILL

```
3DD33      xOR              ( x y → x' )
                            OR Func
                            --
                            Returns the logical OR of two arguments.
                            --
                            #n1    #n2    → #n3
                            "str1" "str2" → "str3"
                            T/F1   T/F2   → 0/1
                            T/F    'sym'  → 'T/F OR sym'
                            'sym'  T/F    → 'sym OR T/F'
                            'sym1' 'sym2' → 'sym1 OR sym2'
                            --
                            Related: AND,NOT,XOR
```

3DC60   xPARAMETRIC   (→)
                            Parametric Plot Type Cmd
                            --
                            Sets the plot type to PARAMETRIC.
                            --
                            Related: BAR,CONTOUR,DIFFEQ,FUNCTION,
                            GRIDMAP,HISTOGRAM,PARSURFACE,PCONTOUR,
                            POLAR,SCATTER,SLOPEFIELD,TRUTH,   WIRE-
                            FRAME,YSLICE

0110AB   ~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

3C624   xPERM   (n k → n')
                            Permutations Func
                            --
                            Returns the number of possible permutations of n
                            items taken m at a time.
                            --

```
                            n      m      → Pn,m
                            'symn' m      → 'PERM(symn,m)'
                            n      'symm' → 'PERM(n,symm)'
                            'symn' 'symm' → 'PERM(symn,symm)'
```

                            --
                            Related: COMB,!

0120AB	~xPEVAL	([coefs] x → x')
		Polynomial Evaluation Cmd
		--
		Evaluates an n-degree polynomial at x.
		--
		Related: PCOEFF,PROOT
03F701	~xPINIT	(→)
		Port Initialize Cmd
		--
		Initializes all currently active ports. Does not affect data already stored in a port.
3DC54	xPOLAR	(→)
		Polar Plot Type Cmd
		--
		Sets the plot type to POLAR.
		--
		Related: BAR,CONIC,DIFFEQ,FUNCTION, GRIDMAP,HISTOGRAM,PARAMETRIC,PARSURFACE, PCONTOUR,SCATTER,SLOPEFIELD,TRUTH, WIREFRAME,YSLICE
0100AB	~xPROOT	([coefs] → [roots])
		Polynomial Roots Cmd
		--
		Returns all roots of an n-degree polynomial having real or complex roots.
		--
		Related: PCOEFF,PEVAL
0A20F2	~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
2BDA2	xPVARS	(nport → {} mem)
		Port-Variables Cmd
		--
		Returns a list of the backup objects (:nport:name) and the lib- rary objects (:nport:nlibrary) in the specified port. Also returns the available memory size (if RAM) or the memory type.
		--
		Related: VARS

```
3D80A      x->Q                ( x → a/b )
                               To Quotient Cmd
                               --
                               Returns a rational form of the argument.
                               --
                               x       → 'a/b'
                               (x,y)   → 'a/b+c/d*i'
                               'sym1' → 'sym2'
                               --
                               Related: →Qπ,/ UserRPL: x→Q
3D829      x->QPI              ( x → symb )
                               To Quotient Times π Cmd
                               --
                               Returns a rational form of the argument, or a ra-
                               tional form of the argument with π factored out,
                               whichever yields the smaller denominator.
                               --
                               x       → 'a/b*π'
                               x       → 'a/b'
                               'sym1' → 'symb2'
                               (x,y)   → 'a/b*π+c/d*π*i'
                               (x,y)   → 'a/b+c/d*i'
                               --
                               Related: →Q,/,π UserRPL: x→Qπ
00C0AB     ~xQR                ( [[]] → [[Q]] [[R]] [[P]] )
                               QR Factorization of a Matrix Cmd
                               --
                               Returns the QR factorization of an nm matrix.
                               --
                               Related: LQ,LSQ
3DB03      xQUAD               ( symb var → symb' )
                               Solve Quadratic Equation Cmd
                               --
                               Solves an algebraic object symb for the variable var,
                               and returns an expression symb' representing the so-
                               lution.
                               --
                               Related: COLCT,EXPAN,ISOL,SHOW,SOLVE
```

3D068	xQUOTE	(ob → 'ob)
		Quote Argument Func
		--
		Returns its argument unevaluated.
		--
		'sym' → 'sym'
		obj → obj
		--
		Related: APPLY,|
3C5AD	xRAND	(→ x)
		Random Number Cmd
		--
		Returns a pseudo-random number generated using a seed value, and updates the seed value.
		--
		Related: COMB,PERM,RDZ,!
0050AB	~xRANK	([[]] → n)
		Matrix Rank Cmd
		--
		Returns the rank of a rectangular matrix.
		--
		Related: LQ,LSQ,QR
03C0AB	~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

```
3D977      xPREDIV                       ( x y → x/y )
                                         Prefix Divide Func
                                         --
                                         Prefix form of / (divide) generated by the Equation
                                         Writer Application.
                                         --
                                         z1     z2      → z1/z2
                                         [arr]  [[mat]] → [[arrmat^-1]]
                                         [arr]  z       → [arr/z]
                                         z      'sym'   → 'z/sym'
                                         'sym'  z       → 'sym/z'
                                         'sym1' 'sym2'  → 'sym1/sym2'
                                         #n1    n2      → #n3
                                         n1     #n2     → #n3
                                         #n1    #n2     → #n3
                                         x_u1   y_u2    → (x/y)_u1/u2
                                         x      y_u     → (x/y)_1/u
                                         x_u    y       → (x/y)_u
                                         'sym'  x_u     → 'sym/x_u'
                                         x_u    'sym'   → 'x_u/sym'
                                         --
                                         Related: / UserRPL: xRATIO
0490AB     ~xRCI                         ( [[]] x nrow → [[]]' )
                                         [] x n []'
                                         Multiply Row by Constant Cmd
                                         --
                                         Multiplies row n of a matrix (or element n of a vec-
                                         tor) by a const x, and returns the modified matrix.
                                         --
                                         Related: RCIJ
04A0AB     ~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
```

```
3C5C2      xRDZ                 ( x → )
                                Randomize Cmd
                                --
                                Uses a real number xseed as a seed for the RAND
                                command.
                                --
                                Related: COMB,PERM,RAND,!
3CA30      xRE                  ( (x,y) → x )
                                ( [] → []' )
                                Real Part Func
                                --
                                Returns the real part of the argument.
                                --
                                x           → x
                                x_u         → x
                                (x,y)       → x
                                [ R-arr ] → [ R-arr ]
                                [ C-arr ] → [ R-arr ]
                                'sym'       → 'RE(sym)'
                                --
                                Related: C→R,IM,R→C
3E634      xREPEAT              ( 1/0 → )
                                REPEAT Cmd
                                --
                                Starts loop clause in
                                WHILE ... REPEAT ... END
                                indefinite loop structure.
                                --
                                Related: END,WHILE
0180AB     ~xREVLIST            ( {1...n} → {n...1}' )
                                Reverse List Cmd
                                --
                                Reverses the order of the elements in a list.
                                --
                                Related: SORT
```

3C07F	xRND	(x n → x')

Round Func

--

Rounds an object to a specified number of decimal places or significant digits, or to fit the current display format.

--

```
z1    nrnd      → z2
z     'symrnd'  → 'RND(z,symrnd)'
'sym'  nrnd      → 'RND(symb,nrnd)'
'sym1' 'symrnd' → 'RND(sym1,symrnd)'
[arr1] nrnd      → [arr2]
x_u    nrnd      → y_u
x_u    'symrnd' → 'RND(x_u,symrnd)'
```

--

Related: TRNC

3C307	xRNRM	([] → x)

Row Norm Cmd

--

Returns the row norm (infinity norm) of its argument array.

--

Related: CNRM,CROSS,DET,DOT

3CCB5	xROOT	(prog/s var guess → x)
		(prog/s var {guesses} → x)

Root-Finder Cmd

--

Returns a real number xroot that is a value of the specified variable var for which the specified program or algebraic object most nearly evaluates to zero or a local extremum.

0430AB	~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

0440AB	~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
00F0AB	~xRREF	([[]] → [[]]') Reduced Row Echelon Form Cmd -- Converts a rectangular matrix to a reduced row echelon form.
3C430	xRSD	([B] [[A]] [Z] → []') ([[B]] [[A]] [[Z]] → [[]]') Residual Cmd -- Computes the residual B - AZ of the arrays B, A, and Z.
0470AB	~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-
3DA79	xRULES	
3C24F	xR>D	(x → (180/π)x) Radians to Degrees Func -- Converts a real number expressed in radians to its equivalent in degrees. -- x → (180/π)x 'sym' → 'R→D(sym)' -- Related: D→R UserRPL: xR→D

3DC72	xSAME	(ob1 ob2 → 1/0) Display information about the makers of the calcu- lator. Same Object Cmd -- Compares two objects, and returns a true result (1) if they are identical, and a false result (0) if they are not. -- Related: TYPE,==
00E0AB	~xSCHUR	([[]] → [[Q]] [[T]]) Schur Decomp. of Squ. Matrix Cmd -- Returns the Schur decomposition of a square matrix. -- Related: LQ,LU,QR,SVD,SVL,TRN
0130AB	~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
3DB51	xSHOW	(symb name → symb') (symb {names} → symb') Show Variable Cmd -- Returns symb' which is equivalent to symb except that all implicit references to a variable name are made explicit. -- Related: COLCT,EXPAN,ISOL,QUAD
3B5EF	xSIGN	(x → x') Sign Func -- Returns the sign of a real number argument, the sign of the numerical part of a unit object argument, or the unit vector in the direction of a complex number argument. -- Related: ABS,MANT,XPON

```
3B781      xSIN              ( x → x' )
                            Sine Analytic Func
                            --
                            z          → sin z
                            'sym'      → 'SIN(sym)'
                            x_uangular → sin(x_uangular)
                            --
                            Related: ASIN,COS,TAN
3B87D      xSINH             ( x → x' )
                            Hyperbolic Sine Analytic Func
                            --
                            Returns the hyperbolic sine of the argument.
                            --
                            z     → sinh z
                            'sym' → 'SINH(sym)'
                            --
                            Related: ANUSH,COSH,TANH
0040AB     ~xSNRM            ( [] → x )
                            Spectral Norm Cmd
                            --
                            Returns the spectral norm of an array.
                            --
                            Related:
                            ABS,CNRM,COND,RNRM,SRAD,TRACE

0190AB     ~xSORT            ( {} → {}' )
                            Ascending Order Sort Cmd
                            --
                            Sorts the elements in a list in ascending order.
                            --
                            Related: REVLIST
3B6E6      xSQ               ( x → x' )
                            Square Analytic Func
                            --
                            Returns the square of the argument.
                            --
                            z          → z2
                            x_u        → x2_u2
                            [[ mat ]]  → [[ mat  mat ]]
                            'sym'      → 'SQ(sym)'
                            --
```
Related: $\sqrt{a}$,^

```
0030AB    ~xSRAD              ( [[]] → x )
                              Spectral Radius Cmd
                              --
                              Returns the spectral radius of a square matrix.
                              --
                              Related: COND,SNRM,TRACE
3E710     xSTART              ( start finish → )
                              START Definite Loop Structure Cmd
                              --
                              START xstart xfinish        →
                              NEXT  xstart xfinish        →
                                    STEP   xincrement     →
                                    STEP  'symbincrement' →
                              --
                              Related: FOR,NEXT,STEP
3EABD     xSTEP               ( n → )
                              ( symb → )
                              STEP Cmd
                              --
                              Defines the increment (step) value, and ends definite
                              loop struct See the FOR and START command en-
                              tries for syntax information.
                              --
                              Related: FOR,BEXT,START
0570AB    ~xSTREAM            ( {} prog → x )
                              Stream Execution Cmd
                              --
                              Moves the first two elements from the list onto the
                              stack, and executes prog. The moves the next ele-
                              ment (if any) onto the stack, and executes obj again
                              using the previous result and the new element. Re-
                              peats this until the list is exhausted, and returns the
                              final result.
                              --
                              Related: DOSUBS
0090AB    ~xSVD               ( [[]] → [[U]] [[V]] [S] )
                              Singular Value Decomposition Cmd
                              --
                              Returns the sigular value decomposition of an mn
                              matrix.
                              --
                              Related: DIAG→,MIN,SVL
```

00A0AB	~xSVL	([[]] → [])
		Singular Values Cmd
		--
		Returns the singular values of an mn matrix.
		--
		Related: MIN,SVD
3A5C4	xSYSEVAL	(# → ?)
		Evaluate System Object Cmd
		--
		Evaluates unnamed operating system objects specified by their memory addresses.
		--
		Related: EVAL,LIBEVAL,FLASHEVAL

6.4 T-Z

3C4C9	x%T	(x y → 100y/x)
		Percent of Total Function
		--
		Returns the percent of the level 2 argument that is represented by the level 1 argument.

```
x       y        →  100y/x
x       'sym'    →  '%T(x,sym)'
'sym'   x        →  '%T(sym,x)'
'sym1'  'sym2'   →  '%T(sym1,sym2)'
x_u1    y_u2     →  100y_u2/x_u1
x_u     'sym'    →  '%T(x_u,sym)'
'sym'   x_u      →  '%T(sym,x_u)'
```

		--
		Related: %,%ch
0530AB	~xTAIL	({} → {}')
		($ → $')
		Last Listed Elements Cmd
		--
		Returns all but the first element of a list or string.
		--
		Related: HEAD

```
3B829      xTAN              ( x → x' )
                            Tangent Analytic Func
                            --
                            Returns the tangent of the argument.
                            --
                            z          → tan z
                            'sym'      → 'TAN(symb)'
                            x_unitang  → tan(x_unitang)
                            -
                            Related: ATAN,COS,SIN
3B911      xTANH             ( x → x' )
                            Hyperbolic Tangent Analytic Func
                            --
                            Returns the hyperbolic tangent of the argument.
                            --
                            z      → tanh z
                            'sym'  → 'TANH(sym)'
                            -
                            Related: ATANH,COSH,SINH
3DBDC      xTAYLR            ( symb var n → symb' )
                            Taylor's Polynomial Cmd
                            --
```

Calculates the nth order Taylor's polynomial of 'symb' in the variable var.

--

Related: $\partial, \int, \Sigma$

```
3E4AB      xTHEN             ( 0/1 → )
                            THEN Cmd
                            --
```

Starts the true-clause in conditional or error-trapping structure

--

Related: CASE,ELSE,END,IFERR

```
3EF1F      xTHENCASE
```

THEN in a CASE statement.

--

Related: CASE,ELSE,END,IFERR UserRPL: `xTHEN`

```
3EE34      xERRTHEN
```

THEN in an ON `ERROR` construct.

--

Related: CASE,ELSE,END,IFERR UserRPL: `xTHEN`

```
0020AB     ~xTRACE           ( [[]] → x )
                            Matrix Trace Cmd
                            --
```

Returns the trace of a square matrix.

```
3C111      xTRNC                     ( x n → )
                                     Truncate Func
                                     --
                                     Truncates an object to a specified number of deci-
                                     mal places or significant digits, or to fit the current
                                     display format.
                                     --
                                     z1        ntrnc      → z2
                                     z1        'symtrnc' →
                                     'TRNC(z1,symtrnc)'
                                     'sym1'    ntrnc      →
                                     'TRNC(sym1,ntrnc)'
                                     'sym1'    'symtrnc' →
                                     'TRNC(sym1,symtrnc)'
                                     [ arr ]1 ntrnc       → [ arr ]2
                                     x_u       ntrnc      → y_u
                                     x_u       'symtrnc' →
                                     'TRNC(x_u,symtrnc)'
                                     --
                                     Related: RND
3E700      xUNTIL                    ( → )
                                     UNTIL Cmd
                                     --
                                     Starts test-clause in
                                     DO ... UNTIL ... END
                                     indefinite loop structure.
                                     --
                                     See the DO entry for syntax info.
                                     --
                                     Related: DO,END
03A0AB     ~xVERSION                 ( → $ $ )
                                     Software Version Cmd
                                     --
                                     Displays the software version and copyright message.

3A777      xWAIT                     ( sec → )
                                     ( 0 → rc.p )
                                     Wait Cmd
                                     --
                                     Suspends program execution for specified time, or
                                     until a key is pressed.
                                     --
                                     Related: KEY
```

```
3E5C1      xWHILE                    ( → )
                                     WHILE Indefinite Loop Struct Cmd
                                     --
                                     Starts the
                                     WHILE ... REPEAT ... END
                                     indefinite loop structure.
                                     --
                                     Related: DO,END,REPEAT
3DE2A      xXOR                      ( # #' → #'' )
                                     ( $ $' → $'' )
                                     ( 1/0 1/0 → 1/0 )
                                     Exclusive OR Cmd
                                     --
                                     Returns the logical exclusive OR of two arguments.
                                     #n1     #n2      → #n3
                                     "str1" "str2" → "str3"
                                     T/F1    T/F2     → 0/1
                                     T/F     'sym'    → 'T/F XOR sym'
                                     'sym'   T/F      → 'sym XOR T/F'
                                     'sym1' 'sym2' → 'sym1 XOR sym2'
                                     --
                                     Related: AND,OR,NOT
3BEED      xXPON                     ( % → n )
                                     Exponent Func
                                     --
                                     Returns the exponent of the arg.
                                     --
                                     Related: MANT,SIGN
0510AB     ~xXRECV                   ( name → )
                                     XModem Receive Cmd
                                     --
                                     Prepares the HP 48 to receive an object via XMo-
                                     dem.  The received object is stored in the given
                                     name.
                                     --
                                     Related: BAUD,RECV,RECN,SEND,XSEND
3B36B      xXROOT                    ( y x → Y' )
                                     xth Root of y Cmd
                                     --
                                     Computes the xth root of a real number.
                                     y        x       → x ROOT y
                                     'sym1' 'sym2' → 'XROOT(sym2,sym1)'
                                     'sym'   x       → 'XROOT(x,sym)'
                                     y       'sym'   → 'XROOT(sym,y)'
                                     y_u     x       → x ROOT y_u1/x
                                     y_u     'sym'   → 'XROOT(sym,y_u)'
```

```
0500AB    ~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

```
3B178     x^                        ( y x → y^x )
                                     Power Analytic Func
                                     --
```
Returns the value of the level 2 object raised to the
power of the level 1 object.
```
                                     w       z        → w^z
                                     z       'sym'    → 'z^sym'
                                     'sym'   z        → '(sym)^z'
                                     'sym1' 'sym2'    → 'sym1^(sym2)'
                                     x_u     y        → xy_uy
                                     x_u     'sym'    → '(x_u)^(sym)'
                                     --
                                     Flags: -1 -3
                                     Principal soln  -1
                                     Numeric results -3
                                     --
```
Related: EXP,ISOL,LN,XROOT
```
3B639     xSQRT                      ( x → x' )
                                     Square Root Analytic Func
                                     --
```
Returns the (+ve) square root of the argument.
```
                                     --
                                     z      → √a z
                                     x_u    → √a (x)_u
                                     'sym'  → 'SQRT(sym)'
                                     --
                                     Flags: -1 -3
                                     --
```
Related: SQ,^,ISOL UserRPL: x√a

```
3AAF5      xPI                        ( → π )
                                      PI Func
                                      --
```
Returns the symbolic constant 'π' or its numerical
representation, 3.14159265359.
```
                                      → 'π'
                                      → 3.14159265359
                                      --
```
Flags: -2 -3
--
Related: e,i,MAXR,MINR,→Qπ UserRPL: xπ
```
3E049      x<=?                       ( x y → 1 )
                                      ( x y → 0 )
                                      Less Than or Equal Func
                                      --
```
Tests whether one object is less than or equal to
another object.
--
```
x        y         → 0/1
#n1      #n2       → 0/1
"str1" "str2"  → 0/1
x         'sym'   → 'x<=sym'
'sym'   z         → 'sym<=z'
'sym1' 'sym2'  → 'sym1<=sym2'
x_u1    y_u2    → 0/1
x_u      'sym'   → 'x_unit<=sym'
'sym'   x_u      → 'sym<=x_unit'
--
```
Flags: -3
--
Related: <,>,≥,==,≠ UserRPL: x≤

```
3E0ED      x>=?                    ( x y → 1 )
                                   ( x y → 0 )
                                   Greater Than or Equal Func
                                   --
                                   x       y       → 0/1
                                   #n1     #n2     → 0/1
                                   "str1" "str2" → 0/1
                                   x       'sym'   → 'x≥sym'
                                   'sym'   z       → 'sym≥z'
                                   'sym1' 'sym2' → 'sym1≥sym2'
                                   x_u1    y_u2    → 0/1
                                   x_u     'sym'   → 'x_u≥sym'
                                   'sym' x_u       → 'sym≥x_u'
                                   --
                                   Flags: -3
                                   --
                                   Related: <,≤,>,==,≠ UserRPL: x≥
3DF0F      x#?                     ( x y → 1 )
                                   ( x y → 0 )
                                   Not Equal Func
                                   --
                                   Tests if two objects are equal.
                                   obj1    obj2    → 0/1
                                   (x,0)   x       → 0/1
                                   x       (x,0)   → 0/1
                                   z       'sym'   → 'z≠sym'
                                   'sym'   z       → 'sym≠z'
                                   'sym1' 'sym2' → 'sym1≠sym2'
                                   --
                                   Flags: -3
                                   --
                                   Related: SAME,TYPE,<,≤,>,≥, == UserRPL: x≠
3EC15      xRPN->                  ( ob1 .. obn → )
                                   Create Local Variables Cmd
                                   --
                                   Creates local variables.
                                   obj1 ... objn →
                                   --
                                   Syntax:
                                   → name1 name2 ... nameN ≪ prog ≫
                                   → name1 name2 ... nameN 'Expr'
                                   --
                                   Related: DEFINE,STO UserRPL: x→
3E57E      xALG->
                                   Create local variable comand. <REF>xRPN-> User-
                                   RPL: x→
```

```
3BD96      xFACT                      ( x → x' )
                                      Factorial (Gamma) Func
                                      --
                                      Returns the factorial n! of a positive integer argu-
                                      ment n, or the gamma function (x+1) of a non-integer
                                      argument x.
                                      n     → n!
                                      x     → (x+1)
                                      'sym' → '(sym!)'
                                      --
                                      Flags: -3 -20 -21
                                      Numerical Results   -3
                                      Underflow exception -20
                                      Overflow exception  -21
                                      --
                                      Related: COMB,PERM UserRPL: x!
3C452      x%                         ( x y → xy/100 )
                                      Percent Func
                                      --
                                      Returns x (level 2) percent of y (level 1).
                                      x       y        → xy/100
                                      x       'sym'    → '%(x,sym)'
                                      'sym'   x        → '%(sym,x)'
                                      'sym1' 'sym2'    → '%(sym1,sym2)'
                                      x       y_unit   → (xy/100)_unit
                                      x_unit y         → (xy/100)_unit
                                      'sym'   x_unit   → '%(sym,x_unit)'
                                      x_unit 'sym'     → '%(x_unit,sym)'
                                      --
                                      Flags:
                                      Numerical Results -3
                                      --
                                      Related: %CH,%T
```

```
3AB9F        x+                      ( x y → x+y )
                                     Add Analytic Func
                                     --
                                     Returns the sum of the arguments. Addition. If one
                                     arg is list, insert element in list or concatenate lists.
                                     <REF>xADD
                                     z1        z2         → z1+z2
                                     [ arr ]1 [ arr ]2 → [ arr ]1+2
                                     z         'sym'      → 'z+(sym)'
                                     'symb'    z          → 'sym+z'
                                     'sym1'    'sym2'     → 'sym1 + sym2'
                                     { lst1 } { lst2 } → { lst1 lst2 }
                                     obj       { o... } → { obj o... }
                                     { o... } o         → { o... obj }
                                     "str1"    "str2"    → "str1str2"
                                     obj       "str"     → "obj str"
                                     "str"     obj       → "str obj"
                                     #n1       n2        → #n'
                                     n1        #n2       → #n'
                                     #n1       #n2       → #n'
                                     x1_u1     y_u2      → (x2+y)_u2
                                     'sym'     x_u       → 'sym+x_u'
                                     x_u       'sym'     → 'x_u+sym'
                                     grob1     grob2     → grob'
                                     --
                                     Flags: -3 -4 -5 -6 -7 -8 -9 -10
                                     Numeric results -3
                                     Bint wordsize   -5 → -10
                                     --
                                     Related: -,*,/,=
```

```
3AD68        x-                      ( x y → x-y )
                                     Subtract Analytic Func
                                     --
```
Returns the difference of the arguments: the object in level 1 is subtracted from the object in level 2.
```
                                     z1         z2          → z1-z2
                                     [ arr ]1 [ arr ]2 → [ arr ]1_2
                                     z          'sym'       → 'z-sym'
                                     'sym'      z           → 'sym-z'
                                     'sym1'     'sym2'      → 'sym1 - sym2'
                                     #n1        n2          → #n'
                                     n1         #n2         → #n'
                                     #n1        #n2         → #n'
                                     x1_u1      y_u2        → (x2-y)_u2
                                     'sym'      x_u         → 'sym-x_u'
                                     x_u        'sym'       → 'x_u-sym'
                                     --
```
Flags: -3
```
                                     Numeric results -3
                                     --
```
Related: +,*,/,=

```
3AFCA        x/                      ( x y → x/y )
                                     Divide Analytic Func
                                     --
```
Returns the quotient of the arguments: the level 2 object divided by the level 1 object. (Abbrev. _u = _unit)
```
                                     z1         z2          → z1 / z2
                                     [ arr ] [[ mat ]] → [[mat^-1arr]]
                                     [ arr ] z             → [ arr / z ]
                                     z          'sym'       → 'z / sym'
                                     'sym'      z           → 'sym / z'
                                     'sym1'     'sym2'      → 'sym1 / sym2'
                                     #n1        n2          → #n'
                                     n1         #n2         → #n'
                                     #n1        #n2         → #n'
                                     x_u1       y_u2        → (x/y)_u1/u2
                                     x          y_u         → (x/y)_1/u
                                     x_u        y           → (x/y)_u
                                     'sym'      x_u         → 'sym/x_u'
                                     x_u        'sym'       → 'x_u/sym'
                                     --
```
Related: +,-,*,=,RATIO

```
3C685       x<                  ( x y → 1 )
                                ( x y → 0 )
                                Less Than Func
                                --
                                Tests whether one object is less than another object.
                                x       y       → 0/1
                                #n1     #n2     → 0/1
                                "str1"  "str2"  → 0/1
                                x       'sym'   → 'x<sym'
                                'sym'   x       → 'sym<z'
                                'sym1'  'sym2'  → 'sym1<sym2'
                                x_u1    y_u2    → 0/1
                                x_u     'sym'   → 'x_u<sym'
                                'sym'   x_u     → 'sym<x_u'
                                --
                                Flags: -3
                                Numeric results -3
3A8DD       x=                  ( x y → x=y )
                                Makes equation out of two expressions. Equals An-
                                alytic Func
                                --
                                Returns an equation formed from the two arguments.
                                z1      z2      → 'z1=z2'
                                z       'sym'   → 'z=sym'
                                'sym'   z       → 'sym=z'
                                'sym1'  'sym2'  → 'sym1=sym2'
                                y       x_u     → 'y=x_u'
                                y_u     x       → 'y_u=x'
                                y_u     x_u     → 'y_u=x_u'
                                'sym'   x_u     → 'sym=x_u'
                                x_u     'sym'   → 'x_u=sym'
                                --
                                Flags: -3
                                Numeric results -3
                                --
                                Related: DEFINE,EVAL,-
```

```
3C7B9      x==                        ( x y → 1 )
                                      ( x y → 0 )
                                      Logical Equality Func
                                      --
                                      Tests if two objects are equal.
                                      obj1   obj2   → 0/1
                                      (x,0)  x      → 0/1
                                      x      (x,0)  → 0/1
                                      z      'sym'  → 'z==sym'
                                      'sym'  z      → 'sym==z'
                                      'sym1' 'sym2' → 'sym1==sym2'
                                      --
                                      Flags: -3
                                      Numeric results -3
                                      --
                                      Related: SAME,TYPE,<,≤,>,≥,≠
3C729      x>                         ( x y → 1 )
                                      ( x y → 0 )
                                      Greater Than Func
                                      --
```

Tests whether one object is greater than another object.

```
                                      x      y      → 0/1
                                      #n1    #n2    → 0/1
                                      "str1" "str2" → 0/1
                                      x      'sym'  → 'x>sym'
                                      'sym'  z      → 'sym>z'
                                      'sym1' 'sym2' → 'sym1>sym2'
                                      x_u1   y_u2   → 0/1
                                      x_u    'sym'  → 'x_u>sym'
                                      'sym'  x_u    → 'sym>x_u'
                                      --
                                      Flags: -3
                                      Numeric results -3
                                      --
```

Related: <,≤,≥,==,≠ ;

7 ML Entry Points

7.1 General Purpose

```
06F73      SAVPTR                 D0 to RPLTOP
                                  D1 to DSKTOP
                                  B to RETTOP
                                  D to FREETOP
                                  Clear carry
058FC      GETPTRLOOP             <see>GETPTR , Loop to RPL
0114C      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
```

7.4.2 Moving and Swapping Memory Areas

7.4.3 Allocating Memory in TEMPOB

7.4.4 Resizing TEMPOB Areas

```
199AA      Shrink$                Shrinks a strobj
                                  R0.A=->$ D0=end of $
                                  Uses A.W B.A C.W D.10 D0 D1
```

7.4.5 Working with Memory

7.5 Display

7.6 Popping and Pushing

7.6.1 Pointers

03672	GPOverWrALp	<see>GETPTR , OverWr A, Loop
19ACC	PopASavptr	Pop to A.A, <see>SAVPTR
03A86	PUSHA	Push A, Loop

7.6.2 TRUE and FALSE

0DDA3	GPPushFLoop	<see>GETPTR , Push FALSE, Loop
0DD8A	GPPushTLoop	<see>GETPTR , Push TRUE, Loop
0D5CB	popflag	Pop to A.A, if TRUE then set carry
0F328	$jGPOvrWrFLp	
0F336	$jGPOvrWrTLp	
0F39F	$jOvrWrF/TLp	
0F3A6	$jOvrWrFLoop	
0F3AD	$jOvrWrT/FLp	
0F3B4	$jOvrWrTLoop	
0F40F	$jPshF/TLoop	
0F439	$jPshT/FLoop	

7.6.3 System Binary Integers (BINT)

06E19	POP#	Pop # to A.A
143AF	Push#FLoop	<see>GETPTR , Push R0 as #, Do FALSE

7.6.4 Real and Complex Numbers

1B374	PUSH%LOOP	(A -> %push) Push A as %, <see>GETPTRLOOP

7.7 Keyboard Handling

00B38	OnKeyDown?	Carry if true

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
02775	DOFLASHP	Flash pointer prologue
		3 flash bank #
		4 command #
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 RPL pointers

The contents of the following four locations are only valid after SAVPTR.

```
80E9B      AVMEM                 Free mem / 5 (5)
806F8      DSKTOP                ->Data stack
806F3      RSKTOP                ->Return stack
8076B      INTRPPTR              ->RPL runstream
                                 aka: OBUPSTART
```

8.2 Memory management pointers

```
806E9      TEMPOB                ->Beginning of TempOb area
806EE      TEMPTOP               ->End of TempOb area
80711      USEROB                ->UserOb Area (HOME)
```

8.3 Screen related

```
806D5      ADISP                 ->Stack grob
806E4      GDISP                 ->Blackboard grob
8229E      GROBSCR1              <see>SCREEN1 with GROB header
82B32      GROBSCR2              <see>SCREEN2 with GROB header
833C6      GROBSCR3              <see>SCREEN3 with GROB header
83C5A      GROBSCR4              <see>SCREEN4 with GROB header
844EE      GROBSCR5              <see>SCREEN5 with GROB header
8069C      GreyOn?               Zero if greyscale on (1)
```
If this is set to zero the interrupt system will display in greyscale, by showing each of GreyScrN/GreySoftN for one screen refresh. Note that the entries for PrintLCD use the same memory area!
```
8069D      GreyScr1              ->1st greyscale screen
806A7      GreyScr2              ->2nd greyscale screen
806B1      GreyScr3              ->3rd greyscale screen
806A2      GreySoft1             ->1st greyscale menu
806AC      GreySoft2             ->2nd greyscale menu
```

```
806B6      GreySoft4            ->3rd greyscale menu
822B2      SCREEN1              Space for one screen (2176)
                                aka: ECRAN
82B46      SCREEN2              <see>SCREEN1
833DA      SCREEN3              <see>SCREEN1
83C6E      SCREEN4              <see>SCREEN1
84502      SCREEN5              Extra screen used by <see>DBUG (2176)
806DA      VDISP                ->Display grob
                                aka: VDISP1, SYSUPSTART
806D0      VDISP2               ->Menu grob
806DF      VDISP3               ->Not displayed grob <see>VDISP
```

8.4 Annunciators

```
80F00      ANNUNCIATORS         Annunciator flags (2)
```

8.5 Save areas

```
805DB      INTRAM               Save area for the interrupt sys
                                (16)
806C0      R1[A]save            Used by PrintLCD inside the
                                interrupt system (5)
806BA      R2[A]save            <see>R1[A]save (5)
806BF      R2[S]save            <see>R1[A]save (1)
81269      SAUV_80702           Backup of <see>TEMPENV
                                aka: SavTEMPENV
8126E      SAUV_80865           Backup of <see>FIRSTCHAR
                                aka: SavFIRSTCHAR
818CF      SAUV_CHARS           Used by CHARS (31)
                                aka: SavChars
8221D      SAUV_DIVERS          Free area (128)
                                aka: SavMisc
81278      SAUV_MATRIX          Used by MTRW (40)
                                aka: SavMatrix
818F3      SAUV_REGA            Used by <see>DBUG (5)
                                aka: SavRegA
818F8      SAUV_REGB            Used by <see>DBUG (5)
                                aka: SavRegB
818FD      SAUV_REGC            Used by <see>DBUG (5)
                                aka: SavRegC
81902      SAUV_REGD            Used by <see>DBUG (5)
                                aka: SavRegD
```

81907	SAUV_REGD1	Used by <see>DBUG (5)
		aka: SavRegD1
8190C	SAUV_REGISTR	Used by <see>DBUG (101)
		aka: SavRegisters
80EF0	SAVECLK	Save of CLKON state (1)
80FB7	SAVECROSS	cursor moves in plotting (10)
805F5	SAVE_A	<see>INTRAM (16)
80608	SAVE_B	<see>INTRAM (16)
805F0	SAVE_C[A]	<see>INTRAM (5)
806C5	SAVE_BO	Save BitOffset (1)
80618	SAVE_D	<see>INTRAM (16)
8063D	SAVE_D0	<see>INTRAM (5)
806C6	SAVE_LC	Save LineCount (2)
806C8	SAVE_LN	Save LineNibs (3)
805EB	SAVE_MODES	<see>INTRAM (5)
806CB	SAVE_OFFSET	Save Window Offset (5)
80638	SAVE_PC	<see>INTRAM (5)
80628	SAVE_R0	<see>INTRAM (16)
80605	SAVE_ST	<see>INTRAM (3)
8069C	Stk0save	RSTK0 used by PrintLCD inside
		the interrupt sys (5)
806A1	Stk1save	RSTK1 <see>Stk0save (5)
806A6	Stk2save	RSTK2 <see>Stk0save (5)
806AB	Stk3save	RSTK3 <see>Stk0save (5)
806B0	Stk4save	RSTK4 <see>Stk0save (5)
806B5	Stk5save	RSTK5 <see>Stk0save (5)

8.6 System and User Flags

80F12	FLAG_SYSTEM2	Metakernel system flags (16)
		For compatibility only.
80F32	FLAG_USER2	Metakernel system flags (16)
		Dito.
80F02	SystemFlags	128 System flags (16)
80F22	UserFlags	128 User Flags (16)

8.7 Internal System Flags

Unless otherwise indicated, the description of each MASK shows what this bit means if it's set.

80EC0	SysNib1	ISysFlags 1
001C0	NoRolDA2MASK	DA2 can't be rolled up to become valid <see>SysNib1
002C0	AbbrStkMASK	Display obj types only <see>SysNib1
004C0	DA2bIsEdMASK	DA2b shows the edit line <see>SysNib1
008C0	IgnorAlmMASK	Ignore <see>ALARMSDUE in <see>GETKEY <see>SysNib1
80EC1	SysNib2	ISysFlags 2
001C1	ReqClkOnMASK	Flag for System Request of CLKON state <see>SysNib2
002C1	ServModeMASK	Server mode on <see>SysNib2
004C1	TrackMASK	New context needs to be compared with old <see>SysNib2
008C1	BadMenuMASK	Menu system corrupt <see>SysNib2
80EC2	SysNib3	ISysFlags 3
001C2	UNDOMASK	Automatic stack save <see>SysNib3
002C2	INSERTMASK	Insert/replace mode <see>SysNib3
004C2	ALGMASK	Algebraic entry mode <see>SysNib3
008C2	PRINTINGMASK	<see>SysNib3
80EC3	SysNib4	ISysFlags 4
001C3	DA2aTempMASK	DA2a temporarily valid <see>SysNib4
002C3	DA2bTempMASK	DA2b temporarily valid <see>SysNib4
004C3	DA3TempMASK	DA3 temporarily valid <see>SysNib4
008C3	RebuildMASK	Menu requires TOUCHTAB rebuild each time it is redisplayed <see>SysNib4
80EC4	SysNib5	ISysFlags 5
001C4	COMMANDMASK	CMD history enabled <see>SysNib5
002C4	BLINKMASK	Active Timer1 Int's <see>SysNib5
004C4	LOWERMASK	Lowercase keys <see>SysNib5
008C4	STKDCMASK	Decompilation for stack display (not editing) <see>SysNib5
80EC5	SysNib6	ISysFlags 6
001C5	Do1UserMASK	One-key user mode <see>SysNib6
002C5	ASuspOKMASK	Suspending current environment is allowed <see>SysNib6
004C5	BadPOLUIMASK	POL UI possibly corrupt <see>SysNib6
008C5	DA1TempMASK	DA1 temporarily valid <see>SysNib6

80EC6	SysNib7	ISysFlags 7
001C6	DA1ValidMASK	DA1 known to be valid <see>SysNib7
002C6	DA2aValdMASK	DA2a known to be valid <see>SysNib7
004C6	DA2bValdMASK	DA2b known to be valid <see>SysNib7
008C6	DA3ValidMASK	DA3 known to be valid <see>SysNib7
80EC7	SysNib8	ISysFlags 8
001C7	DA1NoChMASK	DA1 not changed <see>SysNib8
002C7	DA2aNoChMASK	DA2a not changed <see>SysNib8
004C7	DA2bNoChMASK	DA2b not changed <see>SysNib8
008C7	DA3NoChMASK	DA3 not changed <see>SysNib8
80EC8	SysNib9	ISysFlags 9
001C8	DA1BadMASK	DA1 invalid <see>SysNib9
002C8	DA2aBadMASK	DA2a invalid <see>SysNib9
004C8	DA2bBadMASK	DA2b invalid <see>SysNib9
008C8	DA3BadMASK	DA3 invalid <see>SysNib9
80EC9	SysNib10	ISysFlags 10 aka: EDITFLAG, EDITLFLAG
001C9	EDITLMASK	Edit line exists <see>SysNib10
002C9	NAppKeyMASK	Non-app keys allowed in POL <see>SysNib10
004C9	NUsrKeyMASK	Non-user keys allowed in USR mode <see>SysNib10
008C9	AppModeMASK	POL application running <see>SysNib10
80ECA	SysNib11	ISysFlags 11 aka: ParenModFLAG
001CA	ParenModMASK	Implicit parenthesized "/", "^", and "SQRT" in EQW <see>SysNib11
002CA	1PDCMASK	Partial DeCompile info will not be saved <see>SysNib11
004CA	NewEditLMASK	New one-line edit line has been created <see>SysNib11
008CA	DoStdKeyMASK	Do only standard keys <see>SysNib11
80ECB	SysNib12	ISysFlags 12
001CB	DispTimeMASK	Status bar clock may be displayed <see>SysNib12
002CB	NOP2MASK12	unused <see>SysNib12
004CB	CaseSensitiv	unused <see>SysNib12
008CB	SpeedMASK	Metakernel repeat speed <see>SysNib12
80ECC	SysNib13	ISysFlags 13
001CC	InApletMASK	Aplet running <see>SysNib13
002CC	SplitMASK	<see>SysNib13
004CC	RightMASK	<see>SysNib13

008CC	CurTknMASK	<see>SysNib13
80ECD	SizeMLDisp	
		aka: SysNib14
80ECE	SysNib15	ISysFlags 15
001CE	BadTOLUIMASK	TOL UI potentially corrupt
		<see>SysNib15
		aka: NOP1MASK15
002CE	NoAlgProcess	EVAL-> will not create a list nor
		return NOVAL <see>SysNib15
		aka: NOP2MASK15
004CE	InSimplyExpr	<see>SysNib15
		aka: NOP4MASK15
008CE	DoCreateMenu	<see>SysNib15
		aka: NOP8MASK15
80ECF	SysNib16	ISysFlags 16 (unused)
001CF	NOP1MASK16	<see>SysNib16
002CF	NOP2MASK16	<see>SysNib16
004CF	NOP4MASK16	<see>SysNib16
008CF	NOP8MASK16	<see>SysNib16
80ED0	SysNib17	ISysFlags 17 (unused)
001D0	NOP1MASK17	<see>SysNib17
002D0	NOP2MASK17	<see>SysNib17
004D0	NOP4MASK17	<see>SysNib17
008D0	NOP8MASK17	<see>SysNib17
80ED1	SysNib18	ISysFlags 18 (unused)
001D1	NOP1MASK18	<see>SysNib18
002D1	NOP2MASK18	<see>SysNib18
004D1	NOP4MASK18	<see>SysNib18
008D1	NOP8MASK18	<see>SysNib18
80ED2	SysNib19	ISysFlags 19 (unused)
001D2	NOP1MASK19	<see>SysNib19
002D2	NOP2MASK19	<see>SysNib19
004D2	NOP4MASK19	<see>SysNib19
008D2	NOP8MASK19	<see>SysNib19
80ED3	SysNib20	ISysFlags 20 (unused)
001D3	NOP1MASK20	<see>SysNib20
002D3	NOP2MASK20	<see>SysNib20
004D3	NOP4MASK20	<see>SysNib20
008D3	NOP8MASK20	<see>SysNib20

8.8 Warmstart log

80010	FAILSTK1	Warmstart log 1st (newest) entry (18)

Each entry consists of a one-nibble cause (as displayed by `WSLOG`), a 13-nibble time stamp and a 4-nibble CRC of the previous 14 nibbles.

80022	FAILSTK2	\<see>FAILSTK1 2nd entry (18)
80034	FAILSTK3	\<see>FAILSTK1 3rd entry (18)
80046	FAILSTK4	\<see>FAILSTK1 4th entry (18)

8.9 Command line management

810B6	BEG	Absolute BEGIN in CommandLine (5)
810A2	BEGIN_REL	Relative BEGIN in CommandLine (5)
810AC	BEGX	X position of BEGIN (5)
81273	CHECK_TEXTE	Checksum of cmd line (5) aka: CheckCLE
8125F	CHECK_VAL	Backup of the size of the cmd line (5)
81264	CHECK_VAL2	Checksum of the key cmd line definition (5)
80F49	CR_COUNT	# of newlines in editline (5)
80F61	CURSOR	Cursor editline position (5) aka: CURSOREPOSN
80F6E	CURSORCHR	Char under Cursor (2)
80F70	CURSORGROB	Cursor Grob Data (40)
80F6B	CURSOROFFSET	Cursor position from left of screen (2) aka: CURSORPOSN
80F66	CURSORPART	Cursor display row (5) aka: CURSORROW
80F6D	CURSORSTATE	Show cursor/char underneath (1)
80F98	CURSORX	Pxl X-Coord of Cursor (5)
80F9D	CURSORY	Pxl Y-Coord of Cursor (5)
806FD	EDITLINE	->Command line
810BB	END	Absolute END in CmdLine
810B1	ENDX	Y Position of END
810A7	END_REL	Relative END in CmdLine
810C0	SizeCLScreen	Size of CmdLine screen aka: T_ECRAN

8.10 POL variables

```
80ED4      AppCount              # of nested POLs (2)
807DE      AppCursor             ->App cursor sub-programs
807C0      AppDisplay            ->App display object
807E3      AppDoKeyOb            ->App DoKeyOb procedure for POL
807CF      AppError              ->App error handler
807CA      AppExitCond           ->App exit condition
807C5      AppKeys               ->App key assignments
807D9      AppResume             ->App resume procedure of POL
807D4      AppSuspend            ->App suspend procedure of POL
```

8.11 Topic/TOL variables

```
8086A      TopicVar1             ->generic topic var 1
8086F      TopicVar2             ->generic topic var 2
80874      TopicVar3             ->generic topic var 3
80879      TopicVar4             ->generic topic var 4
8087E      TopicVar5             ->generic topic var 5
80883      TopicVar6             ->generic topic var 6
80888      TopicVar7             ->generic topic var 7
8088D      TopicVar8             ->generic topic var 8
80892      TopicVar9             ->generic topic var 9
80897      TopicVar10            ->generic topic var 10
8089C      TopicVar11            ->generic topic var 11
808A1      TopicVar12            ->generic topic var 12
808A6      TopicVar13            ->generic topic var 13
808AB      TopicVar14            ->generic topic var 14
808B0      TopicVar15            ->generic topic var 15
808B5      TopicVar16            ->generic topic var 16
808BA      TopicVar17            ->generic topic var 17
808BF      TopicVar18            ->generic topic var 18
808C4      TopicVar19            ->generic topic var 19
808C9      TopicVar20            ->generic topic var 20
808CE      TopicVar21            ->generic topic var 21
808D3      TopicVar22            ->generic topic var 22
808D8      TopicVar23            ->generic topic var 23
```

```
808DD      TopicVar24              ->generic topic var 24
808E2      TopicVar25              ->generic topic var 25
808E7      TopicVar26              ->generic topic var 26
808EC      TopicVar27              ->generic topic var 27
808F1      TopicVar28              ->generic topic var 28
808F6      TopicVar29              ->generic topic var 29
808FB      TopicVar30              ->generic topic var 30
80900      TopicVar31              ->generic topic var 31
80905      TopicVar32              ->generic topic var 32
8090A      TopicVar33              ->generic topic var 33
8090F      TopicVar34              ->generic topic var 34
80914      TopicVar35              ->generic topic var 35
80919      TopicVar36              ->generic topic var 36
8091E      TopicVar37              ->generic topic var 37
80923      TopicVar38              ->generic topic var 38
80928      TopicVar39              ->generic topic var 39
8092D      TopicVar40              ->generic topic var 40
80932      TopicVar41              ->generic topic var 41
80937      TopicVar42              ->generic topic var 42
8093C      TopicVar43              ->generic topic var 43
80941      TopicVar44              ->generic topic var 44
80946      TopicVar45              ->generic topic var 45
8094B      TopicVar46              ->generic topic var 46
80950      TopicVar47              ->generic topic var 47
80955      TopicVar48              ->generic topic var 48
8095A      TopicVar49              ->generic topic var 49
8095F      TopicVar50              ->generic topic var 50
80964      TopicVar51              ->generic topic var 51
80969      TopicVar52              ->generic topic var 52
8096E      TopicVar53              ->generic topic var 53
80973      TopicVar54              ->generic topic var 54
80978      TopicVar55              ->generic topic var 55
8097D      TopicVar56              ->generic topic var 56
80982      TopicVar57              ->generic topic var 57
80987      TopicVar58              ->generic topic var 58
8098C      TopicVar59              ->generic topic var 59
80991      TopicVar60              ->generic topic var 60
80996      TopicVar61              ->generic topic var 61
```

```
8099B      TopicVar62              ->generic topic var 62
809A0      TopicVar63              ->generic topic var 63
809A5      TopicVar64              ->generic topic var 64
809AA      TopicVar65              ->generic topic var 65
809AF      TopicVar66              ->generic topic var 66
809B4      TopicVar67              ->generic topic var 67
809B9      TopicVar68              ->generic topic var 68
809BE      TopicVar69              ->generic topic var 69
809C3      TopicVar70              ->generic topic var 70
809C8      TopicVar71              ->generic topic var 71
809CD      TopicVar72              ->generic topic var 72
809D2      TopicVar73              ->generic topic var 73
809D7      TopicVar74              ->generic topic var 74
809DC      TopicVar75              ->generic topic var 75
809E1      TopicVar76              ->generic topic var 76
809E6      TopicVar77              ->generic topic var 77
809EB      TopicVar78              ->generic topic var 78
809F0      TopicVar79              ->generic topic var 79
809F5      TopicVar80              ->generic topic var 80
809FA      TopicVar81              ->generic topic var 81
809FF      TopicVar82              ->generic topic var 82
80A04      TopicVar83              ->generic topic var 83
80A09      TopicVar84              ->generic topic var 84
80A0E      TopicVar85              ->generic topic var 85
80A13      TopicVar86              ->generic topic var 86
80A18      TopicVar87              ->generic topic var 87
80A1D      TopicVar88              ->generic topic var 88
80A22      TopicVar89              ->generic topic var 89
80A27      TopicVar90              ->generic topic var 90
80A2C      TopicVar91              ->generic topic var 91
0005B      TopicVarNum             Number of TopicVars
80A31      TOLVar1                 ->TOL var 1
80A36      TOLVar2                 ->TOL var 2
80A3B      TOLVar3                 ->TOL var 3
80A40      TOLVar4                 ->TOL var 4
80A45      TOLVar5                 ->TOL var 5
80A4A      TOLVar6                 ->TOL var 6
80A4F      TOLVar7                 ->TOL var 7
```

```
80A54      TOLVar8                  ->TOL var 8
80A59      TOLVar9                  ->TOL var 9
80A5E      TOLVar10                 ->TOL var 10
80A63      TOLVar11                 ->TOL var 11
80A68      TOLVar12                 ->TOL var 12
80A6D      TOLVar13                 ->TOL var 13
80A72      TOLVar14                 ->TOL var 14
80A77      TOLVar15                 ->TOL var 15
80A7C      TOLVar16                 ->TOL var 16
80A81      TOLVar17                 ->TOL var 17
80A86      TOLVar18                 ->TOL var 18
80A8B      TOLVar19                 ->TOL var 19
80A90      TOLVar20                 ->TOL var 20
80A95      TOLVar21                 ->TOL var 21
80A9A      TOLVar22                 ->TOL var 22
80A9F      TOLVar23                 ->TOL var 23
80AA4      TOLVar24                 ->TOL var 24
80AA9      TOLVar25                 ->TOL var 25
80AAE      TOLVar26                 ->TOL var 26
80AB3      TOLVar27                 ->TOL var 27
80AB8      TOLVar28                 ->TOL var 28
80ABD      TOLVar29                 ->TOL var 29
80AC2      TOLVar30                 ->TOL var 30
80AC7      TOLVar31                 ->TOL var 31
80ACC      TOLVar32                 ->TOL var 32
80AD1      TOLVar33                 ->TOL var 33
80AD6      TOLVar34                 ->TOL var 34
80ADB      TOLVar35                 ->TOL var 35
80AE0      TOLVar36                 ->TOL var 36
80AE5      TOLVar37                 ->TOL var 37
80AEA      TOLVar38                 ->TOL var 38
80AEF      TOLVar39                 ->TOL var 39
80AF4      TOLVar40                 ->TOL var 40
80AF9      TOLVar41                 ->TOL var 41
80AFE      TOLVar42                 ->TOL var 42
80B03      TOLVar43                 ->TOL var 43
80B08      TOLVar44                 ->TOL var 44
80B0D      TOLVar45                 ->TOL var 45
```

```
80B12      TOLVar46              ->TOL var 46
80B17      TOLVar47              ->TOL var 47
80B1C      TOLVar48              ->TOL var 48
80B21      TOLVar49              ->TOL var 49
80B26      TOLVar50              ->TOL var 50
80B2B      TOLVar51              ->TOL var 51
80B30      TOLVar52              ->TOL var 52
80B35      TOLVar53              ->TOL var 53
80B3A      TOLVar54              ->TOL var 54
80B3F      TOLVar55              ->TOL var 55
80B44      TOLVar56              ->TOL var 56
80B49      TOLVar57              ->TOL var 57
80B4E      TOLVar58              ->TOL var 58
80B53      TOLVar59              ->TOL var 59
80B58      TOLVar60              ->TOL var 60
80B5D      TOLVar61              ->TOL var 61
80B62      TOLVar62              ->TOL var 62
80B67      TOLVar63              ->TOL var 63
80B6C      TOLVar64              ->TOL var 64
80B71      TOLVar65              ->TOL var 65
80B76      TOLVar66              ->TOL var 66
80B7B      TOLVar67              ->TOL var 67
80B80      TOLVar68              ->TOL var 68
80B85      TOLVar69              ->TOL var 69
80B8A      TOLVar70              ->TOL var 70
80B8F      TOLVar71              ->TOL var 71
80B94      TOLVar72              ->TOL var 72
80B99      TOLVar73              ->TOL var 73
80B9E      TOLVar74              ->TOL var 74
80BA3      TOLVar75              ->TOL var 75
80BA8      TOLVar76              ->TOL var 76
80BAD      TOLVar77              ->TOL var 77
80BB2      TOLVar78              ->TOL var 78
80BB7      TOLVar79              ->TOL var 79
80BBC      TOLVar80              ->TOL var 80
80BC1      TOLVar81              ->TOL var 81
80BC6      TOLVar82              ->TOL var 82
80BCB      TOLVar83              ->TOL var 83
```

```
80BD0      TOLVar84            ->TOL var 84
80BD5      TOLVar85            ->TOL var 85
80BDA      TOLVar86            ->TOL var 86
80BDF      TOLVar87            ->TOL var 87
80BE4      TOLVar88            ->TOL var 88
80BE9      TOLVar89            ->TOL var 89
80BEE      TOLVar90            ->TOL var 90
80BF3      TOLVar91            ->TOL var 91
80BF8      TOLVar92            ->TOL var 92
80BFD      TOLVar93            ->TOL var 93
80C02      TOLVar94            ->TOL var 94
80C07      TOLVar95            ->TOL var 95
80C0C      TOLVar96            ->TOL var 96
80C11      TOLVar97            ->TOL var 97
80C16      TOLVar98            ->TOL var 98
80C1B      TOLVar99            ->TOL var 99
80C20      TOLVar100           ->TOL var 100
80C25      TOLVar101           ->TOL var 101
80C2A      TOLVar102           ->TOL var 102
80C2F      TOLVar103           ->TOL var 103
80C34      TOLVar104           ->TOL var 104
80C39      TOLVar105           ->TOL var 105
80C3E      TOLVar106           ->TOL var 106
80C43      TOLVar107           ->TOL var 107
80C48      TOLVar108           ->TOL var 108
80C4D      TOLVar109           ->TOL var 109
80C52      TOLVar110           ->TOL var 110
80C57      TOLVar111           ->TOL var 111
80C5C      TOLVar112           ->TOL var 112
80C61      TOLVar113           ->TOL var 113
80C66      TOLVar114           ->TOL var 114
80C6B      TOLVar115           ->TOL var 115
80C70      TOLVar116           ->TOL var 116
80C75      TOLVar117           ->TOL var 117
80C7A      TOLVar118           ->TOL var 118
80C7F      TOLVar119           ->TOL var 119
80C84      TOLVar120           ->TOL var 120
80C89      TOLVar121           ->TOL var 121
```

```
80C8E      TOLVar122              ->TOL var 122
80C93      TOLVar123              ->TOL var 123
80C98      TOLVar124              ->TOL var 124
80C9D      TOLVar125              ->TOL var 125
80CA2      TOLVar126              ->TOL var 126
80CA7      TOLVar127              ->TOL var 127
80CAC      TOLVar128              ->TOL var 128
80CB1      TOLVar129              ->TOL var 129
80CB6      TOLVar130              ->TOL var 130
80CBB      TOLVar131              ->TOL var 131
80CC0      TOLVar132              ->TOL var 132
80CC5      TOLVar133              ->TOL var 133
80CCA      TOLVar134              ->TOL var 134
80CCF      TOLVar135              ->TOL var 135
80CD4      TOLVar136              ->TOL var 136
80CD9      TOLVar137              ->TOL var 137
80CDE      TOLVar138              ->TOL var 138
80CE3      TOLVar139              ->TOL var 139
80CE8      TOLVar140              ->TOL var 140
80CED      TOLVar141              ->TOL var 141
80CF2      TOLVar142              ->TOL var 142
80CF7      TOLVar143              ->TOL var 143
80CFC      TOLVar144              ->TOL var 144
80D01      TOLVar145              ->TOL var 145
80D06      TOLVar146              ->TOL var 146
80D0B      TOLVar147              ->TOL var 147
80D10      TOLVar148              ->TOL var 148
80D15      TOLVar149              ->TOL var 149
80D1A      TOLVar150              ->TOL var 150
80D1F      TOLVar151              ->TOL var 151
80D24      TOLVar152              ->TOL var 152
80D29      TOLVar153              ->TOL var 153
80D2E      TOLVar154              ->TOL var 154
80D33      TOLVar155              ->TOL var 155
80D38      TOLVar156              ->TOL var 156
80D3D      TOLVar157              ->TOL var 157
80D42      TOLVar158              ->TOL var 158
80D47      TOLVar159              ->TOL var 159
```

```
80D4C      TOLVar160              ->TOL var 160
80D51      TOLVar161              ->TOL var 161
80D56      TOLVar162              ->TOL var 162
80D5B      TOLVar163              ->TOL var 163
80D60      TOLVar164              ->TOL var 164
80D65      TOLVar165              ->TOL var 165
80D6A      TOLVar166              ->TOL var 166
80D6F      TOLVar167              ->TOL var 167
80D74      TOLVar168              ->TOL var 168
80D79      TOLVar169              ->TOL var 169
80D7E      TOLVar170              ->TOL var 170
80D83      TOLVar171              ->TOL var 171
80D88      TOLVar172              ->TOL var 172
80D8D      TOLVar173              ->TOL var 173
80D92      TOLVar174              ->TOL var 174
80D97      TOLVar175              ->TOL var 175
80D9C      TOLVar176              ->TOL var 176
80DA1      TOLVar177              ->TOL var 177
80DA6      TOLVar178              ->TOL var 178
80DAB      TOLVar179              ->TOL var 179
80DB0      TOLVar180              ->TOL var 180
80DB5      TOLVar181              ->TOL var 181
80DBA      TOLVar182              ->TOL var 182
80DBF      TOLVar183              ->TOL var 183
80DC4      TOLVar184              ->TOL var 184
80DC9      TOLVar185              ->TOL var 185
80DCE      TOLVar186              ->TOL var 186
80DD3      TOLVar187              ->TOL var 187
80DD8      TOLVar188              ->TOL var 188
80DDD      TOLVar189              ->TOL var 189
80DE2      TOLVar190              ->TOL var 190
80DE7      TOLVar191              ->TOL var 191
80DEC      TOLVar192              ->TOL var 192
80DF1      TOLVar193              ->TOL var 193
80DF6      TOLVar194              ->TOL var 194
80DFB      TOLVar195              ->TOL var 195
80E00      TOLVar196              ->TOL var 196
80E05      TOLVar197              ->TOL var 197
```

```
80E0A        TOLVar198                  ->TOL var 198
80E0F        TOLVar199                  ->TOL var 199
80E14        TOLVar200                  ->TOL var 200
80E19        TOLVar201                  ->TOL var 201
80E1E        TOLVar202                  ->TOL var 202
80E23        TOLVar203                  ->TOL var 203
80E28        TOLVar204                  ->TOL var 204
80E2D        TOLVar205                  ->TOL var 205
80E32        TOLVar206                  ->TOL var 206
80E37        TOLVar207                  ->TOL var 207
80E3C        TOLVar208                  ->TOL var 208
80E41        TOLVar209                  ->TOL var 209
80E46        TOLVar210                  ->TOL var 210
80E4B        TOLVar211                  ->TOL var 211
80E50        TOLVar212                  ->TOL var 212
80E55        TOLVar213                  ->TOL var 213
80E5A        TOLVar214                  ->TOL var 214
80E5F        TOLVar215                  ->TOL var 215
80E64        TOLVar216                  ->TOL var 216
000D8        TOLVarNum                  number of TOLVars
```

8.12 User interrupts

```
8600D        UserInt1                   ->User interrupt routine 1
```
This interrupt handler is called *before* the normal one. Only D1, P, Hex/Dec, CRY, SB, C.W and A.W are saved at that point.

```
86017        UserInt1g                  Copy of <see>UserInt1
```
If this address is not equal to the one in `UserInt1`, none of the two will be called.

```
86012        UserInt2                   ->User interrupt routine 2
```
This interrupt handler is called *after* the normal one, before RESTORECPU. All registers are still saved.

```
8601C        UserInt2g                  Copy of <see>UserInt2
```
If this address is not equal to the one in `UserInt2`, none of the two will be called.

8.13 UART buffering

```
80519        uart_buf_end           # of bytes in the UART buffer (2)

8051C        uart_buf_st            UART buffer offset (2)

80319        uart_buffer            UART buffer area (512)

8051B        uart_error             UART error flag (1)

8051E        uart_handshk           UART handshake (1)

8051F        uart_modes             UART mode (1)

80520        uart_parity            (1)

80521        uart_timeout           (2)
```

8.14 ROM Part Tables

```
8605E        FROMPTAB0_15           Bank switcher addresses (16*5)

860AE        FROMPTABPTR            -> <see>FROMPTAB0_15

8611D        ROMPTAB                Library table (3+n*16)
```
Header: 3 number of libraries For each library: 3 library ID 5 address 5 switch routine (0 if none) 3 000 aka: RESRAMEND, FlashROMPTAB
```
860CC        FlashROMTAB2           Bank switcher addresses (16*5)
                                    sorted by physical bank number
```

8.15 Fonts

```
81971        ArryFont               Array of used fonts (1708)
                                    aka: @FONTE
84D82        FONTE_SYSTEM           Big system font (4626)
                                    aka: SystemFont
81098        FontHeight             Height of the current font (5)
                                    aka: H_FONTE
8201D        HashArryFont           Font hash table (512)
                                    aka: TAB_FONTE
812CF        MINI_FONT              Minifont (1536)
                                    aka: MiniFont
812C3        MINI_FONT.OBJ          <see>MINI_FONT with font header
                                    aka: MiniFontObj
812AA        NB_FONTE               Number of detected fonts (5)
                                    aka: NbFont
```

8.16 Constants

The entries in this section do not denote actual memory addresses, but constants related to them.

```
00008      IRAMHOMEmsn            MSN of the IRAM base address
0001D      LOCUPSIZE              Number of variables between
                                  <see>SYSUPSTART and <see>OBUPSTART
000F4      NBMAXFONT              Maximum number of fonts
0016F      OBUPSIZE               Number of variables between
                                  <see>OBUPSTART and <see>OBUPEND
00001      mEditLExists

                                  aka: ParenModmask
0018C      SYSUPSIZE              <see>OBUPSIZE + <see>LOCUPSIZE
```

8.17 Other/Uncategorized

```
80FF1      ACCUM                  (1)
8072A      ALARMS                 ->System Alarm List (5)
80EF1      ALARMSDUE              Flags Alarm Due (1)
80EAB      ATTNFLG                Counts ON presses (5)
800E6      AccessInit             Saved value of INITEN & sALLOWINTR (2)
86051      BounceTiming           Minimum time between 2 same key
                                  press for key validation (8)
80734      CALCCXT                ->Calculator variables dir (5)
80000      CMOS                   Quick RAM corrupt check (5)
                                  aka: HARDROMEND, RAMSTART
81001      COLCOUNT               Dot Cols on line (2)
80FF3      COLWIDTH               (2)
80524      CONFRAM                RAM configuration (7)
```
Port1: 1 Status [r w s 0] 1 Size/Address Code
Port2: 1 Status [r w s 0] 1 Size/Address Code where
r=readable, w=writable, s=system RAM 2 #banks
1 ID
```
8052B      CONFTAB                RAM configuration with CRC (11)
```
4 nibbles for CRC 7 nibbles as in CONFRAM
```
8071B      CONTEXT                ->Current dir
800EB      COVERsave              Save area for G/DoCovered (10)
800E8      COVERstate             Iram state before uncovering (3)
80076      TIMEOUTCLK             ScratchPad (4)
80655      CSPEED                 CPU speed (16hz units) (5)
80FA2      CURRENTMENU            Menu ID of current menu (2)
80E69      CatalogCache           ->CAT list
86059      CatalogEntry           ->Last CAT item selected
80E6E      Clipboard              ->Clipboard
80FFA      ClkOnNib               Clock display on/off (1)
```

85FBE	CplxX	Complex number used by plotter (37)
85FE3	CplxY	`<see>`CplxX
807E8	CtlAlarm	->Control alarm data
860BD	CurRAMBank1	Backup of current RAM view 1 (5)
860C2	CurRAMBank2	Backup of current RAM view 2 (5)
860C7	CurRAMBank3	Backup of current RAM view 3 (5)
860B3	CurROMBank1	Backup of current ROM view 1 (5)
860B8	CurROMBank2	Backup of current ROM view 2 (5)
80EDC	DEPTHSAVE	Saved user stack depth (5)
86008	DIGITS	Infinite precision digits (5)
8065B	DISABLE_KBD	Keyboard handshake (1) aka: HANDSHK
8068D	DISP1CTLg	Ghost for DISP1CTL (5)
80695	DISP2CTLg	Ghost for DISP2CTL (5)
80707	DOLPENV	->DO LOOP environments
80EF3	DOUSEALARM	Flags Deactivate Curr Alarm (1)
8064A	DREND	Display Refresh Hi Bound (5)
80645	DRSTART	Display Refresh Lo Bound (5)
80FCD	DcompWidth	String Decomp Width (2)
80FFD	DelayCt	REDEYE Print time/line (2)
80F42	ELEMENT	decompile obj depth counter (2)
80FF5	ENTRWISE	(1)
80EA5	ERROR	(5)
807BB	EXITMSG	->msg set by user in EXIT word
8102B	EqPtr	Points to Curr Eqn in EqList (5)
80F44	FIRSTCHAR	offset to 1st visible (5)
80EB0	FIRSTPROC	->StartupProc Secondary (5)
80FD1	FONTCOUNT	counter (3)
80FCF	FONTHEIGHT	font-height selector (1)
80FD0	FONTWIDTH	font-width selector (1)
8072F	FSTVGERPTR	aka: VSTACK
80085	FailTime	SelfTest Fail Time (Ticks) (13)
81009	FifoByteCt	Sum of FIFO Line Counts (2)
80E73	FindPattern	->Find Pattern address
80833	FlagMBox	->Flag mailbox
81082	FlashPtrBkp	Space to create a FPTR (12)
818EE	FreeRoom	DSKTOP-RSKTOP, used by SWAPMEM (5)
80FAD	GARBSCRATCH1	Saves 1 RSTK level in G.C. (5)

80FB2	GARBSCRATCH2	Saves counter in G.C. (5)
80FFF	GCOLCOUNT	Graphics #Cols (2)
8085B	GraphContext	->Graphic Context
8030E	GraphPrtHook	(11)

90000 HARDRAMEND
aka: IRAMBEND
IRAM Home ends at #7FFFF

Appears to be an obsolete constant from the 48G, where IRAM was only 32kB big and thus ranged from #80000 to #8FFFF. The description even seems to come from the 48S!

80798	HISTORY1	-> $ with the most recent CMD history entry
8079D	HISTORY2	->2nd entry <see>HISTORY1
807A2	HISTORY3	->3rd entry <see>HISTORY1
807A7	HISTORY4	->4th (oldest) entry <see>HISTORY1
80F59	HISTORYLEVEL	which stack level is next (1)
8000A	HOMEMASK	Home Size of RAM (mask) (5)
8000F	HRAMEND	M.S.N. of size of RAM chip (1)
80851	HStackPtr	->Highlight in stack
80856	HStackTop	->How many items on stack
810E8	HashCLE	Command line hash table (360) aka: TAB_CMD
8108E	HeaderHeight	Header size in lines (5) aka: T_HEADER
80847	HiLitePtr	->Highlight in window
8065A	INITEN	Warmstart Enable flag (1)
80669	INPUTSTREAM	Key Buffer (max 15 keys). (34) aka: KEYBUFFER
80523	IOCNIB	Saves IOC in OUTUART (1)
81006	IOCsave	Save of IOC before change (1)
80654	IOSAVE	Saves HiNib of ANNCTRL (1)
00219	IRAMBSIZE	Size of <see>IRAMBUFF
800F5	IRAMBUFF	Exec Buff (code under IRAM) (537)
80127	IRAMBUFF2	<see>IRAMBUFF +50
80005	IRAMMASK	IRAM Size Config Mask (5)
8064F	IREG	Saves Interrupt History (3)
80ED6	ITEM1LINES	# display lines currently (1)
80793	ITEM1STATE	->list of lists describing stack level 1
807B1	KERMERRM	->Kermit error message aka: PDCSYMB
80FCC	KERMMODE	Kermit Mode information (1)

```
80FEB      KEYLIST              (5)
80FF0      KEYLOCK              (1)
8065C      KEYSTATE             location of kbd state (16)
86037      KSTATEVGER           KeyState for Vger Keyboard
```
From rammap.a: "(we didn't use the previous \
KEYSTATE to maintain the entry \ points)"
```
8082E      KeyOb                ->Pending key-object
81030      KeyRomPtr0           RomPtr for KeyOb (11)
8103B      KeyRomPtr1           RomPtr for MenuKey 1 (11)
81046      KeyRomPtr2           RomPtr for MenuKey 2 (11)
81051      KeyRomPtr3           RomPtr for MenuKey 3 (11)
8105C      KeyRomPtr4           RomPtr for MenuKey 4 (11)
81067      KeyRomPtr5           RomPtr for MenuKey 5 (11)
81072      KeyRomPtr6           RomPtr for MenuKey 6 (11)
80EA0      LANGUAGE             (5)
80775      LASTARG              ->1st argument saved in CK<n>
                                aka: LASTARG1
8077A      LASTARG2             ->2nd <see>LASTARG
8077F      LASTARG3             ->3rd <see>LASTARG
80784      LASTARG4             ->4th <see>LASTARG
80789      LASTARG5             ->5th <see>LASTARG
80F5A      LASTARGCOUNT         # of args saved by CK<n> (1)
80F5B      LASTARGf             Flag #Args>3 (1)
80F5C      LASTERROR            Save area for error number (5)
80FDA      LASTOP               3-state encoding of operand/
                                unary/binary (1)
80829      LASTROMWDOB          ->Last user-level ROM-WORD
                                evaluated (set by CK<n>)
80FDB      LEFTTREE             (3)
8069A      LINECOUNTg           Ghost for LINECOUNT (2)
80692      LINENIBSg            Ghost for LINENIBS (3)
80EFF      LPD_HIST             Low Power Detect History (1)
80801      LabelDef             ->How to make menu labels
8081A      LastContext          ->RRP saved for CheckContext
86047      LastKey              Last key press (2)
86049      LastKeyTime          Last key press time (8)
807F2      LastMenuDef          ->Last menu definition
8107D      LastMenuRow          (5)
8100B      LastPrntTime         Time (Upper 11 nibs) (11)
81007      LineByteCt           Line Byte Counter (2)
```

80077	LoBatTime	Flag periodic ((*)) updates (1)
80FA4	MENULEVEL	User-menu level (5)
807F7	MenuData	->Menu data for touch table
807ED	MenuDef	->Current menu definition
80824	MenuExitAct	->Menu exit action definition
8080B	MenuKeyLS	->Left-shift menu key handler
80806	MenuKeyNS	->No-shift menu key handler
80810	MenuKeyRS	->Right-shift menu key handler
81026	MenuRow	(5)
807FC	MenuRowAct	->Prev/Next action definition
81093	NB_LIGNE	Size of the stack's screen in lines (5) aka: StackHeight
80058	NEXTIRQ	Time at next Timer2 int. (13)
80EF4	NOALARMSRV	Flags Disable Alarm Service (1)
80FD4	NODECOUNT	expr-tree node count (3)
8073E	NOTESCXT	->"notes" directory (5)
80FD7	OBTREELEN	object length (3)
80FA9	OLDMENU	Saves previous menu number (2)
80642	SAVE_OR	aka: ORghost
80770	OSAVE	
80E7D	ObjectU1	->Updatable object 1
80E82	ObjectU2	->Updatable object 2
80E87	ObjectU3	->Updatable object 3
80E8C	ObjectU4	->Updatable object 4
80E91	ObjectU5	->Updatable object 5 aka: OBUPEND
80FAC	PADCOUNT	Indentation count for decomp (1)
80FC1	PADJSAVE1	Status save in PTRADJUST (1)
80FC2	PADJSAVE2	RSTK save in PTRADJUST (10)
807B6	PAINTTREE	->hxs of "textbook-mode" graphics
80FF6	PARENCOUNT	(2)
80FE1	PARENTTREE	(3)
80EF2	PASTDUE	Flags Past Due Alarm (1)
807AC	PDCHXS	->hxs map of outermost symbolic
81016	PFIFO	FIFO Buffer (16)
80739	PGMCXT	->programming dir (5)
8068B	POPPEDKEY	Last Key from POPKEY (2)

80536	PORT0EOS	(5)
8053B	PORT1EOS	(5)
80540	PORT2EOS	(5)
80FE4	PRECSTACK	Op Precedence textbook entry (7)
800E2	Port1CRC	CRC for Device in Port1 (4)
800E1	PortStat	Copy of CARDSTAT Nib (1)
8083D	ProgMBox	->Program mailbox
81003	PrtStatus	CPU Status Bits et al. (3)
80E96	RAMEND	->End of RAM aka: SYSNOUPSTART
8611C	RESRAMEND0	End of statically reserved RAM
80FDE	RIGHTTREE	(3)
80EE1	RNSEED	Random number seed (15)
80716	ROMPARTS	->RomParts Area
85F94	RealX	Real number used by plotter (21)
85FA9	RealY	<see>RealX
80E78	ReplacePatte	->Replace pattern
80815	ReviewKey	->Review-key definition
80652	SEMAPH	Saves control byte for IREG (2)
80F4E	STACKNUM	ref. number of 1st visible (5)
80720	STOPSIGN	(5)
80FF8	STRETCHCOUNT	(2)
812B4	SWITCH	Used by the Memory Manager (15)
800D4	SW_ETime	Stopwatch Elapsed Time Ticks (13)
800BE	SW_Image	"HH:MM:SS:ss" Stopwatch (22)
812A0	SizeLine	Size of one line of text aka: T_LIGNE
80078	StartTime	SelfTest Start Time (Ticks) (13)
80FAB	T1COUNT	Decremented by srvc_timer1 (1)
80702	TEMPENV	->LAM environments (5)
80092	TESTMSG	SelfTest Msg Buffer (44)
80065	TIMECRC	CRC CheckSum for NEXTIRQ (4)
80069	TIMExmit	Time at scheduled timeout (13) aka: TIMEOUT
80F53	TOPLINE	Editline-segment which appears first on the screen (5)
8070C	TOUCHTAB	(5)
8109D	TYPE_HEADER	Type of header (5)
8125A	T_BLOC	Size of a HashCLE block (5)

812A5	T_LARGEUR	Width of the current screen in nibbles (5) aka: WidthScreen
80842	Title	->Home Title
8081F	TrackAct	->Action when CONTEXT changes
80725	UserKeys	->User key assignments (5)
812AF	VERIF_CARD	
0000C	VGERPTRCT	
80ED7	VIEWLEVEL	stack element currently viewed (5)
80838	ViewMBox	->View mailbox
8084C	WindowPtr	->Item at bottom of window
80FFB	XmitSrcvTOut	XMIT/SRECV timeout (2)
80743	apletPTR	->current aplet (5)
80748	funcPTR	->current func instance (5)
86026	has_font_f_s	Tells if the Decompiler has found a special font character (2)
8078E	leeway	->hxs which will be GC'ed in a very-low-memory condition
86028	misc1_f_s	(5)
8602D	misc2_f_s	(5)
86032	misc3_f_s	(5)
86021	nb_line_f_s	Number of line created during decompilation (FSTR3) (5)
80766	otherPTR	->current "other" instance (5)
80752	paramPTR	->current param instance (5)
8074D	polarPTR	->current polar instance (5)
80757	seqPTR	->current sequence instance (5)
80761	solvePTR	->current solve instance (5)
8075C	statPTR	->current stat instance (5)

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

```
3D5B2     xCMDAPPLY
3CB4E     xDER
3D6AE     xEVAL>
3D19D     xFCNAPPLY
3CD50     xINTEGRAL
3D7A7     xNOEVAL>
3D017     xWHERE
3F7E7     !MATTRNnc
3F80B     !REDIMTEMP
3F80C     !REDIMUSER
26A01     CtlAlarm!
269E8     DoCAlarmKey
15EAD     DropSysObs
20BBA     EVALCRUNCH
0A021     InitEnab
123E8     lbrac
3F657     LINECHANGE
3F397     NEWINDEP
35798     POLErrorTrap
028FC     PRLG
347CC     PTYPE>PINFO
0131D     SavPtrTime*
3F73C     SYMBNUMSOLVE
1708D     WaitTbz0
09C41     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.

00001	mEditLExists	002C1	ServModeMASK	004CE	NOP4MASK15
00001	ParenModmask	002C2	INSERTMASK	004CE	InSimplyExpr
00008	IRAMHOMEmsn	002C3	DA2bTempMASK	004CF	NOP4MASK16
0000C	VGERPTRCT	002C4	BLINKMASK	004D0	NOP4MASK17
0001D	LOCUPSIZE	002C5	ASuspOKMASK	004D1	NOP4MASK18
0005B	TopicVarNum	002C6	DA2aValdMASK	004D2	NOP4MASK19
000D8	TOLVarNum	002C7	DA2aNoChMASK	004D3	NOP4MASK20
000F4	NBMAXFONT	002C8	DA2aBadMASK	008C0	IgnorAlmMASK
0016F	OBUPSIZE	002C9	NAppKeyMASK	008C1	BadMenuMASK
0018C	SYSUPSIZE	002CA	1PDCMASK	008C2	PRINTINGMASK
001C0	NoRolDA2MASK	002CB	NOP2MASK12	008C3	RebuildMASK
001C1	ReqClkOnMASK	002CC	SplitMASK	008C4	STKDCMASK
001C2	UNDOMASK	002CE	NOP2MASK15	008C5	DA1TempMASK
001C3	DA2aTempMASK	002CE	NoAlgProcess	008C6	DA3ValidMASK
001C4	COMMANDMASK	002CF	NOP2MASK16	008C7	DA3NoChMASK
001C5	Do1UserMASK	002D0	NOP2MASK17	008C8	DA3BadMASK
001C6	DA1ValidMASK	002D1	NOP2MASK18	008C9	AppModeMASK
001C7	DA1NoChMASK	002D2	NOP2MASK19	008CA	DoStdKeyMASK
001C8	DA1BadMASK	002D3	NOP2MASK20	008CB	SpeedMASK
001C9	EDITLMASK	004C0	DA2bIsEdMASK	008CC	CurTknMASK
001CA	ParenModMASK	004C1	TrackMASK	008CE	NOP8MASK15
001CB	DispTimeMASK	004C2	ALGMASK	008CE	DoCreateMenu
001CC	InApletMASK	004C3	DA3TempMASK	008CF	NOP8MASK16
001CE	NOP1MASK15	004C4	LOWERMASK	008D0	NOP8MASK17
001CE	BadTOLUIMASK	004C5	BadPOLUIMASK	008D1	NOP8MASK18
001CF	NOP1MASK16	004C6	DA2bValdMASK	008D2	NOP8MASK19
001D0	NOP1MASK17	004C7	DA2bNoChMASK	008D3	NOP8MASK20
001D1	NOP1MASK18	004C8	DA2bBadMASK	00B38	OnKeyDown?
001D2	NOP1MASK19	004C9	NUsrKeyMASK	00C3A	FLUSHKEYS
001D3	NOP1MASK20	004CA	NewEditLMASK	00C3A	FLUSH
00219	IRAMBSIZE	004CB	CaseSensitiv	0114C	DisableIntr
002C0	AbbrStkMASK	004CC	RightMASK	0131D	SavPtrTime*

01F83	CLCD10	031D9	NDUP	03EE1	#AND
02775	DOFLASHP	03223	SWAP	03EF2	#XOR
028FC	PRLG	03244	DROP	03F12	#*
02911	DOBINT	03258	2DROP	03F47	#/
02933	DOREAL	0326E	NDROP	03FDB	TYPEREAL
02955	DOEREL	03295	ROT	03FE5	TYPECMP
02977	DOCMP	032C2	OVER	03FEF	TYPELIST
0299D	DOECMP	032E2	PICK	03FF9	TYPEIDNT
029BF	DOCHAR	03325	ROLL	04003	TYPECOL
029E8	DOARRY	0339E	UNROLL	0400D	TYPESYMB
02A0A	DOLNKARRY	03672	GPOverWrALp	04017	TYPERRP
02A2C	DOCSTR	0371D	GETATELN	04021	TYPELAM
02A4E	DOHSTR	03A81	TRUE	0402B	TYPEEREL
02A4E	DOHXS	03A86	PUSHA	04035	TYPEEXT
02A74	DOLIST	03AC0	FALSE	04049	any
02A96	DORRP	03ADA	XOR	04049	ZERO
02AB8	DOSYMB	03AF2	NOT	04049	BINT0
02ADA	DOEXT	03B2E	EQ	04053	real
02AFC	DOTAG	03B46	AND	04053	BINT1
02B1E	DOGROB	03B75	OR	04053	MEMERR
02B40	DOLIB	03B97	EQUAL	04053	ONE
02B62	DOBAK	03C64	TYPE	0405D	BINT2
02B88	DOEXT0	03CA6	#0=	0405D	TWO
02BAA	DOEXT1	03CC7	#0<>	0405D	cmp
02BAA	DOACPTR	03CE4	#<	04067	str
02BCC	DOEXT2	03D19	#=	04067	BINT3
02BEE	DOEXT3	03D4E	#<>	04067	THREE
02C10	DOEXT4	03D83	#>	04071	FOUR
02D9D	DOCOL	03DBC	#+	04071	BINT4
02DCC	DOCODE	03DE0	#-	04071	arry
02E48	DOIDNT	03DEF	#1+	0407B	FIVE
02E6D	DOLAM	03E0E	#1-	0407B	BINT5
02E92	DOROMP	03E2D	#2+	0407B	list
03019	SKIPOB	03E4E	#2-	04085	id
0312B	SEMI	03E6F	#2*	04085	BINT6
0314C	DEPTH	03E8E	#2/	04085	SIX
03188	DUP	03EB1	#NOT	04085	idnt
031AC	2DUP	03ED0	#OR	0408F	lam

0408F	BINT7	04111	BINT20	041BB	BINT37
0408F	SEVEN	04111	TWENTY	041C5	THIRTYEIGHT
04099	seco	0411B	TWENTYONE	041C5	BINT38
04099	EIGHT	0411B	BINT21	041CF	BINT39
04099	BINT8	04125	BINT22	041CF	THIRTYNINE
040A3	symb	04125	TWENTYTWO	041D9	FOURTY
040A3	BINT9	0412F	BINT23	041D9	BINT40
040A3	NINE	0412F	TWENTYTHREE	041D9	FORTY
040AD	sym	04139	BINT24	041E3	FORTYONE
040AD	BINT10	04139	TWENTYFOUR	041E3	BINT41
040AD	TEN	04143	BINT25	041ED	BINT42
040B7	hxs	04143	TWENTYFIVE	041ED	FORTYTWO
040B7	BINT11	0414D	REALSYM	041F7	BINT43
040B7	ELEVEN	0414D	BINT26	041F7	FORTYTHREE
040C1	BINT12	0414D	TWENTYSIX	04201	BINT44
040C1	grob	04157	TWENTYSEVEN	04201	FORTYFOUR
040C1	TWELVE	04157	BINT27	0420B	FORTYFIVE
040CB	TAGGED	04161	TWENTYEIGHT	0420B	BINT45
040CB	BINT13	04161	BINT28	04215	BINT46
040CB	THIRTEEN	0416B	BINT29	04215	FORTYSIX
040D5	EXT	0416B	TWENTYNINE	0421F	BINT47
040D5	BINT14	04175	BINT30	0421F	FORTYSEVEN
040D5	FOURTEEN	04175	REALEXT	0421F	rrp
040D5	unitob	04175	THIRTY	04229	BINT48
040DF	BINT15	0417F	THIRTYONE	04229	FORTYEIGHT
040DF	rompointer	0417F	BINT31	04233	BINT49
040DF	FIFTEEN	04189	BINT32	04233	FORTYNINE
040E9	REALOB	04189	THIRTYTWO	0423D	BINT50
040E9	BINT16	04193	BINT33	0423D	FIFTY
040E9	SIXTEEN	04193	THIRTYTHREE	04247	2STR
040F3	SEVENTEEN	0419D	THIRTYFOUR	04247	FIFTYONE
040F3	2REAL	0419D	BINT34	04247	BINT51
040F3	REALREAL	041A7	BINT35	04251	BINT52
040F3	BINT17	041A7	THIRTYFIVE	04251	THREEFOUR
040FD	EIGHTEEN	041B1	THIRTYSIX	04251	FIFTYTWO
040FD	BINT18	041B1	TTHIRTYSIX	0425B	BINT53
04107	NINETEEN	041B1	BINT36	0425B	THREEFIVE
04107	BINT19	041BB	THIRTYSEVEN	0425B	STRLIST

0425B	FIFTYTHREE	042F1	SIXTYEIGHT	04391	FIVEFOUR
04265	FIFTYFOUR	042F1	BINT68	0439B	BINT85
04265	BINT54	042FB	BINT69	0439B	2LIST
0426F	BINT55	042FB	FOURFIVE	043A5	BINT86
0426F	FIFTYFIVE	042FB	ARRYLIST	043A5	LISTID
0426F	#THREESEVEN	04305	SEVENTY	043A5	FIVESIX
04279	BINT56	04305	ARRYID	043AF	BINT87
04279	FIFTYSIX	04305	BINT70	043AF	LISTLAM
04283	FIFTYSEVEN	0430F	SEVENTYONE	043B9	EIGHTYEIGHT
04283	BINT57	0430F	BINT71	043B9	BINT88
0428D	FIFTYEIGHT	04319	BINT72	043C3	BINT89
0428D	BINT58	04319	SEVENTYTWO	043C3	EIGHTYNINE
04297	BINT59	04323	BINT73	043CD	BINT90
04297	FIFTYNINE	04323	SEVENTYTHREE	043CD	NINETY
042A1	SIXTY	0432D	ARRYSYM	043D7	BINT_91d
042A1	BINT60	0432D	BINT74	043D7	BINT91
042AB	SIXTYONE	0432D	SEVENTYFOUR	043E1	BINT92
042AB	BINT61	04337	SEVENTYFIVE	043E1	NINETYTWO
042B5	BINT62	04337	BINT75	043EB	NINETYTHREE
042B5	SIXTYTWO	04341	SEVENTYSIX	043EB	BINT93
042BF	BINT63	04341	BINT76	043F5	NINETYFOUR
042BF	BINT3Fh	0434B	BINT77	043F5	BINT94
042BF	SIXTYTHREE	0434B	SEVENTYSEVEN	043FF	BINT95
042C9	SIXTYFOUR	04355	SEVENTYEIGHT	043FF	NINETYFIVE
042C9	BINT64	04355	BINT78	04409	idntany
042C9	BINT40h	0435F	SEVENTYNINE	04409	BINT_96d
042C9	YHI	0435F	BINT79	04409	BINT96
042D3	SIXTYFIVE	04369	EIGHTY	04413	BINT97
042D3	BINT_65d	04369	LISTOB	04413	IDREAL
042D3	BINT65	04369	BINT80	0441D	BINT98
042D3	ARRYREAL	04373	LISTREAL	0441D	NINETYEIGHT
042DD	FOURTWO	04373	EIGHTYONE	0441D	IDCMP
042DD	ARRYCMP	04373	BINT81	04427	NINETYNINE
042DD	BINT66	0437D	BINT82	04427	BINT99
042DD	SIXTYSIX	0437D	LISTCMP	04431	IDARRY
042E7	FOURTHREE	04387	BINT83	04431	ONEHUNDRED
042E7	BINT67	04387	FIVETHREE	04431	BINT100
042F1	2ARRY	04391	BINT84	0443B	IDLIST

0443B SIXFIVE	04549 BINT_128d	04657 BINT155
0443B BINT101	04549 BINT128	04661 BINT156
04445 BINT102	04553 BINT129	0466B BINT157
0444F BINT103	0455D BINT130d	04675 BINT158
04459 BINT104	0455D XHI-1	04675 SYMBUNIT
04463 BINT105	0455D BINT_130d	0467F backup
0446D BINT106	0455D BINT130	0467F BINT159
04477 BINT107	04567 XHI	04689 BINT160
04481 BINT108	04567 BINT131	04689 SYMOB
0448B BINT109	04567 BINT131d	04693 SYMREAL
04495 BINT110	04567 BINT_131d	04693 BINT161
0449F char	04571 BINT132	0469D BINT162
0449F BINT111	0457B BINT133	0469D SYMCMP
044A9 lamany	04585 BINT134	046A7 BINT163
044A9 BINT112	0458F BINT135	046A7 SYMSTR
044B3 BINT113	04599 BINT136	046B1 SYMARRY
044B3 LAMREAL	045A3 BINT137	046B1 BINT164
044BD BINT114	045AD BINT138	046BB BINT165
044BD BINT_114	045B7 BINT139	046BB SYMLIST
044C7 BINT_115d	045C1 BINT140	046C5 SYMID
044C7 BINT115	045CB BINT141	046C5 BINT166
044D1 BINT_116d	045D5 BINT142	046CF SYMLAM
044D1 BINT116	045DF library	046CF BINT167
044DB BINT_117d	045DF BINT143	046D9 BINT168
044DB BINT117	045E9 BINT144	046E3 BINT169
044E5 LAMLIST	045F3 SYMBREAL	046E3 SYMSYMB
044E5 BINT118	045F3 BINT145	046ED SYMSYM
044EF BINT119	045FD SYMBCMP	046ED BINT170
044F9 BINT120	045FD BINT146	046F7 BINT171
04503 BINT121	04607 BINT147	04701 BINT172
0450D BINT122	04611 BINT148	0470B BINT173
0450D BINT_122d	0461B BINT149	04715 SYMEXT
04517 BINT123	04625 BINT150	04715 BINT174
04521 BINT124	0462F BINT151	0471F BINT175
0452B BINT125	04639 BINT152	04729 BINT176
04535 BINT126	04643 BINT153	04733 BINT177
0453F BINT127	0464D SYMBSYM	04733 HXSREAL
04549 BINT80h	0464D BINT154	0473D BINT178

04747 BINT179	0489B BINT213	049E5 BINT246
04751 BINT180	048A5 BINT214	049EF BINT247
0475B BINT181	048AF BINT215	049F9 BINT248
04765 BINT182	048B9 BINT216	04A03 BINT249
0476F BINT183	048C3 BINT217	04A0D BINT250
04779 BINT184	048CD BINT218	04A17 BINT251
04783 BINT185	048D7 BINT219	04A21 BINT252
0478D BINT186	048E1 BINT220	04A2B BINT253
04797 2HXS	048EB BINT221	04A35 BINT254
04797 BINT187	048F5 BINT222	04A3F BINT255d
047A1 BINT188	048FF BINT223	04A3F BINT255
047AB BINT189	04909 BINT224	04A49 REALOBOB
047B5 BINT190	04913 BINT225	04A49 BINT256
047BF BINT191	04913 EXTREAL	04A49 FSTFLOATROM#
047C9 BINT192	0491D BINT226	04A53 TurnOff
047C9 BINTC0h	04927 BINT227	04F92 CHECKKEY
047D3 BINT193	04931 BINT228	04F9E GETTOUCH
047DD BINT194	0493B BINT229	05051 REPKEY?
047E7 BINT195	04945 BINT230	054F8 ERROR@
047F1 BINT196	0494F BINT231	05520 ERRORSTO
047FB BINT197	04959 BINT232	05545 ERRORCLR
04805 BINT198	04963 BINT233	05550 DROPNULL$
0480F BINT199	0496D EXTSYM	05569 TWODROPNULL$
04819 BINT200	0496D UNITSYM	05576 GETTHEMESG
04823 BINT201	0496D BINT234	05599 JstGETTHEMSG
0482D BINT202	04977 BINT235	05599 JstGetTHEMESG
04837 BINT203	04981 BINT236	05605 GETEXITMSG
04841 BINT204	0498B BINT237	05635 EXITMSGSTO
04841 2GROB	04995 BINT238	0565C ERRSET
0484B BINT205	04995 2EXT	056A2 ABORT
04855 BINT206	0499F BINT239	056B6 ERRTRAP
0485F BINT207	049A9 BINT240	056CF ERRJMP
04869 TAGGEDANY	049A9 ROMPANY	05763 SETLBERR
04869 BINT208	049B3 BINT241	0576F SETMEMERR
04873 BINT209	049BD BINT242	057CF SETROMPERR
0487D BINT210	049C7 BINT243	05821 ATTNFLGCLR
04887 BINT211	049D1 BINT244	05842 CARCOMP
04891 BINT212	049DB BINT245	058A6 CAR$

058FC	GETPTRLOOP	06E19	POP#	07C70	ABND
0590C	CDRCOMP	06E2F	TOTEMPOB	07CA9	BIND
05925	CDR$	06F73	SAVPTR	07CBD	DOBIND
05947	&$	0731C	INTEMNOTREF?	07D7E	GETLAM
059D3	&COMP	0765C	NOP	07DC2	PUTLAM
05A0F	>H$	07665	'	07E87	OFFSRRP
05A7A	>HCOMP	076B9	'R	07EE2	TOSRRP
05AA2	>T$	07734	'REVAL	07F66	ONSRRP?
05AAE	>TCOMP	0775C	EVAL	07FBD	MAKERRP
05BFE	::N	07778	>R	080E7	@LAM
05C12	{}N	07790	RDROP	081D3	@
05C26	SYMBN	077AA	COLA	0852B	COMPILEID
05C3A	EXTN	077EB	R@	08530	COMPROMID
05C68	INNERCOMP	077F8	R>	08638	STOLAM
05D28	NULL$?	0789C	RPITE	08644	STO
05D70	NULLCOMP?	078D6	RPIT	0876D	#>ROMPTR
05D8E	NULLHXS	07903	?SKIP	087B6	ROMPTR@
05D98	NULL$	07903	NOT_IT	08AEC	BAKNAME
05DA2	NULL{}	07926	SKIP	08BF2	RAM-WORDNAME
05DAC	NULLSYMB	0797B	BEGIN	08C35	LASTRAM-WORD
05DB6	NULL::	07984	AGAIN	08C85	PREVRAM-WORD
05DEF	LEN$	079A1	UNTIL	08ED9	REPLACE
05E34	LENCOMP	079BE	REPEAT	08F9C	CREATE
05E6F	NTHELCOMP	079C7	WHILE	0952D	PURGE
05EEC	SUB$	079FA	INDEX@	095D2	ROMPTR>#
05FDA	SUBCOMP	07A22	ISTOP@	0960E	CONTEXT!
060BB	OSIZE	07A31	JINDEX@	09660	CONTEXT@
060FD	OCRC	07A3D	JSTOP@	09698	SYSCONTEXT
06185	#>HXS	07A49	INDEXSTO	09698	HOMEDIR
0620A	CHR>#	07A6E	ISTOPSTO	096DA	SYSRRP?
0622E	#>CHR	07A86	JINDEXSTO	099D6	XSTOp
0626C	CHANGETYPE	07A9B	JSTOPSTO	09BA1	XSTOp0
062CE	$>ID	07B0D	LOOP	09BBF	XRCLp
063C9	ID>$	07B7E	+LOOP	09C19	XRCLp0
0642E	%>C%	07B9C	ZERO_DO	09C3C	PORTOADDR
06530	C%>%	07BA7	ONE_DO	09C41	X@
06742	GARBAGE	07BB4	#1+_ONE_DO	09D31	XPURGEp
06761	MEM	07BD0	DO	09D6D	XPURGEp0

09DA5	BAK>OB	0CDB0	FIVEROLL	0D04B	#1+ROLL
09DBB	HOME>BAK	0CDDA	6ROLL	0D05D	get1
09FB0	BAK>HOME	0CDDA	SIXROLL	0D070	#2+ROLL
0A021	InitEnab	0CE14	EIGHTROLL	0D084	#-UNROLL
0B964	dirstrucchk	0CE14	8ROLL	0D096	#+UNROLL
0B9C9	loopdirck	0CE43	SEVENROLL	0D0AB	#1+UNROLL
0C476	TrueTrue	0CE43	7ROLL	0D0BD	#2+UNROLL
0C476	completed	0CE71	9ROLL	0D0D8	DUPUNROT
0C48A	failed	0CE8A	FOURUNROLL	0D0D8	SWAPOVER
0C558	!*triand	0CE8A	XYZW>WXYZ	0D16A	1PUTLAM
0C5DD	tok8cktrior	0CE8A	4UNROLL	0D16F	1GETLAM
0C5E9	tok8trior	0CEB0	5UNROLL	0D174	2PUTLAM
0CC9D	RSWAP	0CEB0	FIVEUNROLL	0D179	2GETLAM
0CCC7	XYZ>YXZ	0CEE6	6UNROLL	0D17E	3PUTLAM
0CCC7	ROTSWAP	0CEE6	SIXUNROLL	0D183	3GETLAM
0CCEE	XYZ>ZY	0CF16	XYZ>Z	0D188	4PUTLAM
0CCEE	ROTDROPSWAP	0CF16	UNROT2DROP	0D18D	4GETLAM
0CD01	XYZ>YZ	0CF16	ROTROT2DROP	0D192	5PUTLAM
0CD01	ROTDROP	0CF28	4UNROLL3DROP	0D197	5GETLAM
0CD13	XYZ>ZYX	0CF28	XYZW>W	0D19C	6PUTLAM
0CD13	UNROTSWAP	0CF3A	2RDROP	0D1A1	6GETLAM
0CD13	SWAPROT	0CF4C	3RDROP	0D1A6	7PUTLAM
0CD2B	3DROP	0CF5E	#-PICK	0D1AB	7GETLAM
0CD2B	XYZ>	0CF70	#+PICK	0D1B0	8PUTLAM
0CD30	4DROP	0CF8F	#1+PICK	0D1B5	8GETLAM
0CD30	XYZW>	0CFAA	#2+PICK	0D1BA	9PUTLAM
0CD35	5DROP	0CFBE	#3+PICK	0D1BF	9GETLAM
0CD41	6DROP	0CFCD	#4+PICK	0D1C4	10PUTLAM
0CD4D	7DROP	0CFDE	3PICK	0D1C9	10GETLAM
0CD73	XY>Y	0CFE3	4PICK	0D1CE	11PUTLAM
0CD73	SWAPDROP	0CFE8	5PICK	0D1D3	11GETLAM
0CD84	3UNROLL	0CFED	6PICK	0D1D8	12PUTLAM
0CD84	UNROT	0CFF2	7PICK	0D1DD	12GETLAM
0CD84	XYZ>ZXY	0CFF7	8PICK	0D1E2	13PUTLAM
0CD93	XYZW>YZWX	0CFFC	9PICK	0D1E7	13GETLAM
0CD93	4ROLL	0D001	10PICK	0D1EC	14PUTLAM
0CD93	FOURROLL	0D024	#-ROLL	0D1F1	14GETLAM
0CDB0	5ROLL	0D036	#+ROLL	0D1F6	15PUTLAM

0D1FB	15GETLAM	0D5F5 NOT?SEMI	0DCB7 DUPTYPESYMB?
0D200	16PUTLAM	0D604 ?SEMI	0DCBC TYPESYMB?
0D205	16GETLAM	0D621 ITE_DROP	0DCC6 DUPTYPECOL?
0D20A	17PUTLAM	0D636 COLA_EVAL	0DCC6 DTYPECOL?
0D20F	17GETLAM	0D657 COLARPITE	0DCCB TYPECOL?
0D214	18PUTLAM	0D6A1 ITE	0DCD5 DUPTYPEGROB?
0D219	18GETLAM	0D6B2 2'RCOLARPITE	0DCDA TYPEGROB?
0D21E	19PUTLAM	0D752 ticR	0DCE4 DTYPELIST?
0D223	19GETLAM	0D7E5 EXPAND	0DCE4 DUPTYPELIST?
0D228	20PUTLAM	0D863 CACHE	0DCE9 TYPELIST?
0D22D	20GETLAM	0DA21 DUMP	0DCF3 DUPTYPETAG?
0D232	21PUTLAM	0DB30 ROM-WORD?	0DD02 DUPTYPEEXT?
0D237	21GETLAM	0DB7B 2SWAP	0DD07 TYPEEXT?
0D23C	22PUTLAM	0DB9A DUPTYPECHAR?	0DD8A GPPushTLoop
0D241	22GETLAM	0DB9F TYPECHAR?	0DDA3 GPPushFLoop
0D3A1	GETLAMPAIR	0DBA9 DUPTYPEIDNT?	0DDBC OVER#=
0D435	#=case	0DBAE TYPEIDNT?	0DDD4 DROPTRUE
0D445	OVER#=case	0DC30 DUPTYPELAM?	0DDDD DROPFALSE
0D45A	DUP#0=case	0DC35 TYPELAM?	0DDE6 TYPERARRY?
0D45F	#0=case	0DC3F DUPTYPEBINT?	0DE01 TYPECARRY?
0D471	DUP#0=csedrp	0DC44 TYPEBINT?	0DE11 DUP#0=
0D483	EQcasedrop	0DC4E DUPTYPEHSTR?	0DE34 #3=
0D49C	#=casedrop	0DC53 TYPEHSTR?	0DE45 #2=
0D4B1	NOTcasedrop	0DC5D DUPTYPECSTR?	0DE52 #1=
0D4C0	casedrop	0DC5D DTYPECSTR?	0DE61 #1<>
0D4D9	NOTcase2drop	0DC62 TYPECSTR?	0DE70 DUP#1=
0D4E8	case2drop	0DC6C DUPTYPEREAL?	0DE7F DUP#0<>
0D4FC	EQcase	0DC6C DTYPEREAL?	0DE9A SWAP&$
0D4FC	OVEREQcase	0DC71 TYPEREAL?	0DF21 !append$
0D514	caseDROP	0DC7B DUPTYPECMP?	0DF4B !!append$
0D529	NOTcaseDROP	0DC80 TYPECMP?	0E065 #MIN
0D539	case2DROP	0DC8A DTYPEARRY?	0E071 #MAX
0D54D	NOTcase2DROP	0DC8A DUPTYPEARRY?	0E0A6 #-#2/
0D55C	case	0DC8F TYPEARRY?	0E0E0 DROPZERO
0D576	NOTcase	0DC99 DUPTYPEROMP?	0E0F9 2DROP00
0D585	IT	0DC9E TYPEROMP?	0E133 #3-
0D5CB	popflag	0DCA8 DUPTYPERRP?	0E151 #3+
0D5E1	#0=?SEMI	0DCAD TYPERRP?	0E156 #4+

0E15B	#5+	
0E16A	#8+	
0E287	DROPSWAP	
0E2A1	XYZ>Y	
0E2A1	DROPSWAPDROP	
0E2A1	ROT2DROP	
0E2C2	SWAPDUP	
0E2F0	ROTDUP	
0E30F	SWAP#-	
0E322	DROPDUP	
0E336	DUPLEN$	
0E350	#+DUP	
0E3AB	SWAPDROPDUP	
0E3C6	SWAPDROPSWAP	
0E3C6	XYZ>ZX	
0E3C6	UNROTDROP	
0E416	2DUP#<	
0E430	2DUP#=	
0E44C	2DUP#>	
0E466	DUP#1+	
0E47F	SWP1+	
0E47F	SWAP#1+	
0E4C1	DROPONE	
0E4D3	RDROPCOLA	
0E4E8	COLACOLA	
0E501	COLAcase	
0E537	ORcase	
0E5DC	?>ROMPTR	
0E5FF	?ROMPTR>	
0E767	CKREAL	
0E8C2	Embedded?	
0E996	ZEROZERO	
0ED0F	POSCOMP	
0ED6E	AttnPOSCOMP	
0EDF0	Lookup	
0EE09	Lookup.1	
0EE54	POS$	
0EE54	POSCHR	

0EE60	POSCHRREV	
0EE60	POS$REV	
0F15A	NEXTCOMPOB	
0F328	$jGPOvrWrFLp	
0F336	$jGPOvrWrTLp	
0F39F	$jOvrWrF/TLp	
0F3A6	$jOvrWrFLoop	
0F3AD	$jOvrWrT/FLp	
0F3B4	$jOvrWrTLoop	
0F40F	$jPshF/TLoop	
0F439	$jPshT/FLoop	
0F44E	2DROPFALSE	
0F462	INCOMPDROP	
0F476	NTHCOMPDROP	
0F48A	APPEND_SPACE	
0F49E	7UNROLL	
0F4B2	#0=UNTIL	
0F4DF	DUP@	
0F4F3	DUPROMPTR@	
0F507	#=ITE	
0F51B	INNERDUP	
0F52F	NOTAND	
0F543	TOTEMPSWAP	
0F557	ROT2DUP	
0F56B	ROTAND	
0F57F	ROTOVER	
0F593	DUPDUP	
0F5A7	OVERDUP	
0F5BB	COERCEDUP	
0F5CF	UNROTDUP	
0F5E3	4UNROLLDUP	
0F5F7	NTHCOMDDUP	
0F60B	OVERUNROT	
0F60B	OVERSWAP	
0F61F	ROLLSWAP	
0F633	NULL$SWAP	
0F647	SUB$SWAP	
0F65B	%MAXorder	

0F679	?SKIPSWAP	
0F68D	1ABNDSWAP	
0F6A6	ROT+SWAP	
0F6A6	ROT#+SWAP	
0F6BF	4PICK+SWAP	
0F6BF	4PICK#+SWAP	
0F6D8	#+SWAP	
0F6EC	#-SWAP	
0F700	#1+SWAP	
0F714	ZEROSWAP	
0F728	#1-1SWAP	
0F741	ONESWAP	
0F755	COERCESWAP	
0F769	%>%%SWAP	
0F791	XYZ>ZTRUE	
0F7A5	4ROLLSWAP	
0F7B9	3PICKSWAP	
0F7CD	4PICKSWAP	
0F7E1	1GETSWAP	
0F7F5	?SWAP	
0F809	!append$SWAP	
0F81D	NOT?SWAPDROP	
0F836	?SWAPDROP	
0F84F	#1+NDROP	
0F84F	N+1DROP	
0F863	ROLLDROP	
0F877	MDIMSDROP	
0F892	DUPROT	
0F8A6	DROPROT	
0F8BA	#1-ROT	
0F8E2	FOURROLLROT	
0F8E2	4ROLLROT	
0F8F6	4UNROLLROT	
0F90A	DROPOVER	
0F91E	EQOVER	
0F932	#+OVER	
0F946	#-OVER	
0F95A	ZEROOVER	

0F96E	UNROTOVER	0FC7D	ONE_EQ	0FF93	#5=
0F982	4ROLLOVER	0FC91	#>?SKIP	0FFA7	#2<>
0F996	3PICKOVER	0FCAA	COLASKIP	0FFBB	OVER#>
0F9AA	4PICKOVER	0FCBE	NOT_UNTIL	0FFCF	ONE#>
0F9BE	DUPPICK	0FCD7	NOT_WHILE	0FFCF	#>1
0F9D2	DUPROLL	0FCF0	DUP#0<>WHILE	0FFE3	DUP3PICK#+
0F9FA	8UNROLL	0FD09	DUPINDEX@	0FFE3	2DUP#+
0FA0E	10UNROLL	0FD1D	SWAPINDEX@	0FFF7	ROT#+
0FA22	OVERARSIZE	0FD31	OVERINDEX@	1000B	OVER#+
0FA3D	'ERRJMP	0FD45	SWAPLOOP	1001F	3PICK#+
0FA51	caseERRJMP	0FD5E	DROPLOOP	10033	4PICK#+
0FA65	?CARCOMP	0FD77	DUP#0_DO	10047	ROT#-
0FA79	NEWLINE$&$	0FD90	toLEN_DO	1005B	OVER#-
0FA79	NEWLINE&$	0FDAE	1GETABND	1006F	INDEX@#-
0FA8D	#1-{}N	0FDC2	DUP1LAMBIND	10083	SWAPOVER#-
0FAA1	TWO{}N	0FDC7	1LAMBIND	10097	ROT#1+
0FAB5	THREE{}N	0FDDB	caseTRUE	100AB	SWAP#1-
0FAC9	DUPINCOMP	0FDEF	TRUEFALSE	100BF	DROP#1-
0FADD	SWAPINCOMP	0FDEF	TrueFalse	100D3	ERROROUT
0FAF1	DUPNULL$?	0FE03	FALSETRUE	100E7	RSKIP
0FB05	DUPNULLCOMP?	0FE03	FalseTrue	10105	GROB!ZERODRP
0FB19	DUPLENCOMP	0FE17	ZEROFALSE	10119	casedrptru
0FB2D	#1-SUB$	0FE2B	ONEFALSE	10132	NOTcaseTRUE
0FB41	1_#1-SUB	0FE3F	#=casedrpfls	1014B	NOTcaseFALSE
0FB41	1_#1-SUB$	0FE62	casedrpfls	10164	?SEMIDROP
0FB55	LAST$	0FE7B	case2drpfls	1017D	SWAPUnDROP
0FB69	#1+LAST$	0FE94	caseFALSE	10191	SWAPUnNDROP
0FB7D	DUP$>ID	0FEA8	ORNOT	101A5	DUP'
0FB91	SWAP%>C%	0FEBC	EQUALNOT	101B9	SWAP'
0FBA5	'NOP	0FED0	2DUPEQ	101CD	DROP'
0FBB9	::NEVAL	0FEE4	EQOR	101E1	OVER'
0FBCD	2GETEVAL	0FEF8	EQUALOR	101F5	STO'
0FBE1	DROPRDROP	0FF0C	2#0=OR	10209	TRUE'
0FBFA	SWAPCOLA	0FF2A	OVER#0=	1021D	ONEFALSE'
0FC0E	XYZ>ZCOLA	0FF3E	OVER#<	10236	FALSE'
0FC22	#0=?SKIP	0FF52	#<3	1024A	#1+'
0FC4B	#1=?SKIP	0FF66	DUP#<7	1025E	'R'R
0FC64	#=?SKIP	0FF7F	INNER#1=	1027C	'RRDROP

10295	ONECOLA	105B5	#<>case	1089E	1GETLAM#0=
102A9	dvarlsBIND	105C9	#>2case	108B2	OVER#1-
102BD	'LAMLNAMESTO	105E2	#>case	108C6	#=POSCOMP
102D6	'xDEREQ	105F6	j%0=case	108DF	EQUALPOSCOMP
102EF	DUPNULL{}?	1060A	REALcase	108DF	EQUALPOSCMP
1031C	DUPZERO	1061E	dARRYcase	108F8	NTHOF
10330	DUPONE	10632	dLISTcase	1090C	EQLookup
10344	SWAPONE	10646	EditExstCase	109DD	CHR>$
10358	ONEDUP	1065A	ANDNOTcase	109F6	1NULLLAM{}
10358	ONEONE	1066E	EQUALNOTcase	109FB	NULLLAM
1036C	DUPTWO	10682	dIDNTNcase	10A07	DOINDIR
10380	NOTcsdrpfls	10696	dREALNcase	10A1C	CHR_00
10399	caseSIZEERR	106AA	EQIT	10A23	CHR_...
103AD	NcaseSIZEERR	106C3	DUP#0=IT	10A2A	CHR_'
103C1	NcaseTYPEERR	106DC	ANDITE	10A31	CHR_DblQuote
103E9	'xDER	106F0	EQITE	10A38	CHR_#
103FD	%%/>%	10704	#0=ITE	10A3F	CHR_*
10411	UNCOERCE%%	10718	#<ITE	10A46	CHR_+
10425	DUP%0=	1072C	#>ITE	10A4D	CHR_,
10439	SWAP%%/	10740	DUP#0=ITE	10A54	CHR_-
1044D	caseDrpBadKy	10754	UserITE	10A5B	CHR_.
10466	caseDEADKEY	10768	SysITE	10A62	CHR_/
10466	caseDoBadKey	1077C	top&Cr	10A69	CHR_0
1047F	GROBDIMw	10795	metaROTDUP	10A70	CHR_1
104A7	XYZW>YWZX	107A9	ROTUntop&	10A77	CHR_2
104A7	SWAP4ROLL	107BD	roll2top&	10A7E	CHR_3
104BB	2DUP5ROLL	107BD	rolltwotop&	10A85	CHR_4
104CF	SWAP3PICK	107D1	plDRPpZparg	10A8C	CHR_5
104E3	3PICK3PICK	107E5	&$SWAP	10A93	CHR_6
104F7	SWAP4PICK	107F9	SWAPCKREF	10A9A	CHR_7
1050B	OVER5PICK	1080D	pZpargSWAPUn	10AA1	CHR_8
1051F	EQUALcasedrp	10821	DROPNDROP	10AA8	CHR_9
10538	DUP#0=csDROP	10835	2OVER	10AAF	CHR_:
10551	jEQcase	10849	?Ob>Seco	10AB6	CHR_;
10565	ANDcase	10862	Ob>Seco	10ABD	CHR_<
10579	EQUALcase	10876	2Ob>Seco	10AC4	CHR_=
1058D	#<case	1088A	ExitAtLOOP	10ACB	CHR_>
105A1	#1=case	1088A	ZEROISTOPSTO	10AD2	CHR_A

10AD9 CHR_B	10BE3 CHR_n	10CED STypeLstChr
10AE0 CHR_C	10BEA CHR_o	10CF4 STypeMatChr
10AE7 CHR_D	10BF1 CHR_p	10CFB STypeLibChr
10AEE CHR_E	10BF8 CHR_q	10D02 STypeTgtChr
10AF5 CHR_F	10BFF CHR_r	10D09 STypeVecChr
10AFC CHR_G	10C06 CHR_s	10D10 STypeAnyChr
10B03 CHR_H	10C0D CHR_t	10D17 $_R<<
10B0A CHR_I	10C14 CHR_u	10D27 $_R<Z
10B11 CHR_J	10C1B CHR_v	10D37 $_XYZ
10B18 CHR_K	10C22 CHR_w	10D47 $_<<>>
10B1F CHR_L	10C29 CHR_x	10D55 $_{}
10B26 CHR_M	10C30 CHR_y	10D71 $_''
10B2D CHR_N	10C37 CHR_z	10D7F $_::
10B34 CHR_O	10C3E CHR_->	10D8D $_LRParens
10B3B CHR_P	10C45 CHR_<<	10D9B $_2DQ
10B42 CHR_Q	10C4C CHR_>>	10DA9 $_ECHO
10B49 CHR_R	10C53 CHR_Angle	10DBB $_EXIT
10B50 CHR_S	10C5A CHR_Deriv	10DCD $_Undefined
10B57 CHR_T	10C61 CHR_Integral	10DE9 $_RAD
10B5E CHR_U	10C68 CHR_LeftPar	10DF9 $_GRAD
10B65 CHR_V	10C6F CHR_Newline	10E0B ?DispMenu
10B6C CHR_W	10C76 CHR_Pi	10E29 DispMenu
10B73 CHR_X	10C7D CHR_RightPar	10E3D DispMenu.1
10B7A CHR_Y	10C84 CHR_Sigma	10F64 MakeStdLabel
10B81 CHR_Z	10C8B CHR_Space	10FC6 MakeBoxLabel
10B88 CHR_a	10C92 CHR_UndScore	11028 MakeDirLabel
10B8F CHR_b	10C99 CHR_[	1108A MakeInvLabel
10B96 CHR_c	10CA0 CHR_]	11C0D #_102
10B9D CHR_d	10CA7 CHR_{	11C0D #_258_d
10BA4 CHR_e	10CAE CHR_}	11C0D illnameerr
10BAB CHR_f	10CB5 CHR_<=	11C0D Err#Chr00
10BB2 CHR_g	10CBC CHR_>=	11C17 #SyntaxErr
10BB9 CHR_h	10CC3 CHR_<>	11C21 BINT_263d
10BC0 CHR_i	10CCA STypeDirChr	11C21 #ObTypeBase
10BC7 CHR_j	10CD1 STypeZapChr	11C2B REALREALOB
10BCE CHR_k	10CD8 STypeNteChr	11C35 3REAL
10BD5 CHR_l	10CDF STypePgmChr	11C3F #_291_d
10BDC CHR_m	10CE6 STypeVarChr	11C3F #_123

11C3F	Err#Kill	11D57	IDLISTOB	11F12	%3
11C49	#_124	11D61	LAMANYANY	11F27	%4
11C49	Err#NoLstStk	11D61	LSTBIMACROM#	11F3C	%5
11C49	#_292_d	11D61	FSTMACROROM#	11F51	%6
11C53	#NoRoomForSt	11D89	PROGIDREAL	11F66	%7
11C53	BINT_305d	11D93	PROGIDCMP	11F7B	%8
11C5D	#Warning:NL	11D9D	PROGIDLIST	11F90	%9
11C5D	BINT_306d	11DA7	PROGIDEXT	11FA5	%-1
11C67	REALSTRSTR	11DB1	ATTNERR	11FBA	%-2
11C67	BINT_307d	11DCF	SYMREALREAL	11FCF	%-3
11C67	#Error:	11DD9	SYMREALCMP	11FE4	%-4
11C71	VLMpurgemsg	11DE3	SYMREALSYM	11FF9	%-5
11C7B	VLMmsg	11DED	SYMCMPREAL	1200E	%-6
11C85	REALSTRID	11DF7	SYMCMPCMP	12023	%-7
11C85	VLMstkmsg	11E01	SYMCMPSYM	12038	%-8
11C8F	VLMundomsg	11E0B	SYMIDREAL	1204D	%-9
11C99	VLMhistmsg	11E15	SYMIDCMP	12062	%PI
11CA3	VLMUserKeys	11E1F	SYMIDLIST	12077	%%PI
11CAD	VLMAlarmMsg	11E29	SYMIDEXT	12091	%MAXREAL
11CB7	VLMlastargs	11E33	SYMSYMREAL	120A6	%-MAXREAL
11CC1	VLMcmdlmsg	11E3D	SYMSYMCMP	120BB	%MINREAL
11CCB	Err#Cont	11E47	3SYM	120D0	%-MINREAL
11CD5	REALLISTREAL	11E51	XFERFAIL	120E5	%%0
11CD5	INTEGER337	11E5B	PROTERR	120FF	%%1
11CDF	CMPOBOB	11E65	InvalServCmd	12119	%%2
11CE9	#_517_d	11E6F	Connecting	12133	%%3
11CE9	Err#NoLstArg	11E79	Retry	1214D	%%4
11CE9	#_205	11E83	#CAlarmErr	12167	%%5
11CF3	STRREALREAL	11E8D	EXTOBOB	12181	%%.1
11CFD	ARRYREALREAL	11E97	#EXITERR	1219B	%%.5
11D07	ARRYREALCMP	11EA1	MINUSONE	121B5	%%10
11D11	3ARRY	11EAB	MINUSTWO	121CF	%e
11D1B	ARRYLISTREAL	11EB5	MINUSTHREE	121E4	%.5
11D25	ARRYLISTCMP	11EBF	MINUSFOUR	121F9	%-.5
11D2F	LISTREALOB	11EC9	MINUSFIVE	1220E	%10
11D39	LISTREALREAL	11ED3	%0	12277	%15
11D43	LISTLISTOB	11EE8	%1	12349	%25
11D4D	IDREALOB	11EFD	%2	12388	%180

1239D	%200	12588	tok-	127F2	doptr@
123B2	%360	12594	tok=	12802	doptr!
123C7	%400	125A0	tokSQRT	1281D	TOLVarN!
123DC	rbrac	125AC	tokDER	12840	TOLVarN@
123DC	tok]	125B8	tokCTGROB	12863	ClrAllTVars
123E8	lbrac	125CA	tokCTSTR	12899	ClrAllTOLVs
123F6	tok[	125D8	tok0	128B5	%0AllTopicVs
12402	tok{	125E4	tok1	128F0	%0AllTOLVars
1240E	tok}	125F0	tok2	1292B	TOLVarSet!
1241A	toksharp	125FC	tok3	129F8	%0TOLVarSet
12426	tokuscore	12608	tok4	12A48	1getcxt!
12432	tok$	12614	tok5	12A5C	DoInCxt
1243E	tok&	12620	tok6	12AB6	DoInCalcCxt
1244A	tokESC	1262C	tok7	12ACF	DoInAppCxt
12456	tok>>	12638	tok8	12AE8	DoInFuncCxt
12462	tok<<	12644	tok9	12B01	DoInPolarCxt
1246E	tokexponent	12696	CALCCXT!	12B1A	DoInParamCxt
1247A	tokanglesign	126A6	CALCCXT@	12B33	DoInSeqCxt
12486	tokSIGMA	126B6	PGMCXT!	12B4C	DoInStatCxt
12492	tokWHERE	126C6	PGMCXT@	12B65	DoInSolveCxt
1249E	14SPACES$	126D6	NOTESCXT!	12B7E	DoInOtherCxt
124C4	LF$	126E6	NOTESCXT@	12C4B	otherNG?
124C4	NEWLINE$	126F6	apletPTR!	12C7E	GET@tTYPER
124D0	$DER	12706	apletPTR@	12CA1	TopicVar1!
124E0	tok_	12716	funcPTR!	12CA2	TopicVar2!
124E0	SPACE$	12726	funcPTR@	12CA3	TopicVar3!
124EC	tokUNKNOWN	12736	polarPTR!	12CA4	TopicVar4!
12504	tokquote	12746	polarPTR@	12CA5	TopicVar5!
12510	tok'	12756	paramPTR!	12CA6	TopicVar6!
1251C	tok,	12766	paramPTR@	12CA7	TopicVar7!
12528	tok.	12776	seqPTR!	12CA8	TopicVar8!
12534	tok;	12786	seqPTR@	12CA9	TopicVar9!
12540	toklparen	12796	statPTR!	12CAA	TopicVar10!
1254C	tokrparen	127A6	statPTR@	12CAB	TopicVar11!
12558	tok^	127B6	solvePTR!	12CAC	TopicVar12!
12564	tok*	127C6	solvePTR@	12CAD	TopicVar13!
12570	tok/	127D6	otherPTR!	12CAE	TopicVar14!
1257C	tok+	127E6	otherPTR@	12CAF	TopicVar15!

12CB0	TopicVar16!	12CD6	TopicVar54!	12CFC	TopicVar1@
12CB1	TopicVar17!	12CD7	TopicVar55!	12CFD	TopicVar2@
12CB2	TopicVar18!	12CD8	TopicVar56!	12CFE	TopicVar3@
12CB3	TopicVar19!	12CD9	TopicVar57!	12CFF	TopicVar4@
12CB4	TopicVar20!	12CDA	TopicVar58!	12D00	TopicVar5@
12CB5	TopicVar21!	12CDB	TopicVar59!	12D01	TopicVar6@
12CB6	TopicVar22!	12CDC	TopicVar60!	12D02	TopicVar7@
12CB7	TopicVar23!	12CDD	TopicVar61!	12D03	TopicVar8@
12CB8	TopicVar24!	12CDE	TopicVar62!	12D04	TopicVar9@
12CB9	TopicVar25!	12CDF	TopicVar63!	12D05	TopicVar10@
12CBA	TopicVar26!	12CE0	TopicVar64!	12D06	TopicVar11@
12CBB	TopicVar27!	12CE1	TopicVar65!	12D07	TopicVar12@
12CBC	TopicVar28!	12CE2	TopicVar66!	12D08	TopicVar13@
12CBD	TopicVar29!	12CE3	TopicVar67!	12D09	TopicVar14@
12CBE	TopicVar30!	12CE4	TopicVar68!	12D0A	TopicVar15@
12CBF	TopicVar31!	12CE5	TopicVar69!	12D0B	TopicVar16@
12CC0	TopicVar32!	12CE6	TopicVar70!	12D0C	TopicVar17@
12CC1	TopicVar33!	12CE7	TopicVar71!	12D0D	TopicVar18@
12CC2	TopicVar34!	12CE8	TopicVar72!	12D0E	TopicVar19@
12CC3	TopicVar35!	12CE9	TopicVar73!	12D0F	TopicVar20@
12CC4	TopicVar36!	12CEA	TopicVar74!	12D10	TopicVar21@
12CC5	TopicVar37!	12CEB	TopicVar75!	12D11	TopicVar22@
12CC6	TopicVar38!	12CEC	TopicVar76!	12D12	TopicVar23@
12CC7	TopicVar39!	12CED	TopicVar77!	12D13	TopicVar24@
12CC8	TopicVar40!	12CEE	TopicVar78!	12D14	TopicVar25@
12CC9	TopicVar41!	12CEF	TopicVar79!	12D15	TopicVar26@
12CCA	TopicVar42!	12CF0	TopicVar80!	12D16	TopicVar27@
12CCB	TopicVar43!	12CF1	TopicVar81!	12D17	TopicVar28@
12CCC	TopicVar44!	12CF2	TopicVar82!	12D18	TopicVar29@
12CCD	TopicVar45!	12CF3	TopicVar83!	12D19	TopicVar30@
12CCE	TopicVar46!	12CF4	TopicVar84!	12D1A	TopicVar31@
12CCF	TopicVar47!	12CF5	TopicVar85!	12D1B	TopicVar32@
12CD0	TopicVar48!	12CF6	TopicVar86!	12D1C	TopicVar33@
12CD1	TopicVar49!	12CF7	TopicVar87!	12D1D	TopicVar34@
12CD2	TopicVar50!	12CF8	TopicVar88!	12D1E	TopicVar35@
12CD3	TopicVar51!	12CF9	TopicVar89!	12D1F	TopicVar36@
12CD4	TopicVar52!	12CFA	TopicVar90!	12D20	TopicVar37@
12CD5	TopicVar53!	12CFB	TopicVar91!	12D21	TopicVar38@

12D22	TopicVar39@	12D48	TopicVar77@	12D6E	TOLVar24!
12D23	TopicVar40@	12D49	TopicVar78@	12D6F	TOLVar25!
12D24	TopicVar41@	12D4A	TopicVar79@	12D70	TOLVar26!
12D25	TopicVar42@	12D4B	TopicVar80@	12D71	TOLVar27!
12D26	TopicVar43@	12D4C	TopicVar81@	12D72	TOLVar28!
12D27	TopicVar44@	12D4D	TopicVar82@	12D73	TOLVar29!
12D28	TopicVar45@	12D4E	TopicVar83@	12D74	TOLVar30!
12D29	TopicVar46@	12D4F	TopicVar84@	12D75	TOLVar31!
12D2A	TopicVar47@	12D50	TopicVar85@	12D76	TOLVar32!
12D2B	TopicVar48@	12D51	TopicVar86@	12D77	TOLVar33!
12D2C	TopicVar49@	12D52	TopicVar87@	12D78	TOLVar34!
12D2D	TopicVar50@	12D53	TopicVar88@	12D79	TOLVar35!
12D2E	TopicVar51@	12D54	TopicVar89@	12D7A	TOLVar36!
12D2F	TopicVar52@	12D55	TopicVar90@	12D7B	TOLVar37!
12D30	TopicVar53@	12D56	TopicVar91@	12D7C	TOLVar38!
12D31	TopicVar54@	12D57	TOLVar1!	12D7D	TOLVar39!
12D32	TopicVar55@	12D58	TOLVar2!	12D7E	TOLVar40!
12D33	TopicVar56@	12D59	TOLVar3!	12D7F	TOLVar41!
12D34	TopicVar57@	12D5A	TOLVar4!	12D80	TOLVar42!
12D35	TopicVar58@	12D5B	TOLVar5!	12D81	TOLVar43!
12D36	TopicVar59@	12D5C	TOLVar6!	12D82	TOLVar44!
12D37	TopicVar60@	12D5D	TOLVar7!	12D83	TOLVar45!
12D38	TopicVar61@	12D5E	TOLVar8!	12D84	TOLVar46!
12D39	TopicVar62@	12D5F	TOLVar9!	12D85	TOLVar47!
12D3A	TopicVar63@	12D60	TOLVar10!	12D86	TOLVar48!
12D3B	TopicVar64@	12D61	TOLVar11!	12D87	TOLVar49!
12D3C	TopicVar65@	12D62	TOLVar12!	12D88	TOLVar50!
12D3D	TopicVar66@	12D63	TOLVar13!	12D89	TOLVar51!
12D3E	TopicVar67@	12D64	TOLVar14!	12D8A	TOLVar52!
12D3F	TopicVar68@	12D65	TOLVar15!	12D8B	TOLVar53!
12D40	TopicVar69@	12D66	TOLVar16!	12D8C	TOLVar54!
12D41	TopicVar70@	12D67	TOLVar17!	12D8D	TOLVar55!
12D42	TopicVar71@	12D68	TOLVar18!	12D8E	TOLVar56!
12D43	TopicVar72@	12D69	TOLVar19!	12D8F	TOLVar57!
12D44	TopicVar73@	12D6A	TOLVar20!	12D90	TOLVar58!
12D45	TopicVar74@	12D6B	TOLVar21!	12D91	TOLVar59!
12D46	TopicVar75@	12D6C	TOLVar22!	12D92	TOLVar60!
12D47	TopicVar76@	12D6D	TOLVar23!	12D93	TOLVar61!

12D94	TOLVar62!	12DBA	TOLVar100!	12DE0	TOLVar138!
12D95	TOLVar63!	12DBB	TOLVar101!	12DE1	TOLVar139!
12D96	TOLVar64!	12DBC	TOLVar102!	12DE2	TOLVar140!
12D97	TOLVar65!	12DBD	TOLVar103!	12DE3	TOLVar141!
12D98	TOLVar66!	12DBE	TOLVar104!	12DE4	TOLVar142!
12D99	TOLVar67!	12DBF	TOLVar105!	12DE5	TOLVar143!
12D9A	TOLVar68!	12DC0	TOLVar106!	12DE6	TOLVar144!
12D9B	TOLVar69!	12DC1	TOLVar107!	12DE7	TOLVar145!
12D9C	TOLVar70!	12DC2	TOLVar108!	12DE8	TOLVar146!
12D9D	TOLVar71!	12DC3	TOLVar109!	12DE9	TOLVar147!
12D9E	TOLVar72!	12DC4	TOLVar110!	12DEA	TOLVar148!
12D9F	TOLVar73!	12DC5	TOLVar111!	12DEB	TOLVar149!
12DA0	TOLVar74!	12DC6	TOLVar112!	12DEC	TOLVar150!
12DA1	TOLVar75!	12DC7	TOLVar113!	12DED	TOLVar151!
12DA2	TOLVar76!	12DC8	TOLVar114!	12DEE	TOLVar152!
12DA3	TOLVar77!	12DC9	TOLVar115!	12DEF	TOLVar153!
12DA4	TOLVar78!	12DCA	TOLVar116!	12DF0	TOLVar154!
12DA5	TOLVar79!	12DCB	TOLVar117!	12DF1	TOLVar155!
12DA6	TOLVar80!	12DCC	TOLVar118!	12DF2	TOLVar156!
12DA7	TOLVar81!	12DCD	TOLVar119!	12DF3	TOLVar157!
12DA8	TOLVar82!	12DCE	TOLVar120!	12DF4	TOLVar158!
12DA9	TOLVar83!	12DCF	TOLVar121!	12DF5	TOLVar159!
12DAA	TOLVar84!	12DD0	TOLVar122!	12DF6	TOLVar160!
12DAB	TOLVar85!	12DD1	TOLVar123!	12DF7	TOLVar161!
12DAC	TOLVar86!	12DD2	TOLVar124!	12DF8	TOLVar162!
12DAD	TOLVar87!	12DD3	TOLVar125!	12DF9	TOLVar163!
12DAE	TOLVar88!	12DD4	TOLVar126!	12DFA	TOLVar164!
12DAF	TOLVar89!	12DD5	TOLVar127!	12DFB	TOLVar165!
12DB0	TOLVar90!	12DD6	TOLVar128!	12DFC	TOLVar166!
12DB1	TOLVar91!	12DD7	TOLVar129!	12DFD	TOLVar167!
12DB2	TOLVar92!	12DD8	TOLVar130!	12DFE	TOLVar168!
12DB3	TOLVar93!	12DD9	TOLVar131!	12DFF	TOLVar169!
12DB4	TOLVar94!	12DDA	TOLVar132!	12E00	TOLVar170!
12DB5	TOLVar95!	12DDB	TOLVar133!	12E01	TOLVar171!
12DB6	TOLVar96!	12DDC	TOLVar134!	12E02	TOLVar172!
12DB7	TOLVar97!	12DDD	TOLVar135!	12E03	TOLVar173!
12DB8	TOLVar98!	12DDE	TOLVar136!	12E04	TOLVar174!
12DB9	TOLVar99!	12DDF	TOLVar137!	12E05	TOLVar175!

12E06	TOLVar176!	12E2C	TOLVar214!	12E52	TOLVar36@
12E07	TOLVar177!	12E2D	TOLVar215!	12E53	TOLVar37@
12E08	TOLVar178!	12E2E	TOLVar216!	12E54	TOLVar38@
12E09	TOLVar179!	12E2F	TOLVar1@	12E55	TOLVar39@
12E0A	TOLVar180!	12E30	TOLVar2@	12E56	TOLVar40@
12E0B	TOLVar181!	12E31	TOLVar3@	12E57	TOLVar41@
12E0C	TOLVar182!	12E32	TOLVar4@	12E58	TOLVar42@
12E0D	TOLVar183!	12E33	TOLVar5@	12E59	TOLVar43@
12E0E	TOLVar184!	12E34	TOLVar6@	12E5A	TOLVar44@
12E0F	TOLVar185!	12E35	TOLVar7@	12E5B	TOLVar45@
12E10	TOLVar186!	12E36	TOLVar8@	12E5C	TOLVar46@
12E11	TOLVar187!	12E37	TOLVar9@	12E5D	TOLVar47@
12E12	TOLVar188!	12E38	TOLVar10@	12E5E	TOLVar48@
12E13	TOLVar189!	12E39	TOLVar11@	12E5F	TOLVar49@
12E14	TOLVar190!	12E3A	TOLVar12@	12E60	TOLVar50@
12E15	TOLVar191!	12E3B	TOLVar13@	12E61	TOLVar51@
12E16	TOLVar192!	12E3C	TOLVar14@	12E62	TOLVar52@
12E17	TOLVar193!	12E3D	TOLVar15@	12E63	TOLVar53@
12E18	TOLVar194!	12E3E	TOLVar16@	12E64	TOLVar54@
12E19	TOLVar195!	12E3F	TOLVar17@	12E65	TOLVar55@
12E1A	TOLVar196!	12E40	TOLVar18@	12E66	TOLVar56@
12E1B	TOLVar197!	12E41	TOLVar19@	12E67	TOLVar57@
12E1C	TOLVar198!	12E42	TOLVar20@	12E68	TOLVar58@
12E1D	TOLVar199!	12E43	TOLVar21@	12E69	TOLVar59@
12E1E	TOLVar200!	12E44	TOLVar22@	12E6A	TOLVar60@
12E1F	TOLVar201!	12E45	TOLVar23@	12E6B	TOLVar61@
12E20	TOLVar202!	12E46	TOLVar24@	12E6C	TOLVar62@
12E21	TOLVar203!	12E47	TOLVar25@	12E6D	TOLVar63@
12E22	TOLVar204!	12E48	TOLVar26@	12E6E	TOLVar64@
12E23	TOLVar205!	12E49	TOLVar27@	12E6F	TOLVar65@
12E24	TOLVar206!	12E4A	TOLVar28@	12E70	TOLVar66@
12E25	TOLVar207!	12E4B	TOLVar29@	12E71	TOLVar67@
12E26	TOLVar208!	12E4C	TOLVar30@	12E72	TOLVar68@
12E27	TOLVar209!	12E4D	TOLVar31@	12E73	TOLVar69@
12E28	TOLVar210!	12E4E	TOLVar32@	12E74	TOLVar70@
12E29	TOLVar211!	12E4F	TOLVar33@	12E75	TOLVar71@
12E2A	TOLVar212!	12E50	TOLVar34@	12E76	TOLVar72@
12E2B	TOLVar213!	12E51	TOLVar35@	12E77	TOLVar73@

12E78	TOLVar74@	12E9E	TOLVar112@	12EC4	TOLVar150@
12E79	TOLVar75@	12E9F	TOLVar113@	12EC5	TOLVar151@
12E7A	TOLVar76@	12EA0	TOLVar114@	12EC6	TOLVar152@
12E7B	TOLVar77@	12EA1	TOLVar115@	12EC7	TOLVar153@
12E7C	TOLVar78@	12EA2	TOLVar116@	12EC8	TOLVar154@
12E7D	TOLVar79@	12EA3	TOLVar117@	12EC9	TOLVar155@
12E7E	TOLVar80@	12EA4	TOLVar118@	12ECA	TOLVar156@
12E7F	TOLVar81@	12EA5	TOLVar119@	12ECB	TOLVar157@
12E80	TOLVar82@	12EA6	TOLVar120@	12ECC	TOLVar158@
12E81	TOLVar83@	12EA7	TOLVar121@	12ECD	TOLVar159@
12E82	TOLVar84@	12EA8	TOLVar122@	12ECE	TOLVar160@
12E83	TOLVar85@	12EA9	TOLVar123@	12ECF	TOLVar161@
12E84	TOLVar86@	12EAA	TOLVar124@	12ED0	TOLVar162@
12E85	TOLVar87@	12EAB	TOLVar125@	12ED1	TOLVar163@
12E86	TOLVar88@	12EAC	TOLVar126@	12ED2	TOLVar164@
12E87	TOLVar89@	12EAD	TOLVar127@	12ED3	TOLVar165@
12E88	TOLVar90@	12EAE	TOLVar128@	12ED4	TOLVar166@
12E89	TOLVar91@	12EAF	TOLVar129@	12ED5	TOLVar167@
12E8A	TOLVar92@	12EB0	TOLVar130@	12ED6	TOLVar168@
12E8B	TOLVar93@	12EB1	TOLVar131@	12ED7	TOLVar169@
12E8C	TOLVar94@	12EB2	TOLVar132@	12ED8	TOLVar170@
12E8D	TOLVar95@	12EB3	TOLVar133@	12ED9	TOLVar171@
12E8E	TOLVar96@	12EB4	TOLVar134@	12EDA	TOLVar172@
12E8F	TOLVar97@	12EB5	TOLVar135@	12EDB	TOLVar173@
12E90	TOLVar98@	12EB6	TOLVar136@	12EDC	TOLVar174@
12E91	TOLVar99@	12EB7	TOLVar137@	12EDD	TOLVar175@
12E92	TOLVar100@	12EB8	TOLVar138@	12EDE	TOLVar176@
12E93	TOLVar101@	12EB9	TOLVar139@	12EDF	TOLVar177@
12E94	TOLVar102@	12EBA	TOLVar140@	12EE0	TOLVar178@
12E95	TOLVar103@	12EBB	TOLVar141@	12EE1	TOLVar179@
12E96	TOLVar104@	12EBC	TOLVar142@	12EE2	TOLVar180@
12E97	TOLVar105@	12EBD	TOLVar143@	12EE3	TOLVar181@
12E98	TOLVar106@	12EBE	TOLVar144@	12EE4	TOLVar182@
12E99	TOLVar107@	12EBF	TOLVar145@	12EE5	TOLVar183@
12E9A	TOLVar108@	12EC0	TOLVar146@	12EE6	TOLVar184@
12E9B	TOLVar109@	12EC1	TOLVar147@	12EE7	TOLVar185@
12E9C	TOLVar110@	12EC2	TOLVar148@	12EE8	TOLVar186@
12E9D	TOLVar111@	12EC3	TOLVar149@	12EE9	TOLVar187@

12EEA	TOLVar188@	1340B	COMPEVAL	15EAD	DropSysObs
12EEB	TOLVar189@	1341F	CK1&Dispatch	161CC	RESOROMP
12EEC	TOLVar190@	13430	CK2&Dispatch	16451	CKREF
12EED	TOLVar191@	13441	CK3&Dispatch	1663B	%%-1
12EEE	TOLVar192@	13452	CK4&Dispatch	16747	P{}N
12EEF	TOLVar193@	13463	CK5&Dispatch	1677E	P::N
12EF0	TOLVar194@	1352A	CK&DISPATCH1	16C46	CRLF$
12EF1	TOLVar195@	13573	CK&DISPATCH0	16C54	STArryList
12EF2	TOLVar196@	13582	CK&DISPATCH2	16C90	Done?CkNoNul:
12EF3	TOLVar197@	13A13	#*OVF	16D7E	FuncSplitViewers
12EF4	TOLVar198@	13AF2	!>ARRAY	16EB1	PUTXMAX
12EF5	TOLVar199@	13C06	NUMEVAL1	16ECA	PUTYMAX
12EF6	TOLVar200@	13EA3	solverTTT	16EE3	NULLargcase
12EF7	TOLVar201@	140B7	INHARDROM?	16F3F	SV_keycancel
12EF8	TOLVar202@	14370	SUB$1#	16F53	SV_keyok
12EF9	TOLVar203@	143AF	Push#FLoop	16F67	SV_keycheck
12EFA	TOLVar204@	14D89	'Rapndit	16F7B	SV_keyedit
12EFB	TOLVar205@	14E94	'RSWP1+	16F9C	SV_keyeval
12EFC	TOLVar206@	14ED5	EQcaseDROP	16FBD	SV_keyshow
12EFD	TOLVar207@	14F3E	2Col?Case2Col	1708D	WaitTbz0
12EFE	TOLVar208@	14F70	#1-SWAP	170BB	CloseUart
12EFF	TOLVar209@	14F70	pull	171E2	ARSIZE
12F00	TOLVar210@	14F98	psh1&	171EE	DIMLIMITS
12F01	TOLVar211@	15068	psh	171FA	MAKEARRY
12F02	TOLVar212@	150A4	GROBDIM	17206	MATCON
12F03	TOLVar213@	150FC	ROT#1+UNROT	17212	MATREDIM
12F04	TOLVar214@	15194	reversym	1721E	MATTRN
12F05	TOLVar215@	15337	NDUPN	1722A	MDIMS
12F06	TOLVar216@	1538F	metaDUP	17236	DoInApLbCxt
12F1E	CK0	154A7	PageUpYGrob	17357	tok:
131EC	SETNONEXTERR	154EB	PageUpNGrob	1747B	REPEATER
131FC	SETSIZEERR	1552F	PageDnYGrob	1747B	REPEATERCH
1320C	SETTYPEERR	15573	PageDnNGrob	17C61	symcomp
1321C	SETSTACKERR	156DA	StoAns@	17C89	GetPart1
13231	%ABSCOERCE	15A41	DoInputForm	17C9D	GetPart2
13236	ABSCOERCE	15B08	#=Lookup	17CB1	GetPart3
13244	COERCE	15BE9	?GetMsg	17CC5	EvalPart1
13319	UNCOERCE	15C02	GetElNoRomp	17CD9	EvalPart2

17CED	EvalPart3	1B7E2	%=	1BD83	%COMB
17D47	STBigKeyDef	1B7ED	%<>	1BD96	%PERM
17D98	STEditKeyDef	1B7F8	%SGN	1BE6E	%FP
17DC8	STInsKeyDef	1B821	%ABS	1BE81	%IP
17DF6	STStatsKeyDef	1B841	%CHS	1BE94	%CEIL
18D1D	TBStndSplitInfo	1B851	%MANTISSA	1BEA7	%FLOOR
18DBC	PUTEL	1B895	%+	1BEE3	%RAN
1924D	Docrunchc	1B8A2	%-	1BF65	%RANDOMIZE
192D4	DOSHOWIT1C	1B8DD	%*	1BFE5	%FACT
1930B	SWAP2DUP	1B8EA	%OF	1C3AF	%REC>%POL
1939B	SetUserFlag	1B91F	%/	1C3DC	%POL>%REC
193A7	SetSysFlag	1B92C	%T	1C413	%SPH>%REC
193CB	ClrUserFlag	1B951	%CH	1C44A	RNDXY
193D7	ClrSysFlag	1B991	%^	1C45E	TRCXY
193EE	TestUserFlag	1B9A2	%NROOT	1D622	MAKEGROB
193FA	TestSysFlag	1BA2A	%SQRT	1D704	GROB!
199AA	Shrink$	1BA50	%EXP	1D9BA	SUBGROB
19ACC	PopASavptr	1BA63	%EXPM1	1DAF8	GROB!ZERO
1B374	PUSH%LOOP	1BA8F	%LN	1DD7E	$>BIGGROB
1B3EB	%%>%	1BAA2	%LOG	1DD8B	$>GROB
1B3FC	%>%%	1BAC8	%LNP1	1DFAC	$>grob
1B543	%D>R	1BADB	%ALOG	1E5BD	INVGROB
1B576	%R>D	1BAFD	%MOD	1E701	DISPN
1B594	%>HMS	1BB10	%SIN	1E701	BIGDISPN
1B5AD	%HMS>	1BB61	%COS	1E735	DISPROW1
1B5C1	%HMS+	1BBB2	%TAN	1E735	DISP@01
1B5E9	%HMS-	1BBE2	%ASIN	1E735	BIGDISPROW1
1B616	%MAX	1BC12	%ACOS	1E745	DISPROW2
1B62F	%MIN	1BC42	%ATAN	1E745	DISP@09
1B659	%0<	1BC59	%ANGLE	1E745	BIGDISPROW2
1B68C	%0=	1BCCF	%SINH	1E755	BIGDISPROW3
1B6BA	%0>	1BCFB	%COSH	1E755	DISPROW3
1B6F0	%0<>	1BD0E	%TANH	1E755	DISP@17
1B718	%0>=	1BD21	%ASINH	1E765	BIGDISPROW4
1B792	%<	1BD34	%ACOSH	1E765	DISP@25
1B7AB	%>	1BD47	%ATANH	1E765	DISPROW4
1B7C1	%>=	1BD5A	%EXPONENT	1E775	DISPROW5
1B7D7	%<=	1BD6D	%NFACT	1E785	DISPROW6

1E795 DISPROW7	20BBA EVALCRUNCH	269B1 ?ATTN_QUIT
1E7A5 DISPROW8	20CB9 EDITDECOMP$	269E8 DoCAlarmKey
1E94F HARDBUFF	20DC3 DECOMP$	26A01 CtlAlarm!
1E95F HARDBUFF2	21A2C #>$	26E33 'DROPFALSE
1E96F ABUFF	22F3B CREATEDIR	26E61 SetSomeRow
1E97F GBUFF	232A7 0LASTOWDOB!	26F25 StoVar
1E9F9 BLANKIT	232A7 0LastRomWrd!	26F75 STO_tTYPE
1EBCA XYGROBDISP	23460 BadIfEdit	26F7F tTYPE@
1EE86 HARDHEIGHT	234B0 NoExitAction	26F8E Xmin!
1F0EB HEIGHTENGROB	234C4 Box/StdLabel	26F93 Xmin@
1F2AE GROB>GDISP	23537 Std/BoxLabel	26FA2 Xmax!
1F37B KILLGDISP	23C2B Key>StdKeyOb	26FA7 Xmax@
1F3C6 RECLAIMDISP	23E15 DoBadKey	26FBB Ymin@
1F490 TOGDISP	23F4A 'DoBadKey	26FCF Ymax@
1F4A8 TOADISP	23F5E 'DoBadKeyT	26FE3 Xmin2@
1F4C2 GDISP?	23F90 H/W>KeyCode	26FF7 Xmax2@
1F523 WINDOWUP	23FC7 ModifierKey?	2700B Ymin2@
1F57B WINDOWDOWN	24008 ?CaseKeyDef	2701F Ymax2@
1F83F WINDOWLEFT	2403A ?CaseRomptr@	27033 HTick@
1FA0C WINDOWRIGHT	24067 ?NoTaskSwDef	27047 VTick@
1FB06 WINDOWXY	24376 DoRomPtrKey	2705B HZoom@
1FC49 WINDOWCORNER	24448 TakeOver	2706F VZoom@
1FCA3 PIXOFF3	245E2 IsTaskSwKey?	27083 Angle@
1FCB9 PIXON3	24E34 ParseExpr	27097 LastX@
1FCCF PIXOFF	25023 LockAlpha	270AB LastY@
1FCDE PIXON	25037 UnLockAlpha	270BF LastIndep@
1FE1A PIXON?3	25186 SLOW	270D3 LastEq@
1FE26 PIXON?	251A1 VERYSLOW	270E7 PlotFlag@
205CB DO>STR	251B1 VERYVERYSLOW	270FB Plot2Flg@
20652 DODISP	2523D InitMenu	2710F PlotCKSUM@
2074F setbeep	25F78 Ck&DecKeyLoc	27123 Split1CKSUM@
20782 ERRBEEP	26068 CodePl>%rc.p	27137 Split2CKSUM@
20898 RDUP	262DB WaitForKey	2714B TStart@
208ED RROLL	264C6 GETKEY*	2715F TStep@
2094F RPICK	2661F GETKEY	27173 TZoom@
209AE DO#EXIT	26793 ATTN?	27187 TFlags@
209C7 DO$EXIT	26943 KEYINBUFFER?	2719B Trow@
20AFF NULLID	269B1 ?ATTNQUIT	271AF Tcol@

271C3 TCol1@	28934 LastBut4	2B966 xC7
271D7 SplitRow@	2893B LastBut5	2B97D xC8
271EB BTRow1@	28942 LastBut6	2B994 xC9
271FF Format@	28949 LastBut7	2B9AB xC0
27213 Digits@	28950 LastBut8	2BDA2 xPVARS
27353 Root@	28957 LastBut9	2BF46 xCORR
27367 Slope@	2895E LastBut10	2BF61 xCOV
2737B Isect@	28965 LastBut11	2E5CB CopyRegCOb
2738F Area@	2896C LastBut12	2EC6F DefauPtXit
273A3 Extremum@	28973 LastBut13	2ECD8 AlDrawMenu
273DA ListOfEqs@	2897A LastBut14	2F763 DefauTrcInit
27416 funcCache@	28981 LastBut15	2F8BC SplitTraceInit
276CD StatType!	28988 LastBut16	2FCDB DispYFunc
276D2 StatType@	2898F LastBut17	2FF8E TraceY
276E6 SVarType@	28996 LastBut18	30473 DispXFunc
276FA StatPlot@	2899D LastBut19	347CC PTYPE>PINFO
2770E HisWidth@	289A4 LastBut20	352C0 DropJunk
27722 Hmin@	289AB LastBut21	35577 ParOuterLoop
27736 Hmax@	289B2 LastBut22	355AE POLSaveUI
2774A StatMark@	289B9 LastBut23	3565D POLSetUI
2775E StatModel@	289C0 LastBut24	35725 POLSetUIExt
27772 Stat2Flag@	289C7 LastBut25	35743 POLKeyUI
27786 ReCalcFlag@	289CE LastBut26	35798 POLErrorTrap
2779A StatFont@	289D5 LastBut27	357CA POLResUI&Err
277A4 StatMisc[]@	289DC LastBut28	357E3 POLRestoreUI
2783A StatFit@	289E3 LastBut29	35A6F TopOuterLoop
27B91 DoInPgmCxt	289EA LastBut30	35B58 TOLSaveUI
28037 DoInNotCxt	289F1 LastBut31	35BA3 TOLSetTopicUI
28154 CatNot	289F8 LastBut32	35C0C TOLSetViewUI
2815E CatThisCxt	28A1A LASTBUTN	35CD9 TOLKeyUI
281C7 CatPgm	28A40 DummyVar	35D2E TOLErrorTrap
28862 CalcDir+Offs	29B79 NullMenuKey	35D60 TOLResUI&Err
28879 AppDir+Offs	2B8DC xC1	35D79 TOLRestoreUI
28890 Contxt+Offs	2B8F3 xC2	35E0F ?ExitThisTop
28918 LastBut0	2B90A xC3	35F55 DA1OK?
2891F LastBut1	2B921 xC4	35FA5 DA2aOK?
28926 LastBut2	2B938 xC5	35FF5 DA2bOK?
2892D LastBut3	2B94F xC6	36045 DA2OK?

3605E	DA3OK?	3660A	ClrDA1Bad	3864B	INT_02
360AE	DAsOK?	36618	DA2aBad?	38650	INT_03
3611C	DA3OK?NOTIT	36626	SetDA2aBad	38655	INT_04
36162	DA2aLess1OK?	36634	ClrDA2aBad	3865A	INT_05
3617B	SetDA1Valid	36650	SetDA2bBad	3865F	INT_06
36194	SetDA2aValid	3665E	ClrDA2bBad	38664	INT_07
361A8	SetDA2bValid	3667A	SetDA3Bad	38669	INT_08
361C1	SetDA3Valid	366A4	SetDA1IsStat	3866E	INT_09
361D5	SetDA1Temp	366FB	SetNoRollDA2	38673	INT_0A
361EE	SetDA2aTemp	36A17	DefaultHint	38678	INT_0B
3621B	SetDA3Temp	36F79	KeyFace	3867D	INT_0C
3622F	SetDA2aEcho	37087	#NEG	38682	INT_0D
3624D	ClrDA1OK	3709C	#DIV	38687	INT_0E
36266	ClrDA2aOK	370C3	#MOD	3868C	INT_0F
3627F	ClrDA2bOK	370E8	#>=	38691	INT_10
36298	ClrDA2OK	37101	#<=	38696	INT_11
362AC	ClrDA3OK	37116	#ODD	3869B	INT_12
362DE	SetDA2Valid	37135	d>	386A0	INT_13
3630B	SetDA2NoCh	371A9	d<	386A5	INT_14
3631F	SetDA12NoCh	371BB	d>=	386AA	INT_15
36333	SetDA23NoCh	371D0	d<=	386AF	INT_16
36347	SetDA13NoCh	37205	d*	386B4	INT_17
3635B	SetDA12a3NoCh	372DF	dDIV	386B9	INT_18
3635B	SetDA12a3NCh	37303	dMOD	386BE	INT_19
3636F	SetDAsNoCh	37325	dmuldiv	386C3	INT_1A
36388	SetDA2Temp	374C0	d>%	386C8	INT_1B
3639C	SetDA12Temp	380CC	EnterTextView	386CD	INT_1C
363B0	SetDAsTemp	380E5	LeaveGraphView	386D2	INT_1D
363C9	AnyDABad?	380E5	LeaveTextView	386D7	INT_1E
36482	SetDA3ValidF	380FE	EnterGraphView	386DC	INT_1F
36554	SetDA1NoCh	381B5	topic_install	386E1	CAND
3657E	SetDA2aNoCh	382B4	topic_uninstall	3870B	COR
3659A	DA2bNoCh?	38304	DummyMenuErr	3871B	TopicVarN@
365A8	SetDA2bNoCh	38345	MiniVar	3873F	TopicVarN!
365B6	ClrDA2bNoCh	38381	EnsureMenuOff	38763	BinLookup
365D2	SetDA3NoCh	385D4	INT_NN	38BC8	Putbitmap
365EE	DA1Bad?	38641	INT_00	38C45	Orbitmap
365FC	SetDA1Bad	38646	INT_01	38C58	Xorbitmap

38C6B	NotAndbitmap	3AD12	xNEGNEG	3C07F	xRND
38C7E	Put5x7	3AD68	x-	3C111	xTRNC
3A3A8	xKILL	3AFCA	x/	3C1A3	xMOD
3A3BD	xOFF	3B178	x^	3C1F2	xMANT
3A3D2	xDOERR	3B36B	xXROOT	3C223	xD>R
3A3D2	xEXIT	3B4C3	xINVERSE	3C24F	xR>D
3A400	xERR0	3B528	xINV	3C27B	x>HMS
3A415	xERRN	3B5A0	xARG	3C2A7	xHMS>
3A42A	xERRM	3B5EF	xSIGN	3C2CB	xHMS+
3A43F	xEVAL	3B639	xSQRT	3C2E5	xHMS-
3A475	xIFTE	3B6E6	xSQ	3C307	xRNRM
3A569	xIFT	3B781	xSIN	3C33A	xCNRM
3A5C4	xSYSEVAL	3B7D5	xCOS	3C36D	xDET
3A614	xDISP	3B829	xTAN	3C3A0	xDOT
3A62E	xBEEP	3B87D	xSINH	3C3EC	xCROSS
3A648	x>NUM	3B8C7	xCOSH	3C430	xRSD
3A648	x->NUM	3B911	xTANH	3C452	x%
3A662	xLAST	3B95B	xASIN	3C4C9	x%T
3A777	xWAIT	3B9E1	xACOS	3C53B	x%CH
3A7C7	DOWAIT	3BA49	xATAN	3C5AD	xRAND
3A7DB	dowait	3BA93	xASINH	3C5C2	xRDZ
3A864	xCLLCD	3BAD8	xACOSH	3C5E4	xCOMB
3A879	xKEY	3BB4A	xATANH	3C624	xPERM
3A8C0	xCONT	3BBAD	xEXP	3C685	x<
3A8DD	alg=	3BBF7	xLN	3C685	x<?
3A8DD	x=	3BC69	xLOG	3C729	x>?
3A9B8	xNEG	3BCDB	xALOG	3C729	x>
3A9B8	xCHS	3BD25	xLNP1	3C7B9	x=?
3AA49	xABS	3BD5B	xEXPM	3C7B9	x==
3AA9F	xCONJ	3BD5B	xEXPM1	3C885	xR\8DP
3AAF5	xPI	3BD96	xFACT	3C935	xP\8DR
3AB17	xINF	3BE01	xIP	3CA30	xRE
3AB17	xMAXR	3BE3C	xFP	3CA86	xIM
3AB39	xEPS	3BE77	xFLOOR	3CAD2	xint
3AB39	xMINR	3BEB2	xCEIL	3CB4E	xDER
3AB5B	xCONSTANTe	3BEED	xXPON	3CCB5	xROOT
3AB7D	xi	3BF1E	xMAX	3CD50	xINTEGRAL
3AB9F	x+	3BFF9	xMIN	3D017	xWHERE

3D068	xQUOTE	3E700	xUNTIL	3F0C4	PrintVarOb
3D122	xAPPLY	3E710	xSTART	3F148	apndit
3D19D	xFCNAPPLY	3E7A8	xSTARTVAR	3F149	argswap
3D5B2	xCMDAPPLY	3E98E	xNEXT	3F151	filename
3D6AE	xEVAL>	3EABD	xSTEP	3F16C	onestring
3D7A7	xNOEVAL>	3EB17	xIFERR	3F170	pshtop&
3D80A	x->Q	3EBFB	xSILENT'	3F176	realPA
3D829	x->QPI	3EC15	xRPN->	3F177	resolved
3D898	xMATCHUP	3ED4D	x<<	3F1A2	tok;triand
3D8C6	xMATCHDN	3ED68	x>>	3F1A4	unroll2ND
3D905	xFORMUNIT	3ED7E	x'	3F1CF	XEQSETLIB
3D977	xPREDIV	3ED98	xENDTIC	3F1D0	numargs
3D9F3	xCOLCT	3EDB3	xWHILEEND	3F1D2	psh1&rev
3DA36	xEXPAN	3EDD3	xENDDO	3F1D3	roll3RD
3DA79	xRULES	3EE34	xERRTHEN	3F20A	Put3x5
3DAA0	xISOL	3EE91	xCASE	3F28E	CLOSEUART
3DB03	xQUAD	3EF1F	xTHENCASE	3F28F	1GETapndcpl
3DB51	xSHOW	3EFBF	xDIR	3F290	AllowPRLCD
3DBDC	xTAYLR	3F02B	DoRecv/GetOb	3F291	AllowPrlcdCl
3DC48	xFUNCTION	3F02C	DoSendOb	3F292	BREAK
3DC54	xPOLAR	3F040	OpenIO	3F293	DOFINISH
3DC60	xPARAMETRIC	3F043	PUTLIST	3F294	DoGetObFrSto
3DC72	xSAME	3F063	dvbind	3F295	DoIOErrAlert
3DC94	xAND	3F068	nultrior	3F296	DoIOStatusBox
3DD33	xOR	3F06A	psh1	3F297	DoRecvObFrEls
3DDBE	xNOT	3F06B	psh1top&	3F298	DoSendObToCDi
3DE2A	xXOR	3F06C	pshzer	3F299	DoSendObToDir
3DF0F	x#?	3F06F	pullpsh1&	3F29A	DoSendObToSto
3E049	x<=?	3F070	pullrev	3F29B	ElsieGet
3E0ED	x>=?	3F075	syminner&N	3F29C	ElsiePkt
3E416	xIF	3F077	top&	3F29D	ElsieSend
3E4AB	xTHEN	3F079	unsyminner	3F29E	GETSERIAL
3E54D	xELSE	3F07A	verysyminner	3F29F	GetBVars.1
3E568	xIFEND	3F07B	veryunsymin	3F2A0	GetObTypePr$
3E57E	xALG->	3F07E	&&	3F2A1	GetObType$
3E5C1	xWHILE	3F08F	FOURpsh	3F2A2	M+prep
3E634	xREPEAT	3F090	FalseFalse	3F2A3	OB>BAK
3E695	xDO	3F0A0	GetNextToken	3F2A4	OpenUart?Clr

3F2A5	PRINT	3F2CB	vunsymfcn	3F3C5	SetDAsBad
3F2A6	PUTSERIAL	3F2D8	DoSendObToEls	3F3C6	SetFAreaOK
3F2A7	PrintHist	3F31F	CHECKPICT	3F3C9	SetRedrawFlag
3F2A8	PrintLcd	3F320	CKGROBFITS	3F3CB	StoAplet
3F2A9	PrintVar	3F334	DOCLLCD	3F3CE	StripTicks
3F2AA	Port0Ids	3F336	DOERASE	3F3D0	TOGLINE
3F2AB	SEFINISH	3F338	DOLCD>	3F3D5	TURNMENUOFF
3F2AC	SWAPcompSWAP	3F345	Decomp%Short	3F3D6	TURNMENUON
3F2AD	SetEcma94	3F349	DispMsgBox	3F3E9	Update_tTYPE
3F2AE	StdIOProc	3F34C	DoAlert&Query	3F3EA	UserSto
3F2AF	UARTBUFLEN	3F34E	DoCApNoteV	3F3FD	DOLCD>g0
3F2B0	argswap&&	3F34F	DoCApPlotSV	3F401	DoAs2Col
3F2B1	argswapnext	3F350	DoCApSketchV	3F40D	GETXPOS
3F2B2	chooselst	3F351	DoCApSymbSV	3F40E	GETYPOS
3F2B3	docr	3F352	DoCApTableSV	3F410	GetLastNotThis
3F2B4	elsiename	3F363	DoQueryBox	3F417	InitIndep
3F2B5	metainsert	3F36A	EmptyList?	3F41D	PickedEqs@
3F2B6	newsymbPA	3F36F	FitLeftSmF	3F5A9	2ColChoose
3F2B7	oneexpr	3F375	GetDASpecFlag	3F5AB	GetApDirList
3F2B8	optfilename	3F383	InAplet?	3F5AC	InitListMets
3F2B9	prx1	3F384	LESetItem	3F5AD	SwapL/RMets
3F2BA	psh&	3F388	LINEOFF	3F5C0	ClrInAplet
3F2BB	psharg	3F389	LINEOFF3	3F5C1	DoDispField
3F2BC	pshmonop	3F38A	LINEON	3F5C4	InpFormVExit
3F2BD	roll2ND	3F38B	LINEON3	3F5C5	InpFormViewUI
3F2BE	roll4TH	3F392	MENUOFF?	3F5C6	SetFAreaBad
3F2BF	rollNTH	3F393	MakeNoteVTtl	3F5C9	InpFormVEntry
3F2C0	startSQFORM	3F394	MakeTitleBar	3F5CA	ChooseVEntry
3F2C1	syminner&	3F397	NEWINDEP	3F5CB	ChooseVExit
3F2C2	syminnertwo	3F3AA	PrepEdLKeyOb	3F5CC	ChooseViewUI
3F2C3	threeexprs	3F3AB	PutDASpecFlag	3F5CE	grobCheck
3F2C4	toktriand	3F3B7	RclAplet	3F5CF	grobCheckX
3F2C5	twoexprs	3F3B9	SCROLLDOWN	3F5D0	grobInvChk
3F2C6	twostrings	3F3BA	SCROLLUP	3F5D1	grobInvChkX
3F2C7	unroll3RD	3F3BB	SETPLOTENV	3F5D2	grobUnCheckX
3F2C8	unroll4TH	3F3C1	SafeCrunch%	3F5D3	grobInvUnChkX
3F2C9	unrollNTH	3F3C2	SafeDecomp	3F5D4	grobInvNoChk
3F2CA	unsymone	3F3C4	SelectAplet	3F5D7	grobQueryIcon

3F5D8	grobTitleBar	3F612	DoCApPlotV	3F63F	IFEDispHelp
3F5E7	DecompOb	3F613	DoCApSymbV	3F640	IFEDispLabel
3F5E8	Seq_eval0	3F614	DoCApTableV	3F641	IFEDispTitle
3F5E9	Seq_eval1	3F615	DoCurrAplet	3F642	IFEUnShowSel
3F5EA	Seq_eval2	3F616	DoDispBorder	3F643	IFEdLineMenu
3F5EB	Seq_evaln	3F617	DoDispList	3F644	IFMenu
3F5EC	GraphTableUI	3F618	DoDispPrompt	3F645	Id>ApNameId
3F5ED	GraphZoomUI	3F619	DoKeyCheck	3F646	InApletF?
3F5F0	EraseGraph	3F61A	DoKeyChoose	3F647	Init2ColMets
3F5F1	CHECKHEIGHT	3F61B	DoKeyChoos/Ck	3F648	Init_window
3F5F2	2CDispList	3F61E	DoSpecAlert	3F649	InvertField
3F5F3	2CKeyOK	3F61F	EmptyRList?	3F64A	IsIFMenu?
3F5F4	?AdjFocusPos	3F620	EraseGrob	3F64B	IsNullField?
3F5F5	?DispMoreU/D	3F622	FTypeFixedL?	3F64C	JUMPBOT
3F5F6	?FixFieldKeys	3F623	FTypeCheck?	3F64D	JUMPLEFT
3F5F7	?GetFObTypes	3F624	FTypeList?	3F64E	JUMPRIGHT
3F5F8	AddEq$	3F625	FTypeText?	3F64F	JUMPTOP
3F5F9	AngleField	3F626	FUNCPLOT	3F650	LEDispBorder
3F5FA	AngleLabel	3F627	FitRightSmF	3F654	LESetIDecomp
3F5FB	ApName$>Id	3F628	FuncPLoop	3F655	LESetRowWidth
3F5FC	ApNameId>$	3F629	FuncPSetup1	3F656	LHighlight
3F5FD	ApNameId>Id	3F62A	PlotOvViewUI	3F657	LINECHANGE
3F5FE	BlankClient	3F62B	ZoomAutoUI	3F658	LUnHighlight
3F603	Choose&DoTask	3F62F	GDISPCENTER	3F659	ListBar
3F604	ClearList0	3F630	BlankHelp	3F65A	ListNames
3F605	ClearList1	3F631	FAreaBad?	3F65B	MakeNumVTtl
3F606	ClearList2	3F632	GROB+#	3F65C	Make1stAplet
3F607	ClearList3	3F633	GROBBUFF!	3F65D	MakeNumSVTtl
3F608	ClearList4	3F634	GROBVIEW	3F65E	MakePlotSVTtl
3F609	ClearList5	3F635	GetChoiceFmt	3F65F	MakePlotVTtl
3F60A	ClearList6	3F636	GetChoiceList	3F660	MakeSketVTtl
3F60B	ClearList7	3F637	GetNextFId	3F661	MakeSymbSVTtl
3F60C	ClearList8	3F638	GetPrevFId	3F662	MakeSymbVTtl
3F60D	ClearList9	3F639	GtoField	3F663	MakeViewTitle
3F60E	ClrListUtil	3F63A	IFECheck	3F664	MoreDown?
3F60F	DOSHOWIT1	3F63B	IFEChoosByChr	3F665	MoreU/D?
3F610	DecompNoNL	3F63C	IFEChooseNext	3F666	MoreUp?
3F611	DoApletLib	3F63D	IFEDispClient	3F667	NextApOfType

3F668	NextTextLine	3F691	check_xrange	3F737	STTableExit
3F669	PlaceFVals	3F692	check_yrange	3F738	STTableHKeys
3F66A	PlotViewUI	3F693	grobMoreDown	3F739	STTableInit
3F66B	PrepCurrAplet	3F694	grobMoreUp	3F73B	SWAPStatFlag
3F66C	PurgeApletOb	3F695	grobNoMore	3F73B	SWAPStatFlags
3F66D	PurgeAplet	3F696	grobPOBoxP2	3F73C	SYMBNUMSOLVE
3F66E	RclAplet?Err	3F697	grobPOBoxP3	3F741	StoAns@Drp
3F66F	RclFieldVal	3F698	grobPOBoxP4	3F742	TBEdSfKeys
3F670	RclResetVal	3F699	grobPOBox2	3F743	TBErrorGrob
3F671	Replace_List	3F69A	grobPOBox3	3F744	TBFormat
3F672	SCROLLLEFT	3F69B	grobPOBox4	3F745	TBFuncDecomp
3F673	SCROLLRIGHT	3F69C	grobPOBox5	3F747	TBInv3x5C0
3F674	SV_?defined	3F69D	s1>	3F748	TBInv3x5C1
3F675	SV_actual	3F69E	PUTYMIN	3F749	TBInv3x5C2
3F676	SV_calledit	3F69F	PUTXMIN	3F74A	TBInv3x5C3
3F677	SV_getbody	3F6A0	GETYMAX	3F752	TBStdZooms
3F67B	SV_setgrob19	3F6A1	GETXMAX	3F78F	DOROOT
3F67C	SV_setitem0	3F6A2	GETYMIN	3F7A7	SysTime
3F67D	SaveListMets	3F6A3	GETXMIN	3F7A7	CLKTICKS
3F67E	Sel&DispNextF	3F6A9	NotesViewUI	3F7E7	!MATTRNnc
3F67F	SelNextField	3F6AA	IFSymbViewUI	3F804	#+#2-
3F680	SelPrevField	3F6AB	PlotSetViewUI	3F805	STDoMedium
3F681	SetFAreasBad	3F6AC	SeqTableViewUI	3F806	STDoSmall
3F682	SetFAreasOK	3F6AD	StatTableViewUI	3F807	STInitCols
3F683	SetInAplet	3F6AE	SolveNumViewUI	3F808	STJump
3F684	Set_window	3F6AF	SymbSetViewUI	3F809	STJumpN
3F685	StdTableViewUI	3F6B0	PictViewUI	3F80A	STSortCol
3F686	StdDecompNoNL	3F6B1	DoSaveCovWin	3F80B	!REDIMTEMP
3F687	StndXYCoord?	3F6B2	DoRestCovWin	3F80C	!REDIMUSER
3F688	SymbViewUI	3F6F1	Bubble	3F80D	DoResultTab
3F689	TOGLINE3	3F6FA	Disp5x7	3F80E	REDIMPREP
3F68A	TablSetViewUI	3F6FD	DoEditLCancel	3F80F	TBColL3x5
3F68B	TempMenuBuff	3F6FF	DoEditLOK	3F810	TBColL5x7
3F68C	WINDOWBOT?	3F701	DoEditLine	3F811	TBColR3x5
3F68D	WINDOWLEFT?	3F702	DoMatEdit	3F812	TBColR5x7
3F68E	WINDOWRIGHT?	3F712	MemStoAns	3F813	TBDrawF3x5
3F68F	WINDOWTOP?	3F735	STErrorGrob	3F814	TBDrawF5x7
3F690	WidthSmF	3F736	STTableDisp	3F815	TBInv5x7C0

3F816	TBInv5x7C1	80077	LoBatTime	8063D	SAVE_D0
3F817	TBInv5x7C2	80078	StartTime	80642	SAVE_OR
3F818	TBRollD3x5	80085	FailTime	80642	ORghost
3F819	TBRollD5x7	80092	TESTMSG	80645	DRSTART
3F81A	TBRollU3x5	800BE	SW_Image	8064A	DREND
3F81B	TBRollU5x7	800D4	SW_ETime	8064F	IREG
3F81C	TDDat3x5C1	800E1	PortStat	80652	SEMAPH
3F81D	TDDat3x5C2	800E2	Port1CRC	80654	IOSAVE
3F81E	TDDat3x5C3	800E6	AccessInit	80655	CSPEED
3F81F	TDDat3x5C4	800E8	COVERstate	8065A	INITEN
3F820	TDDat3x5C5	800EB	COVERsave	8065B	DISABLE_KBD
3F821	TDDatD3x5	800F5	IRAMBUFF	8065B	HANDSHK
3F822	TDDatL3x5	80127	IRAMBUFF2	8065C	KEYSTATE
3F823	TDDatR3x5	8030E	GraphPrtHook	80669	KEYBUFFER
3F824	TDDatU3x5	8030E	IRAMBEND	80669	INPUTSTREAM
3F825	TDDataF3x5	80319	uart_buffer	8068B	POPPEDKEY
3F826	blackbox	80519	uart_buf_end	8068D	DISP1CTLg
3F8D0	DemoFrames	8051B	uart_error	80692	LINENIBSg
3F8D1	DoDemo	8051C	uart_buf_st	80695	DISP2CTLg
3F8D2	UnpackGrob	8051E	uart_handshk	8069A	LINECOUNTg
3F8FB	{NoteText}	8051F	uart_modes	8069C	GreyOn?
3F900	{SketchSet}	80520	uart_parity	8069C	Stk0save
3F903	{NumVars}	80521	uart_timeout	8069D	GreyScr1
80000	RAMSTART	80523	IOCNIB	806A1	Stk1save
80000	CMOS	80524	CONFRAM	806A2	GreySoft1
80000	HARDROMEND	8052B	CONFTAB	806A6	Stk2save
80005	IRAMMASK	80536	PORT0EOS	806A7	GreyScr2
8000A	HOMEMASK	8053B	PORT1EOS	806AB	Stk3save
8000F	HRAMEND	80540	PORT2EOS	806AC	GreySoft2
80010	FAILSTK1	805DB	INTRAM	806B0	Stk4save
80022	FAILSTK2	805EB	SAVE_MODES	806B1	GreyScr3
80034	FAILSTK3	805F0	SAVE_C[A]	806B5	Stk5save
80046	FAILSTK4	805F5	SAVE_A	806B6	GreySoft4
80058	NEXTIRQ	80605	SAVE_ST	806BA	R2[A]save
80065	TIMECRC	80608	SAVE_B	806BF	R2[S]save
80069	TIMExmit	80618	SAVE_D	806C0	R1[A]save
80069	TIMEOUT	80628	SAVE_R0	806C5	SAVE_B0
80076	TIMEOUTCLK	80638	SAVE_PC	806C6	SAVE_LC

806C8	SAVE_LN	80770	OSAVE	80824	MenuExitAct
806CB	SAVE_OFFSET	80775	LASTARG	80829	LASTROMWDOB
806D0	VDISP2	80775	LASTARG1	8082E	KeyOb
806D5	ADISP	8077A	LASTARG2	80833	FlagMBox
806DA	SYSUPSTART	8077F	LASTARG3	80838	ViewMBox
806DA	VDISP	80784	LASTARG4	8083D	ProgMBox
806DA	VDISP1	80789	LASTARG5	80842	Title
806DF	VDISP3	8078E	leeway	80847	HiLitePtr
806E4	GDISP	80793	ITEM1STATE	8084C	WindowPtr
806E9	TEMPOB	80798	HISTORY1	80851	HStackPtr
806EE	TEMPTOP	8079D	HISTORY2	80856	HStackTop
806F3	RSKTOP	807A2	HISTORY3	8085B	GraphContext
806F8	DSKTOP	807A7	HISTORY4	8086A	TopicVar1
806FD	EDITLINE	807AC	PDCHXS	8086F	TopicVar2
80702	TEMPENV	807B1	PDCSYMB	80874	TopicVar3
80707	DOLPENV	807B1	KERMERRM	80879	TopicVar4
8070C	TOUCHTAB	807B6	PAINTTREE	8087E	TopicVar5
80711	USEROB	807BB	EXITMSG	80883	TopicVar6
80716	ROMPARTS	807C0	AppDisplay	80888	TopicVar7
8071B	CONTEXT	807C5	AppKeys	8088D	TopicVar8
80720	STOPSIGN	807CA	AppExitCond	80892	TopicVar9
80725	UserKeys	807CF	AppError	80897	TopicVar10
8072A	ALARMS	807D4	AppSuspend	8089C	TopicVar11
8072F	FSTVGERPTR	807D9	AppResume	808A1	TopicVar12
8072F	VSTACK	807DE	AppCursor	808A6	TopicVar13
80734	CALCCXT	807E3	AppDoKeyOb	808AB	TopicVar14
80739	PGMCXT	807E8	CtlAlarm	808B0	TopicVar15
8073E	NOTESCXT	807ED	MenuDef	808B5	TopicVar16
80743	apletPTR	807F2	LastMenuDef	808BA	TopicVar17
80748	funcPTR	807F7	MenuData	808BF	TopicVar18
8074D	polarPTR	807FC	MenuRowAct	808C4	TopicVar19
80752	paramPTR	80801	LabelDef	808C9	TopicVar20
80757	seqPTR	80806	MenuKeyNS	808CE	TopicVar21
8075C	statPTR	8080B	MenuKeyLS	808D3	TopicVar22
80761	solvePTR	80810	MenuKeyRS	808D8	TopicVar23
80766	otherPTR	80815	ReviewKey	808DD	TopicVar24
8076B	INTRPPTR	8081A	LastContext	808E2	TopicVar25
8076B	OBUPSTART	8081F	TrackAct	808E7	TopicVar26

808EC	TopicVar27	809AA	TopicVar65	80A68	TOLVar12
808F1	TopicVar28	809AF	TopicVar66	80A6D	TOLVar13
808F6	TopicVar29	809B4	TopicVar67	80A72	TOLVar14
808FB	TopicVar30	809B9	TopicVar68	80A77	TOLVar15
80900	TopicVar31	809BE	TopicVar69	80A7C	TOLVar16
80905	TopicVar32	809C3	TopicVar70	80A81	TOLVar17
8090A	TopicVar33	809C8	TopicVar71	80A86	TOLVar18
8090F	TopicVar34	809CD	TopicVar72	80A8B	TOLVar19
80914	TopicVar35	809D2	TopicVar73	80A90	TOLVar20
80919	TopicVar36	809D7	TopicVar74	80A95	TOLVar21
8091E	TopicVar37	809DC	TopicVar75	80A9A	TOLVar22
80923	TopicVar38	809E1	TopicVar76	80A9F	TOLVar23
80928	TopicVar39	809E6	TopicVar77	80AA4	TOLVar24
8092D	TopicVar40	809EB	TopicVar78	80AA9	TOLVar25
80932	TopicVar41	809F0	TopicVar79	80AAE	TOLVar26
80937	TopicVar42	809F5	TopicVar80	80AB3	TOLVar27
8093C	TopicVar43	809FA	TopicVar81	80AB8	TOLVar28
80941	TopicVar44	809FF	TopicVar82	80ABD	TOLVar29
80946	TopicVar45	80A04	TopicVar83	80AC2	TOLVar30
8094B	TopicVar46	80A09	TopicVar84	80AC7	TOLVar31
80950	TopicVar47	80A0E	TopicVar85	80ACC	TOLVar32
80955	TopicVar48	80A13	TopicVar86	80AD1	TOLVar33
8095A	TopicVar49	80A18	TopicVar87	80AD6	TOLVar34
8095F	TopicVar50	80A1D	TopicVar88	80ADB	TOLVar35
80964	TopicVar51	80A22	TopicVar89	80AE0	TOLVar36
80969	TopicVar52	80A27	TopicVar90	80AE5	TOLVar37
8096E	TopicVar53	80A2C	TopicVar91	80AEA	TOLVar38
80973	TopicVar54	80A31	TOLVar1	80AEF	TOLVar39
80978	TopicVar55	80A36	TOLVar2	80AF4	TOLVar40
8097D	TopicVar56	80A3B	TOLVar3	80AF9	TOLVar41
80982	TopicVar57	80A40	TOLVar4	80AFE	TOLVar42
80987	TopicVar58	80A45	TOLVar5	80B03	TOLVar43
8098C	TopicVar59	80A4A	TOLVar6	80B08	TOLVar44
80991	TopicVar60	80A4F	TOLVar7	80B0D	TOLVar45
80996	TopicVar61	80A54	TOLVar8	80B12	TOLVar46
8099B	TopicVar62	80A59	TOLVar9	80B17	TOLVar47
809A0	TopicVar63	80A5E	TOLVar10	80B1C	TOLVar48
809A5	TopicVar64	80A63	TOLVar11	80B21	TOLVar49

80B26	TOLVar50	80BE4	TOLVar88	80CA2	TOLVar126
80B2B	TOLVar51	80BE9	TOLVar89	80CA7	TOLVar127
80B30	TOLVar52	80BEE	TOLVar90	80CAC	TOLVar128
80B35	TOLVar53	80BF3	TOLVar91	80CB1	TOLVar129
80B3A	TOLVar54	80BF8	TOLVar92	80CB6	TOLVar130
80B3F	TOLVar55	80BFD	TOLVar93	80CBB	TOLVar131
80B44	TOLVar56	80C02	TOLVar94	80CC0	TOLVar132
80B49	TOLVar57	80C07	TOLVar95	80CC5	TOLVar133
80B4E	TOLVar58	80C0C	TOLVar96	80CCA	TOLVar134
80B53	TOLVar59	80C11	TOLVar97	80CCF	TOLVar135
80B58	TOLVar60	80C16	TOLVar98	80CD4	TOLVar136
80B5D	TOLVar61	80C1B	TOLVar99	80CD9	TOLVar137
80B62	TOLVar62	80C20	TOLVar100	80CDE	TOLVar138
80B67	TOLVar63	80C25	TOLVar101	80CE3	TOLVar139
80B6C	TOLVar64	80C2A	TOLVar102	80CE8	TOLVar140
80B71	TOLVar65	80C2F	TOLVar103	80CED	TOLVar141
80B76	TOLVar66	80C34	TOLVar104	80CF2	TOLVar142
80B7B	TOLVar67	80C39	TOLVar105	80CF7	TOLVar143
80B80	TOLVar68	80C3E	TOLVar106	80CFC	TOLVar144
80B85	TOLVar69	80C43	TOLVar107	80D01	TOLVar145
80B8A	TOLVar70	80C48	TOLVar108	80D06	TOLVar146
80B8F	TOLVar71	80C4D	TOLVar109	80D0B	TOLVar147
80B94	TOLVar72	80C52	TOLVar110	80D10	TOLVar148
80B99	TOLVar73	80C57	TOLVar111	80D15	TOLVar149
80B9E	TOLVar74	80C5C	TOLVar112	80D1A	TOLVar150
80BA3	TOLVar75	80C61	TOLVar113	80D1F	TOLVar151
80BA8	TOLVar76	80C66	TOLVar114	80D24	TOLVar152
80BAD	TOLVar77	80C6B	TOLVar115	80D29	TOLVar153
80BB2	TOLVar78	80C70	TOLVar116	80D2E	TOLVar154
80BB7	TOLVar79	80C75	TOLVar117	80D33	TOLVar155
80BBC	TOLVar80	80C7A	TOLVar118	80D38	TOLVar156
80BC1	TOLVar81	80C7F	TOLVar119	80D3D	TOLVar157
80BC6	TOLVar82	80C84	TOLVar120	80D42	TOLVar158
80BCB	TOLVar83	80C89	TOLVar121	80D47	TOLVar159
80BD0	TOLVar84	80C8E	TOLVar122	80D4C	TOLVar160
80BD5	TOLVar85	80C93	TOLVar123	80D51	TOLVar161
80BDA	TOLVar86	80C98	TOLVar124	80D56	TOLVar162
80BDF	TOLVar87	80C9D	TOLVar125	80D5B	TOLVar163

80D60	TOLVar164	80E1E	TOLVar202	80EC6	SysNib7
80D65	TOLVar165	80E23	TOLVar203	80EC7	SysNib8
80D6A	TOLVar166	80E28	TOLVar204	80EC8	SysNib9
80D6F	TOLVar167	80E2D	TOLVar205	80EC9	EDITFLAG
80D74	TOLVar168	80E32	TOLVar206	80EC9	SysNib10
80D79	TOLVar169	80E37	TOLVar207	80EC9	EDITLFLAG
80D7E	TOLVar170	80E3C	TOLVar208	80ECA	ParenModFLAG
80D83	TOLVar171	80E41	TOLVar209	80ECA	SysNib11
80D88	TOLVar172	80E46	TOLVar210	80ECB	SysNib12
80D8D	TOLVar173	80E4B	TOLVar211	80ECC	SysNib13
80D92	TOLVar174	80E50	TOLVar212	80ECD	SizeMLDisp
80D97	TOLVar175	80E55	TOLVar213	80ECD	SysNib14
80D9C	TOLVar176	80E5A	TOLVar214	80ECE	SysNib15
80DA1	TOLVar177	80E5F	TOLVar215	80ECF	SysNib16
80DA6	TOLVar178	80E64	TOLVar216	80ED0	SysNib17
80DAB	TOLVar179	80E69	CatalogCache	80ED1	SysNib18
80DB0	TOLVar180	80E6E	Clipboard	80ED2	SysNib19
80DB5	TOLVar181	80E73	FindPattern	80ED3	SysNib20
80DBA	TOLVar182	80E78	ReplacePatte	80ED4	AppCount
80DBF	TOLVar183	80E7D	ObjectU1	80ED6	ITEM1LINES
80DC4	TOLVar184	80E82	ObjectU2	80ED7	VIEWLEVEL
80DC9	TOLVar185	80E87	ObjectU3	80EDC	DEPTHSAVE
80DCE	TOLVar186	80E8C	ObjectU4	80EE1	RNSEED
80DD3	TOLVar187	80E91	OBUPEND	80EF0	SAVECLK
80DD8	TOLVar188	80E91	ObjectU5	80EF1	ALARMSDUE
80DDD	TOLVar189	80E96	SYSNOUPSTART	80EF2	PASTDUE
80DE2	TOLVar190	80E96	RAMEND	80EF3	DOUSEALARM
80DE7	TOLVar191	80E9B	AVMEM	80EF4	NOALARMSRV
80DEC	TOLVar192	80EA0	LANGUAGE	80EFF	LPD_HIST
80DF1	TOLVar193	80EA5	ERROR	80F00	ANNUNCIATORS
80DF6	TOLVar194	80EAB	ATTNFLG	80F02	SystemFlags
80DFB	TOLVar195	80EB0	FIRSTPROC	80F12	FLAG_SYSTEM2
80E00	TOLVar196	80EC0	SysNib1	80F22	UserFlags
80E05	TOLVar197	80EC1	SysNib2	80F32	FLAG_USER2
80E0A	TOLVar198	80EC2	SysNib3	80F42	ELEMENT
80E0F	TOLVar199	80EC3	SysNib4	80F44	FIRSTCHAR
80E14	TOLVar200	80EC4	SysNib5	80F49	CR_COUNT
80E19	TOLVar201	80EC5	SysNib6	80F4E	STACKNUM

80F53	TOPLINE	80FEB	KEYLIST	810AC	BEGX
80F59	HISTORYLEVEL	80FF0	KEYLOCK	810B1	ENDX
80F5A	LASTARGCOUNT	80FF1	ACCUM	810B6	BEG
80F5B	LASTARGf	80FF3	COLWIDTH	810BB	END
80F5C	LASTERROR	80FF5	ENTRWISE	810C0	T_ECRAN
80F61	CURSOREPOSN	80FF6	PARENCOUNT	810C0	SizeCLScreen
80F61	CURSOR	80FF8	STRETCHCOUNT	810E8	HashCLE
80F66	CURSORPART	80FFA	ClkOnNib	810E8	TAB_CMD
80F66	CURSORROW	80FFB	XmitSrcvTOut	8125A	T_BLOC
80F6B	CURSORPOSN	80FFD	DelayCt	8125F	CHECK_VAL
80F6B	CURSOROFFSET	80FFF	GCOLCOUNT	81264	CHECK_VAL2
80F6D	CURSORSTATE	81001	COLCOUNT	81269	SavTEMPENV
80F6E	CURSORCHR	81003	PrtStatus	81269	SAUV_80702
80F70	CURSORGROB	81006	IOCsave	8126E	SavFIRSTCHAR
80F98	CURSORX	81007	LineByteCt	8126E	SAUV_80865
80F9D	CURSORY	81009	FifoByteCt	81273	CHECK_TEXTE
80FA2	CURRENTMENU	8100B	LastPrntTime	81273	CheckCLE
80FA4	MENULEVEL	81016	PFIFO	81278	SAUV_MATRIX
80FA9	OLDMENU	81026	MenuRow	81278	SavMatrix
80FAB	T1COUNT	8102B	EqPtr	812A0	SizeLine
80FAC	PADCOUNT	81030	KeyRomPtr0	812A0	T_LIGNE
80FAD	GARBSCRATCH1	8103B	KeyRomPtr1	812A5	WidthScreen
80FB2	GARBSCRATCH2	81046	KeyRomPtr2	812A5	T_LARGEUR
80FB7	SAVECROSS	81051	KeyRomPtr3	812AA	NbFont
80FC1	PADJSAVE1	8105C	KeyRomPtr4	812AA	NB_FONTE
80FC2	PADJSAVE2	81067	KeyRomPtr5	812AF	VERIF_CARD
80FCC	KERMMODE	81072	KeyRomPtr6	812B4	SWITCH
80FCD	DcompWidth	8107D	LastMenuRow	812C3	MINI_FONT.OBJ
80FCF	FONTHEIGHT	81082	FlashPtrBkp	812C3	MiniFontObj
80FD0	FONTWIDTH	8108E	HeaderHeight	812CF	MINI_FONT
80FD1	FONTCOUNT	8108E	T_HEADER	812CF	MiniFont
80FD4	NODECOUNT	81093	StackHeight	818CF	SavChars
80FD7	OBTREELEN	81093	NB_LIGNE	818CF	SAUV_CHARS
80FDA	LASTOP	81098	FontHeight	818EE	FreeRoom
80FDB	LEFTTREE	81098	H_FONTE	818F3	SAUV_REGA
80FDE	RIGHTTREE	8109D	TYPE_HEADER	818F3	SavRegA
80FE1	PARENTTREE	810A2	BEGIN_REL	818F8	SavRegB
80FE4	PRECSTACK	810A7	END_REL	818F8	SAUV_REGB

818FD SavRegC	86028 misc1_f_s	064004 ~xBLANKGROB
818FD SAUV_REGC	8602D misc2_f_s	0000A1 ~DoMenuKey1N
81902 SavRegD	86032 misc3_f_s	0010A1 ~DoMenuKey2N
81902 SAUV_REGD	86037 KSTATEVGER	0020A1 ~DoMenuKey3N
81907 SAUV_REGD1	86047 LastKey	0030A1 ~DoMenuKey4N
81907 SavRegD1	86049 LastKeyTime	0040A1 ~DoMenuKey5N
8190C SavRegisters	86051 BounceTiming	0050A1 ~DoMenuKey6N
8190C SAUV_REGISTR	86059 CatalogEntry	0090A1 ~UpArrow
81971 @FONTE	8605E FROMPTAB0_15	00B0A1 ~NSKey3.2
81971 ArryFont	860AE FROMPTABPTR	00C0A1 ~NSKey3.3
8201D TAB_FONTE	860B3 CurROMBank1	00D0A1 ~LArrow
8201D HashArryFont	860B8 CurROMBank2	00E0A1 ~DArrow
8221D SavMisc	860BD CurRAMBank1	00F0A1 ~RArrow
8221D SAUV_DIVERS	860C2 CurRAMBank2	0100A1 ~NSKey4.1
8229E GROBSCR1	860C7 CurRAMBank3	0110A1 ~NSKey4.2
822B2 SCREEN1	860CC FlashROMTAB2	0120A1 ~NSKey4.3
822B2 ECRAN	8611C RESRAMEND0	0130A1 ~NSKey4.4
82B32 GROBSCR2	8611D RESRAMEND	0140A1 ~NSKey4.5
82B46 SCREEN2	8611D ROMPTAB	0160A1 ~NSKey5.2
833C6 GROBSCR3	8611D FlashROMPTAB	0170A1 ~NSKey5.3
833DA SCREEN3	90000 HARDRAMEND	0180A1 ~NSKey5.4
83C5A GROBSCR4	058002 ~xFmList	0190A1 ~NSKey5.5
83C6E SCREEN4	059002 ~xFmMat	01A0A1 ~NSKey6.1
844EE GROBSCR5	07E002 ~rpnDER	01B0A1 ~NSKey6.2
84502 SCREEN5	07F002 ~rpnINTG	01C0A1 ~NSKey6.3
84D82 FONTE_SYSTEM	080002 ~rpnWHERE	01D0A1 ~NSKey6.4
84D82 SystemFont	081002 ~rpnAPPLY	01E0A1 ~NSKey6.5
85F94 RealX	083002 ~COMPLEXDUMM	01F0A1 ~NSKey7.1
85FA9 RealY	084002 ~POLARDUMMY	0200A1 ~NSKey7.2
85FBE CplxX	000004 ~xxSIZE	0210A1 ~NSKey7.3
85FE3 CplxY	001004 ~xxPOS	0220A1 ~NSKey7.4
86008 DIGITS	049004 ~*xH	0230A1 ~NSKey7.5
8600D UserInt1	04A004 ~*xW	0240A1 ~NSKey8.1
86012 UserInt2	050004 ~xERASEA	0250A1 ~NSKey8.2
86017 UserInt1g	060004 ~xDISPLAY>	0260A1 ~NSKey8.3
8601C UserInt2g	061004 ~x>DISPLAY	0270A1 ~NSKey8.4
86021 nb_line_f_s	062004 ~xPLOT>	0280A1 ~NSKey8.5
86026 has_font_f_s	063004 ~x>PLOT	02A0A1 ~NSKey9.2

02B0A1 ~NSKey9.3	01E0A2 ~LSKey6.5	0150A4 ~ANSKey5.1
02C0A1 ~NSKey9.4	01F0A2 ~LSKey7.1	0160A4 ~ANSKey5.2
02D0A1 ~NSKey9.5	0200A2 ~LSKey7.2	0170A4 ~ANSKey5.3
02E0A1 ~AttentionKe	0210A2 ~LSKey7.3	0180A4 ~ANSKey5.4
0320A1 ~Enter/Again	0220A2 ~LSKey7.4	0190A4 ~ANSKey5.5
0330A1 ~PlotViewKey	0230A2 ~LSKey7.5	01A0A4 ~ANSKey6.1
0340A1 ~SymbViewKey	0240A2 ~LSKey8.1	01B0A4 ~ANSKey6.2
0350A1 ~TableViewKe	0250A2 ~LSKey8.2	01C0A4 ~ANSKey6.3
0360A1 ~TopicLibKey	0260A2 ~LSKey8.3	01D0A4 ~ANSKey6.4
0000A2 ~LSKey1.1	0270A2 ~LSKey8.4	01E0A4 ~ANSKey6.5
0010A2 ~LSKey1.2	0280A2 ~LSKey8.5	01F0A4 ~ANSKey7.1
0020A2 ~LSKey1.3	0290A2 ~LSKey9.1	0200A4 ~ANSKey7.2
0030A2 ~LSKey1.4	02A0A2 ~LSKey9.2	0210A4 ~ANSKey7.3
0040A2 ~LSKey1.5	02B0A2 ~LSKey9.3	0220A4 ~ANSKey7.4
0050A2 ~LSKey1.6	02C0A2 ~LSKey9.4	0230A4 ~ANSKey7.5
0060A2 ~LSKey2.1	02D0A2 ~LSKey9.5	0240A4 ~ANSKey8.1
0070A2 ~LSKey2.2	0330A2 ~DoExponent	0250A4 ~ANSKey8.2
0080A2 ~LSKey2.3	0000A4 ~ANSKey1.1	0260A4 ~ANSKey8.3
0090A2 ~FarUArrow	0010A4 ~ANSKey1.2	0270A4 ~ANSKey8.4
00A0A2 ~LSKey3.1	0020A4 ~ANSKey1.3	0280A4 ~ANSKey8.5
00B0A2 ~LSKey3.2	0030A4 ~ANSKey1.4	0290A4 ~ANSKey9.1
00C0A2 ~LSKey3.3	0040A4 ~ANSKey1.5	02A0A4 ~ANSKey9.2
00D0A2 ~FarLArrow	0050A4 ~ANSKey1.6	02B0A4 ~ANSKey9.3
00E0A2 ~FarDArrow	0060A4 ~ANSKey2.1	02C0A4 ~ANSKey9.4
00F0A2 ~FarRArrow	0070A4 ~ANSKey2.2	02D0A4 ~ANSKey9.5
0100A2 ~LSKey4.1	0080A4 ~ANSKey2.3	0000A5 ~ALSKey1.1
0110A2 ~LSKey4.2	0090A4 ~ANSKey2.4	0010A5 ~ALSKey1.2
0120A2 ~LSKey4.3	00A0A4 ~ANSKey3.1	0020A5 ~ALSKey1.3
0130A2 ~LSKey4.4	00B0A4 ~ANSKey3.2	0030A5 ~ALSKey1.4
0140A2 ~LSKey4.5	00C0A4 ~ANSKey3.3	0040A5 ~ALSKey1.5
0150A2 ~LSKey5.1	00D0A4 ~ANSKey3.4	0050A5 ~ALSKey1.6
0160A2 ~LSKey5.2	00E0A4 ~ANSKey3.5	0060A5 ~ALSKey2.1
0180A2 ~LSKey5.4	00F0A4 ~ANSKey3.6	0070A5 ~ALSKey2.2
0190A2 ~LSKey5.5	0100A4 ~ANSKey4.1	0080A5 ~ALSKey2.3
01A0A2 ~LSKey6.1	0110A4 ~ANSKey4.2	0090A5 ~ALSKey2.4
01B0A2 ~LSKey6.2	0120A4 ~ANSKey4.3	00A0A5 ~ALSKey3.1
01C0A2 ~LSKey6.3	0130A4 ~ANSKey4.4	00B0A5 ~ALSKey3.2
01D0A2 ~LSKey6.4	0140A4 ~ANSKey4.5	00C0A5 ~ALSKey3.3

00D0A5 ~ALSKey3.4	0050AB ~xRANK	0320AB ~xCOT
00E0A5 ~ALSKey3.5	0060AB ~xLSQ	0330AB ~xSEC
00F0A5 ~ALSKey3.6	0070AB ~xEGV	0340AB ~xCSC
0100A5 ~ALSKey4.1	0080AB ~xEGVL	0350AB ~xACOT
0110A5 ~ALSKey4.2	0090AB ~xSVD	0360AB ~xACSC
0120A5 ~ALSKey4.3	00A0AB ~xSVL	0370AB ~xASEC
0130A5 ~ALSKey4.4	00B0AB ~xLU	0380AB ~xRECURSE
0140A5 ~ALSKey4.5	00C0AB ~xQR	0390AB ~xPOLYFORM
0150A5 ~ALSKey5.1	00D0AB ~xLQ	03A0AB ~xVERSION
0160A5 ~ALSKey5.2	00E0AB ~xSCHUR	03B0AB ~XEQSYMLIN
0170A5 ~ALSKey5.3	00F0AB ~xRREF	03C0AB ~xRANM
0180A5 ~ALSKey5.4	0100AB ~xPROOT	03D0AB ~x>ROW
0190A5 ~ALSKey5.5	0110AB ~xPCOEF	03E0AB ~xROW>
01A0A5 ~ALSKey6.1	0120AB ~xPEVAL	03F0AB ~x>COL
01B0A5 ~ALSKey6.2	0130AB ~xSEQ	0400AB ~xCOL>
01C0A5 ~ALSKey6.3	0140AB ~x\9BLIST	0410AB ~x>DIAG
01D0A5 ~ALSKey6.4	0150AB ~x\85LIST	0420AB ~xDIAG>
01E0A5 ~ALSKey6.5	0160AB ~x\9CLIST	0430AB ~xROW-
01F0A5 ~ALSKey7.1	0170AB ~xCONCAT	0440AB ~xROW+
0200A5 ~ALSKey7.2	0180AB ~xREVLIST	0450AB ~xCOL-
0210A5 ~ALSKey7.3	0190AB ~xSORT	0460AB ~xCOL+
0220A5 ~ALSKey7.4	01A0AB ~xxDEG	0470AB ~xRSWP
0230A5 ~ALSKey7.5	01B0AB ~xxRAD	0480AB ~xCSWP
0240A5 ~ALSKey8.1	01C0AB ~xxGRAD	0490AB ~xRCI
0250A5 ~ALSKey8.2	0210AB ~xFRACTION	04A0AB ~xRCIJ
0260A5 ~ALSKey8.3	0220AB ~xSTAIRSTEP	04B0AB ~XEQDIAG>L
0270A5 ~ALSKey8.4	0230AB ~xCOBWEB	04C0AB ~XEQDIAG>R
0280A5 ~ALSKey8.5	0240AB ~xHIST	04D0AB ~XEQRANM
0290A5 ~ALSKey9.1	0250AB ~xBOXW	04E0AB ~xCHOOSE
02A0A5 ~ALSKey9.2	0290AB ~xPOWERFIT	04F0AB ~xMSGBOX
02B0A5 ~ALSKey9.3	02A0AB ~xQUADFIT	0500AB ~xXSEND
02C0A5 ~ALSKey9.4	02B0AB ~xCUBICFIT	0510AB ~xXRECV
02D0A5 ~ALSKey9.5	02C0AB ~xLOGISFIT	0520AB ~xHEAD
0000AB ~xLININ	02D0AB ~xUSERFIT	0530AB ~xTAIL
0010AB ~xCOND	02E0AB ~xStat1Var	0540AB ~xDOSUBS
0020AB ~xTRACE	02F0AB ~xStat2Var	0550AB ~xNSUB
0030AB ~xSRAD	0300AB ~xxITERATE	0560AB ~xENDSUB
0040AB ~xSNRM	0310AB ~xMKMAT	0570AB ~xSTREAM

0580AB ~ChkDaList	0020C2 ~laEGVL	0280C2 ~laQhB
0590AB ~xDOLIST	0030C2 ~laEGVL%%	0290C2 ~laSETDIAG
05A0AB ~xPoly	0040C2 ~laEGV	02A0C2 ~laRQF
05B0AB ~dopolyz	0050C2 ~laSCHUR	02B0C2 ~laQ2hX
05C0AB ~dopolydiv	0060C2 ~laEgHF	02C0C2 ~laRSVF
05D0AB ~dopcoeff	0070C2 ~laECQhQAQh	02D0C2 ~laQRrank
05E0AB ~metapolyz	0080C2 ~laEgQkHA	02E0C2 ~laQRSVecUp
05F0AB ~polyprep	0090C2 ~laEgRSchur	02F0C2 ~laQRminSV
0600AB ~poly+	00A0C2 ~laEgIsoVal	0300C2 ~laQRmaxSV
0610AB ~polyneg	00B0C2 ~laEgRQRI	0310C2 ~rMAKEPCOPY
0620AB ~poly*	00C0C2 ~laEgQRIk0	0320C2 ~rPACKARRYD
0630AB ~*spec	00D0C2 ~laEgRQRIk1	0330C2 ~MAKEEPIDN
0640AB ~*ord	00E0C2 ~laEgRWilk3	0340C2 ~laSNORM
0650AB ~sn*spec	00F0C2 ~laEgM*RG3	0350C2 ~laSNORM%%
0660AB ~ns*spec	0100C2 ~laEgWilk2	0360C2 ~laRANK
0670AB ~ss*spec	0110C2 ~laEgM*G2	0370C2 ~laSVL
0680AB ~poly^	0120C2 ~laEgValr	0380C2 ~laSVD
0690AB ~bad^	0130C2 ~laEgRotR	0390C2 ~laSvdUBD
06A0AB ~poly^#	0140C2 ~laSchur2	03A0C2 ~laSvdLtUBD
06B0AB ~dopval	0150C2 ~laEgVecR	03B0C2 ~laSvdUQhQA
06C0AB ~pdiv	0160C2 ~laEgVcSngl	03C0C2 ~laSvdAPPhV
06D0AB ~resymb	0170C2 ~laEgVcUrhs	03D0C2 ~laSvdAPk
06E0AB ~poly/	0180C2 ~laEgVcPair	03E0C2 ~laSvdGPROT
0000B9 ~MiscIFMsg	0190C2 ~laEgSclCls	03F0C2 ~laSvdBDQR
0000BB ~StatIFMsg	01A0C2 ~laEgCSchur	0400C2 ~laSvdQR2x2
0000BE ~PlotIFMsg	01B0C2 ~laEgCQRI	0410C2 ~laSvdFDirC
0000BF ~SolveIFMsg	01C0C2 ~laEgCQRIk1	0420C2 ~laSvdBDirC
0000C0 ~proot_c	01D0C2 ~laEgVecC	0430C2 ~laSvdGShft
0010C0 ~proot	01E0C2 ~laEgPrep	0440C2 ~laSvdQRSF
0020C0 ~proot_r	01F0C2 ~laFPMUTE	0450C2 ~laSvdQRSB
0030C0 ~peval	0200C2 ~laBPMUTE	0460C2 ~laSvdQRF
0040C0 ~pcoef	0210C2 ~laFSCALE	0470C2 ~laSvdQRB
0050C0 ~pcoefacc1	0220C2 ~laLSQ	0480C2 ~laSvdCROTR
0060C0 ~pcoefacc2	0230C2 ~laUserQR	0490C2 ~laSvdCROTL
0070C0 ~PolyNSymbP	0240C2 ~laUserLQ	04A0C2 ~laSvdPSort
0080C0 ~PolyNSymbF	0250C2 ~lauserQR	04B0C2 ~laSvdPrep
0000C2 ~laSRAD	0260C2 ~laQRF	04C0C2 ~laTRACE
0010C2 ~laVMAX%%	0270C2 ~laQhA	04D0C2 ~laCOND

04E0C2 ~laCONDdone	0150C3 ~laROW>	0220E8 ~INTEMPOB?
04F0C2 ~laVMAXJRP	0160C3 ~laCOL>	0230E8 ~moresub
0500C2 ~laUserLU	0170C3 ~la>DIAG	0240E8 ~morerepl
0510C2 ~laDT	0180C3 ~laDIAG>	0250E8 ~mrepln
0520C2 ~laRefineDT	0000E8 ~doseq	0260E8 ~mrepll
0530C2 ~laSV	0010E8 ~dosecntuple	0270E8 ~mrepll+
0540C2 ~laSVc	0020E8 ~dontuple	0280E8 ~msubln
0550C2 ~laIV	0030E8 ~dontuple#	0290E8 ~msubnl
0560C2 ~laIVc	0040E8 ~doidseqn	02A0E8 ~msubll
0570C2 ~laIVF	0050E8 ~seqid	02B0E8 ~msubll+
0580C2 ~laULHSVF	0060E8 ~::args	02C0E8 ~msubnn
0590C2 ~laLsdScale	0070E8 ~ptrargs	02D0E8 ~copysub
05A0C2 ~laScIntRnd	0080E8 ~dosecseqn	02E0E8 ~arryspec
05B0C2 ~laRREF	0090E8 ~udfargs	02F0E8 ~etorc
05C0C2 ~laRedRow	00A0E8 ~doptrseqn	0300E8 ~LIXRecv
05D0C2 ~laRedHere?	00B0E8 ~doseqn	0310E8 ~LIXSend
05E0C2 ~laEGetTiny	00C0E8 ~doseqn#	0000F0 ~INTGSIN
0000C3 ~laRANM	00D0E8 ~docmdlist	0010F0 ~I:-InvSin
0010C3 ~laRanInt	00E0E8 ~seqnargs	0020F0 ~INTGCOS
0020C3 ~laRSWP	00F0E8 ~dolatorre	0030F0 ~I:LnTan
0030C3 ~laRSWP2	0100E8 ~elsielists?	0040F0 ~INTGTAN
0040C3 ~laCSWP	0110E8 ~dolatorre+	0050F0 ~INTGASIN
0050C3 ~laRCI	0120E8 ~dolatorre2	0060F0 ~INTGACOS
0060C3 ~laRCIJ	0130E8 ~?NULLSETDIM	0070F0 ~INTGATAN
0070C3 ~la-ROW	0140E8 ~sllatorre	0080F0 ~INTGSINH
0080C3 ~laVec-	0150E8 ~lslatorre	0090F0 ~I:-InvSinh
0090C3 ~la-COL	0160E8 ~lolatorre	00A0F0 ~I:LnTanh
00A0C3 ~la+ROW	0170E8 ~ollatorre	00B0F0 ~INTGCOSH
00B0C3 ~la+ROWs	0180E8 ~ollatorre+	00C0F0 ~I:Tanh
00C0C3 ~la+RCsLP	0190E8 ~dolist+	00D0F0 ~INTGTANH
00D0C3 ~la+COL	01A0E8 ~dosort	00E0F0 ~INTGEXPM
00E0C3 ~la+COLs	01B0E8 ~{id}>{$}	00F0F0 ~INTGALOG
00F0C3 ~la+ELEMr	01C0E8 ~{lam}>{$}	0100F0 ~INTGLN
0100C3 ~la+ELEMc	01D0E8 ~{$}>{id}	0110F0 ~INTGLOG
0110C3 ~la+ELEM	01E0E8 ~{$}>{lam}	0120F0 ~INTGINV
0120C3 ~la>ROW	01F0E8 ~DoNumeric:	0130F0 ~I:Atan
0130C3 ~la>ELEM	0200E8 ~UseHidden{}	0140F0 ~INTGSQ
0140C3 ~la>COL	0210E8 ~NotHidden	0150F0 ~INTGSQRT

0160F0	˜I:Asinh	03C0F0 ˜m1/[]ˆ	0620F0 ˜REIM[]*
0170F0	˜I:Acosh	03D0F0 ˜m1/[]E	0630F0 ˜m->TRG
0180F0	˜INTGSIGN	03E0F0 ˜m-[]L	0640F0 ˜m->DEFSIN
0190F0	˜INTGDER	03F0F0 ˜m-[]*	0650F0 ˜m->DEFCOS
01A0F0	˜mT->=	0400F0 ˜m-[]/	0660F0 ˜m->DEFTAN
01B0F0	˜m<-T=	0410F0 ˜m-[]+	0670F0 ˜m->DEFASIN
01C0F0	˜mAFlq	0420F0 ˜m-[]-	0680F0 ˜m->DEFACOS
01D0F0	˜mAFqq	0430F0 ˜mEˆ*	0690F0 ˜m->DEFATAN
01E0F0	˜mAFrq	0440F0 ˜mEˆ/	06A0F0 ˜m->DEFSINH
01F0F0	˜m<->+	0450F0 ˜mE[]ˆ	06B0F0 ˜m->DEFCOSH
0200F0	˜m<->-	0460F0 ˜mL*ˆ	06C0F0 ˜m->DEFTANH
0210F0	˜m<->*	0470F0 ˜mL[]*	06D0F0 ˜m->DEFASINH
0220F0	˜m<->/	0480F0 ˜mL[]/	06E0F0 ˜m->DEFACOSH
0230F0	˜m<-A-+	0490F0 ˜m<-M+*	06F0F0 ˜m->DEFATANH
0240F0	˜m<-A--	04A0F0 ˜m<-M-*	0700F0 ˜mSIN+
0250F0	˜m<-A/*	04B0F0 ˜adjdivsign	0710F0 ˜mCOS+
0260F0	˜m<-A//	04C0F0 ˜adjsign	0720F0 ˜mTAN+
0270F0	˜m<-Aˆ*	04D0F0 ˜adjdiv	0730F0 ˜mSINH+
0280F0	˜mA->+-	04E0F0 ˜m<-M*ˆ	0740F0 ˜mCOSH+
0290F0	˜mA->--	04F0F0 ˜m<-M/ˆ	0750F0 ˜mTANH+
02A0F0	˜mA->*/	0500F0 ˜m<-M*E	0760F0 ˜covmanCOL
02B0F0	˜mA->//	0510F0 ˜m<-M/E	0770F0 ˜WHEREFCNAPP
02C0F0	˜mA->ˆˆ	0520F0 ˜m<-M+L	0780F0 ˜WHEREDER
02D0F0	˜mD->/+	0530F0 ˜m<-M-L	0790F0 ˜WHEREIFTE
02E0F0	˜mD->/-	0540F0 ˜mM->op	07A0F0 ˜WHEREWHERE
02F0F0	˜mD->ˆ+	0550F0 ˜m<T>+	07B0F0 ˜covWSPLIT
0300F0	˜mD->ˆ-	0560F0 ˜m<T>*	07C0F0 ˜WHEREINTG
0310F0	˜mD->E+	0570F0 ˜m<-T+-	07D0F0 ˜WHERESUM
0320F0	˜mD->E-	0580F0 ˜m<-T*/	07E0F0 ˜XSTOCHECK
0330F0	˜mD->L*	0590F0 ˜m<-[+-	07F0F0 ˜XSTOCHECK10
0340F0	˜mD->L/	05A0F0 ˜m<-[*/	0800F0 ˜covLBSTO
0350F0	˜m[]CHS*	05B0F0 ˜m]->+-	0810F0 ˜XEQXDPTCH
0360F0	˜m[]CHS/	05C0F0 ˜m]->*/	0820F0 ˜Xcont
0370F0	˜m[]CHSL	05D0F0 ˜m->[]<-+-	0830F0 ˜CHECKEXISTS
0380F0	˜m[]INVˆ	05E0F0 ˜m->[]<-*/	0840F0 ˜covmetaLIBS
0390F0	˜m[]INVE	05F0F0 ˜mCONJ[]	0850F0 ˜XRCLp?
03A0F0	˜m1/[]*	0600F0 ˜mRE[]	0860F0 ˜XRCLpNL
03B0F0	˜m1/[]/	0610F0 ˜mIM[]	0870F0 ˜XRCLpL

0880F0 ~XRCLp*	0D30F0 ~covINVTANH	02C0F1 ~xPOLARSYMB
0890F0 ~XEVALp?	0D40F0 ~covINVALOG	02D0F1 ~xPARAMSYMB
08A0F0 ~XEVALp*	0D50F0 ~covINVEXPM1	02E0F1 ~xSEQSYMB
08B0F0 ~XPURGEp?	0000F1 ~nFUNCTION	02F0F1 ~xSTATSYMB
08C0F0 ~XPURGEp*	0010F1 ~nSOLVE	0300F1 ~xSTAT2SYMB
08D0F0 ~convertaddr	0020F1 ~nPOLAR	0310F1 ~xSOLVESYMB
08E0F0 ~XRCLp?acc>	0030F1 ~nPARAMETRIC	0320F1 ~xDFLTNOTE
08F0F0 ~COLCTDER	0040F1 ~nSCATTER	0330F1 ~xDFLTPICT
0900F0 ~COLCTINTG	0050F1 ~nHISTOGRAM	0340F1 ~topic_NONE
0910F0 ~COLCTSUM	0060F1 ~nSEQUENCE	0350F1 ~view_NONE
0920F0 ~COLCTIFTE	0070F1 ~nBOXWHISKER	0360F1 ~view01_CAPLET
0930F0 ~COLCTQUOTE	0080F1 ~xFUNCTAB	0360F1 ~view01_CAPL
0940F0 ~COLCTFCNAP	0090F1 ~xPOLARTAB	0370F1 ~view23_CAPL
0960F0 ~covD/DCROSS	00A0F1 ~xPARAMTAB	0370F1 ~view23_CAPLET
0970F0 ~covD/DDOT	00B0F1 ~xSEQTAB	0380F1 ~view45_CAPLET
0B90F0 ~covD/DINTG	00C0F1 ~xSTATTAB	0380F1 ~view45_CAPL
0BC0F0 ~covMANMENU+	00D0F1 ~xSTAT2TAB	0390F1 ~view6_CAPLE
0BD0F0 ~covMANMENU*	00E0F1 ~xSOLVETAB	03A0F1 ~view7_CAPLE
0BF0F0 ~covMANEXP	00F0F1 ~nFUNCTOPIC	03B0F1 ~topic_CAPLE
0C00F0 ~covMANMENUL	0100F1 ~nPOLARTOPIC	03B0F1 ~topic_CAPLET
0C10F0 ~covMANCSIV	0110F1 ~nPARAMTOPIC	03C0F1 ~xREADNOTE
0C20F0 ~covMANMENUE	0120F1 ~nSEQTOPIC	03D0F1 ~xREADPICT
0C30F0 ~covMANMENUC	0130F1 ~nSTATTOPIC	0000F2 ~xU1
0C40F0 ~covMANTRG	0140F1 ~nSOLVETOPIC	0010F2 ~idU1
0C50F0 ~covMANATG	01E0F1 ~xUndefined	0020F2 ~xU2
0C60F0 ~covINV+	01F0F1 ~GetTypeText	0030F2 ~idU2
0C70F0 ~covINV-	0200F1 ~DefauStatTy	0040F2 ~xU3
0C80F0 ~covINV=	0210F1 ~DefauStat2T	0050F2 ~idU3
0C90F0 ~covINV*	0220F1 ~nPTYPE>PINF	0060F2 ~xU4
0CA0F0 ~covINV/	0230F1 ~StdApEntry	0070F2 ~idU4
0CB0F0 ~covINV^	0240F1 ~SolveApEntr	0080F2 ~xU5
0CC0F0 ~covINV^X	0250F1 ~SETTOPICLAM	0090F2 ~idU5
0CD0F0 ~covINVEXP	0260F1 ~CLRTOPICLAM	00A0F2 ~xU6
0CE0F0 ~covINVSIN	0270F1 ~GETPLTLABLS	00B0F2 ~idU6
0CF0F0 ~covINVCOS	0280F1 ~POINTERR	00C0F2 ~xU7
0D00F0 ~covINVTAN	0290F1 ~POINTEXIT	00D0F2 ~idU7
0D10F0 ~covINVSINH	02A0F1 ~runalias?	00E0F2 ~xU8
0D20F0 ~covINVCOSH	02B0F1 ~xFUNCSYMB	00F0F2 ~idU8

0100F2 ~xU9	0360F2 ~xY4	05C0F2 ~xR7
0110F2 ~idU9	0370F2 ~idY4	05D0F2 ~idR7
0120F2 ~xU0	0380F2 ~xX5	05E0F2 ~xR8
0130F2 ~idU0	0390F2 ~idX5	05F0F2 ~idR8
0140F2 ~xF1	03A0F2 ~xY5	0600F2 ~xR9
0150F2 ~idF1	03B0F2 ~idY5	0610F2 ~idR9
0160F2 ~xF2	03C0F2 ~xX6	0620F2 ~xR0
0170F2 ~idF2	03D0F2 ~idX6	0630F2 ~idR0
0180F2 ~xF3	03E0F2 ~xY6	0640F2 ~xD1
0190F2 ~idF3	03F0F2 ~idY6	0650F2 ~xD2
01A0F2 ~xF4	0400F2 ~xX7	0660F2 ~xD3
01B0F2 ~idF4	0410F2 ~idX7	0670F2 ~xD4
01C0F2 ~xF5	0420F2 ~xY7	0680F2 ~xD5
01D0F2 ~idF5	0430F2 ~idY7	0690F2 ~xD6
01E0F2 ~xF6	0440F2 ~xX8	06A0F2 ~xD7
01F0F2 ~idF6	0450F2 ~idX8	06B0F2 ~xD8
0200F2 ~xF7	0460F2 ~xY8	06C0F2 ~xD9
0210F2 ~idF7	0470F2 ~idY8	06D0F2 ~xD0
0220F2 ~xF8	0480F2 ~xX9	06E0F2 ~xH1
0230F2 ~idF8	0490F2 ~idX9	06F0F2 ~xH2
0240F2 ~xF9	04A0F2 ~xY9	0700F2 ~xH3
0250F2 ~idF9	04B0F2 ~idY9	0710F2 ~xH4
0260F2 ~xF0	04C0F2 ~xX0	0720F2 ~xH5
0270F2 ~idF0	04D0F2 ~idX0	0730F2 ~xS1
0280F2 ~xX1	04E0F2 ~xY0	0740F2 ~xS2
0290F2 ~idX1	04F0F2 ~idY0	0750F2 ~xS3
02A0F2 ~xY1	0500F2 ~xR1	0760F2 ~xS4
02B0F2 ~idY1	0510F2 ~idR1	0770F2 ~xS5
02C0F2 ~xX2	0520F2 ~xR2	0780F2 ~xs1
02D0F2 ~idX2	0530F2 ~idR2	0790F2 ~xs2
02E0F2 ~xY2	0540F2 ~xR3	07A0F2 ~xs3
02F0F2 ~idY2	0550F2 ~idR3	07B0F2 ~xs4
0300F2 ~xX3	0560F2 ~xR4	07C0F2 ~xs5
0310F2 ~idX3	0570F2 ~idR4	07D0F2 ~xn1
0320F2 ~xY3	0580F2 ~xR5	07E0F2 ~xn2
0330F2 ~idY3	0590F2 ~idR5	07F0F2 ~xn3
0340F2 ~xX4	05A0F2 ~xR6	0800F2 ~xn4
0350F2 ~idX4	05B0F2 ~idR6	0810F2 ~xn5

0820F2	~xa	0A90F2 ~xMEANX	0D10F2 ~xg0
0830F2	~xb	0AA0F2 ~xSX	0D20F2 ~xg1
0840F2	~xc	0AB0F2 ~xSX2	0D30F2 ~xg2
0850F2	~xd	0AC0F2 ~xMEANY	0D40F2 ~xg3
0860F2	~xE	0AD0F2 ~xSY	0D50F2 ~xg4
0870F2	~xf	0AE0F2 ~xSY2	0D60F2 ~xg5
0880F2	~xg	0AF0F2 ~xSXY	0D70F2 ~xg6
0890F2	~xh	0B20F2 ~xRELERR	0D80F2 ~xg7
08A0F2	~xI	0B30F2 ~xz0	0D90F2 ~xg8
08B0F2	~xj	0B40F2 ~xz1	0DA0F2 ~xg9
08C0F2	~xk	0B50F2 ~xz2	0DB0F2 ~xAns
08D0F2	~xl	0B60F2 ~xz3	0DC0F2 ~xFIT
08E0F2	~xm	0B70F2 ~xz4	0DD0F2 ~xHAngle
08F0F2	~xn	0B80F2 ~xz5	0DE0F2 ~xHFormat
0900F2	~xo	0B90F2 ~xz6	0DF0F2 ~xHDigits
0910F2	~xp	0BA0F2 ~xz7	0E00F2 ~xRadixMark
0920F2	~xq	0BB0F2 ~xz8	0E10F2 ~xIerr
0930F2	~xr	0BC0F2 ~xz9	0E20F2 ~xTime
0940F2	~xs	0BD0F2 ~xm0	0E30F2 ~xDate
0950F2	~xt	0BE0F2 ~xm1	0E40F2 ~xAngle
0960F2	~xu	0BF0F2 ~xm2	0E50F2 ~xAxes
0970F2	~xv	0C00F2 ~xm3	0E60F2 ~xGrid
0980F2	~xw	0C10F2 ~xm4	0E70F2 ~xConnect
0990F2	~xx	0C20F2 ~xm5	0E80F2 ~xHighRes
09A0F2	~xy	0C30F2 ~xm6	0E90F2 ~xSimult
09B0F2	~xz	0C40F2 ~xm7	0EA0F2 ~xRecenter
09C0F2	~xtheta	0C50F2 ~xm8	0EB0F2 ~xInvCursor
09D0F2	~xMEANS	0C60F2 ~xm9	0EC0F2 ~xLabels
09E0F2	~xTOTS	0C70F2 ~xl0	0ED0F2 ~xTracing
09F0F2	~xSVARS	0C80F2 ~xl1	0EE0F2 ~xCoord
0A10F2	~xSSDEV	0C90F2 ~xl2	0EF0F2 ~xXmin
0A20F2	~xPSDEV	0CA0F2 ~xl3	0F00F2 ~xXmax
0A30F2	~xNS	0CB0F2 ~xl4	0F10F2 ~xYmin
0A40F2	~xMINS	0CC0F2 ~xl5	0F20F2 ~xYmax
0A50F2	~xMAXS	0CD0F2 ~xl6	0F30F2 ~xIndep
0A60F2	~xMEDIAN	0CE0F2 ~xl7	0F40F2 ~xXcross
0A70F2	~xQ1	0CF0F2 ~xl8	0F50F2 ~xYcross
0A80F2	~xQ3	0D00F2 ~xl9	0F60F2 ~xHTick

0F70F2 ~xVTick	11D0F2 ~xNmax	015701 ~xxPLOT>
0F80F2 ~xHzoom	11E0F2 ~xSeqPlot	016701 ~xx>PLOT
0F90F2 ~xVzoom	11F0F2 ~xStatMode	017701 ~xx>GROB
0FA0F2 ~xNumStart	1200F2 ~xStatPlot	018701 ~xxARC
0FB0F2 ~xNumStep	1210F2 ~xHisWidth	019701 ~xxPRSTC
0FC0F2 ~xNumType	1220F2 ~xHmin	01A701 ~xxPRVAR
0FD0F2 ~xNumIndep	1230F2 ~xHmax	01B701 ~xxPRDISPLAY
0FE0F2 ~xNumZoom	1240F2 ~xS1mark	01C701 ~xxINPUT
0FF0F2 ~xNumRow	1250F2 ~xS2mark	01D701 ~xxBREAK
1000F2 ~xNumCol	1260F2 ~xS3mark	01E701 ~xxSTOP
1010F2 ~xNumFont	1270F2 ~xS4mark	01F701 ~xxFREEZE
1020F2 ~xFormat	1280F2 ~xS5mark	020701 ~xxROW-
1030F2 ~xDigits	1290F2 ~xS1fit	021701 ~xxROW+
1040F2 ~xNoteText	12A0F2 ~xS2fit	022701 ~xxCOL-
1050F2 ~xPage	12B0F2 ~xS3fit	023701 ~xxCOL+
1060F2 ~xPageNum	12C0F2 ~xS4fit	024701 ~xxRSWP
1070F2 ~idE1	12D0F2 ~xS5fit	025701 ~xxCSWP
1080F2 ~idE2	000701 ~xxWSLOG	026701 ~xxRCI
1090F2 ~idE3	001701 ~xxTSTR	027701 ~xxRCIJ
10A0F2 ~idE4	002701 ~xxDISP	028701 ~xxTO
10B0F2 ~idE5	003701 ~xxBEEP	029701 ~xxRUNPGM
10C0F2 ~idE6	004701 ~xxWAIT	02A701 ~xxMATEDIT
10D0F2 ~idE7	005701 ~xxRDZ	02B701 ~xxMKGROB
10E0F2 ~idE8	006701 ~xxSUB	02C701 ~xxZEROGROB
10F0F2 ~idE9	007701 ~xxREPL	02D701 ~xxLIBEVAL
1100F2 ~idE0	008701 ~xxRDM	02E701 ~xxSYSEVAL
1110F2 ~xRoot	009701 ~xxERASE	02F701 ~xxSETSAMPLE
1120F2 ~xIsect	00A701 ~xxERASEPLOT	030701 ~xxSETFREQ
1130F2 ~xExtremum	00B701 ~xxPIXON	031701 ~xxDO1VSTATS
1140F2 ~xArea	00C701 ~xxPIXOFF	032701 ~xxSETINDEP
1150F2 ~xSlope	00D701 ~xxLINE	033701 ~xxSETDEPEND
1160F2 ~xTmin	00E701 ~xxTLINE	034701 ~xxDO2VSTATS
1170F2 ~xTmax	00F701 ~xxBOX	035701 ~xxSELECT
1180F2 ~xTStep	010701 ~xxGOR	036701 ~xxRANM
1190F2 ~xThetaMin	011701 ~xxGXOR	037701 ~xxVERSION
11A0F2 ~xThetaMax	012701 ~xxGROBNOT	038701 ~xxDEMO
11B0F2 ~xThetaStep	013701 ~xxDISPLAY>	039701 ~xxMEM
11C0F2 ~xNmin	014701 ~xx>DISPLAY	03A701 ~xxRULES

03C701	~xxSETVIEWS	03F701 ~xPINIT	042701 ~xxGETKEY
03D701	~xCHECK	040701 ~xxMSGBOX	043701 ~xxHELP
03E701	~xUNCHECK	041701 ~xxCHOOSE	044701 ~xHELP

Entry Index

$

%

&

'

*

2

3

4

5

6

7

B

C

D

E

G

M

P

R

S

T

U

V

Y

Z