

# The flexisym package

Morten Høgholm  
Maintainers: `mh.ctan@gmail.com`

2008/08/08 v0.97a

## User's guide

For now, the user's guide is in breqn.

## Implementation

### 1 flexisym

```
1 <*package>
2 \RequirePackage{expl3}[2009/08/05]
3 \ProvidesExplPackage{flexisym}{2013/03/16}{0.97c}{Make math characters macros}
4
5 \edef\do{%
6   \noexpand\AtEndOfPackage{%
7     \catcode\number'\="=\number\catcode'\"
8     \relax
9   }%
10 }
11 \do \let\do\relax
12 \catcode'\="=12
13 \let\@sym\@gobble
14 \DeclareOption{robust}{%
15   \def\@sym#1{%
16     \ifx\protect\@typeset@protect \else\protect#1\exp_after:wN\use_none:nnnn\fi
17   }%
18 }
```

The math groups (mg) here relate to `\textfontn`.

```
19 \def\mg@bin{2}% binary operators
20 \def\mg@rel{2}% relations
21 %%\def\mg@nre{B}% negated relations
22 \def\mg@del{3}% delimiters
23 %%\def\mg@arr{B}% arrows
```

```

24 \def\mg@acc{0}% accents
25 \def\mg@cop{3}% cumulative operators (sum, int)
26 \def\mg@latin{1}% (Latin) letters
27 \def\mg@greek{1}% (lowercase) Greek
28 \def\mg@Greek{0}% (capital) Greek
29 %\def\mg@bflatin{4}% bold upright Latin letters ?
30 %\def\mg@Bbb{B}% blackboard bold
31 \def\mg@cal{2}% script/calligraphic
32 %\def\mg@frak{5}% Fraktur letters
33 \def\mg@digit{0}% decimal digits % 1 = oldstyle, 0 = capital

```

This is how we insert mathchars. The command has three arguments: class, fam and slot position and so it is always given as hexadecimal. This way of separating things should make it easier to get this to work with XeTeX et al. which have many more slot positions

```

34 \cs_set_protected:Nn \math_char:Nnn {
35   \tex_mathchar:D \__int_eval:w " #1#2#3 \__int_eval_end:
36 }

```

Delimiters and radicals are similar except here we have both small and large variant. Radicals have no class.

```

37 \cs_set_protected:Nn \math_delimiter:NnnNn {
38   \tex_delimiter:D \__int_eval:w " #1#2#3#4#5 \__int_eval_end:
39 }
40 \cs_set_protected:Nn \math_radical:NnnNn {
41   \tex_radical:D \__int_eval:w " #1#2#3#4 \__int_eval_end:
42 }
43 \cs_set_protected:Nn \math_accent:Nnnn {
44   \tex_mathaccent:D \__int_eval:w " #1 #2 #3 \__int_eval_end: {#4}
45 }
46
47 \let\sumlimits\displaylimits
48 \let\intlimits\nolimits
49 \let\namelimits\displaylimits

```

TeX defines eight types of atoms.

0. Ordinary
1. Operators
2. Binary
3. Relation
4. Open
5. Close
6. Punctuation
7. Inner

TeX defines eight math classes.

0. Ordinary
1. Operators
2. Binary
3. Relation
4. Open
5. Close
6. Punctuation
7. Variable family

flexisym/breqn extends this to types of classes.

0. Ordinary: (Ord), Bidirectional delimiters (DeB), Radicals (Rad), Accented items (Acc)
1. Operators: Cumulative Operators sum-like (COs), Cumulative Operators integral-like (COi)
2. Binary: (Bin)
3. Relation: (Rel), Arrow delimiters (DeA)
4. Open: (DeL)
5. Close (DeR)
6. Punctuation: (Pun)
7. Variable family: (Var)

Here's an overview of what we are about to do. Math chars of each type as defined by us need a basic operation for inserting it. We will call that function `\math_bsym_⟨type⟩:Nn`. Next there are compund symbols for each type which we name `\math_bcsym_⟨type⟩:Nn`. Also, there is inline mode and display mode which are different. We will call them for `\math_ism_⟨type⟩:Nn` `\math_icsym_⟨type⟩:Nn` for inline mode and `\math_dsym_⟨type⟩:Nn` and `\math_dcsym_⟨type⟩:Nn`. The code uses the terms `\math_sym_⟨type⟩:Nn` and `\math_csym_⟨type⟩:Nn` for the current meaning of things. First up the basic definitions. #1 is the math group it is from and #2 is the slot position.

```

50 \cs_new:Npn \math_bsym_Ord:Nn {\math_char:NNn 0 }% \m@Ord
51 \cs_new:Npn \math_bsym_Var:Nn {\math_char:NNn 7 }% \m@Var
52 \cs_new:Npn \math_bsym_Bin:Nn {\math_char:NNn 2}% \m@Bin
53 \cs_new:Npn \math_bsym_Rel:Nn {\math_char:NNn 3 }% \m@Bin
54 \cs_new:Npn \math_bsym_Pun:Nn {\math_char:NNn 6 }% \m@Pun
55 \cs_new:Nn \math_bsym_COs:Nn { \math_char:NNn 1#1{#2} \sumlimits}% \m@COs
56 \cs_new:Nn \math_bsym_COi:Nn { \math_char:NNn 1#1{#2} \intlimits}% \m@COi

```

```

57 \cs_new:Nn \math_bsym_DeL:Nn { \math_sd_del_aux:Nnn 4 #1{#2} }%\m@DeL
58 \cs_new:Nn \math_bsym_DeR:Nn { \math_sd_del_aux:Nnn 5 #1{#2} }%\m@DeR
59 \cs_new:Nn \math_bsym_DeB:Nn { \math_sd_del_aux:Nnn 0 #1{#2} }%\m@DeB
60 \cs_new:Nn \math_bsym_DeA:Nn { \math_sd_del_aux:Nnn 3 #1{#2} }%\m@DeA
61 \cs_new:Nn \math_bsym_Rad:Nn { \math_sd_rad_aux:Nn #1{#2} }%\m@Rad
62 \cs_new:Npn \math_bsym_Acc:Nn #1#2#3#4 {\math_accent:Nnn #1#2{#3}{#4}}% name is wrong
63

```

Next is somewhat complicated internally. The way it is done is that delimiters and radicals need information about the smallest version of the symbol. If this smallest delimiter (SD) is defined, then use it. We have these functions to help us return the number. Extract the numbers to use and stick a function in front of it.

Code changed because now we require the smallest delimiter to be defined (it may be the same, no problem in that). So the two arguments present in `\math_bsym_DeL:Nn` are the location of extensible version (where the font will do the rest for us automatically). For each delimiter, a pointer is defined using the extensible characters family and slot as name and value equal to family and position of the smallest version. For  $\langle$  in standard L<sup>A</sup>T<sub>E</sub>X this is `{del}{00}` and `{OT1}{28}` respectively. Hence, `\math_bsym_DeL:Nn \mg@del {00}` must expand to `\math_delimiter:NnnNn 4 \mg@OT1 {28}\mg@del{00}`. So first expand away to get to the smallest version. Then call next function which shuffles the arguments around.

```

64 \cs_set:Npn \math_sd_del_aux:Nnn #1#2#3{
65   \exp_args:Nf \math_sd_del_auxi:nN {\use:c{sd@#2#3}} #1 #2{#3}
66 }
67 \cs_set:Npn \math_sd_del_auxi:nN #1#2{ \math_delimiter:NnnNn #2 #1 }

```

Same for radicals.

```

68 \cs_set:Npn \math_sd_rad_aux:Nn #1#2{
69   \exp_args:Nf \math_sd_rad_auxi:n {\use:c{sd@#1#2}} #1 {#2}
70 }
71 \cs_set:Npn \math_sd_rad_auxi:n #1{ \math_radical:NnnNn #1 }
72
73
74 % \cs_set:Npn \math_sd_aux:nn #1#2 {
75 %   %\exp_args:Nnf \use:nn { #1} { \math_sd_auxi:Nn #2 }
76 %   \exp_args:Nnf \use:nn { #1} { \use:c{sd@\use:nn#2} }
77 % }
78 % \cs_set:Npn \math_sd_auxi:Nn #1#2 {
79 %   \cs_if_free:cTF {sd@#1#2}
80 %   { #1{#2} }
81 %   { \use:c{sd@#1#2} }
82 % }

```

compound symbols here

```

83 \cs_set_protected:Npn \math_bcsym_Ord:Nn #1#2{\@syntype\mathord{\OrdSymbol{#2}}}%\@symOrd
84 \cs_set_protected:Npn \math_bcsym_Var:Nn #1#2{\@syntype\mathord{\OrdSymbol{#2}}}%\@symVar
85 \cs_set_protected:Npn \math_bcsym_Bin:Nn #1#2{\@syntype\mathbin{\OrdSymbol{#2}}}%\@symBin
86 \cs_set_protected:Npn \math_bcsym_Rel:Nn #1#2{\@syntype\mathrel{\OrdSymbol{#2}}}%\@symRel
87 \cs_set_protected:Npn \math_bcsym_Pun:Nn #1#2{\@syntype\mathpunct{\OrdSymbol{#2}}}%\@symPun

```

```

88 \cs_set_protected:Npn \math_bcsym_COi:Nn #1#2{\@syntype\mathop{\OrdSymbol{#2}\intlimits}}%\@
89 \cs_set_protected:Npn \math_bcsym_COs:Nn #1#2{\@syntype\mathop{\OrdSymbol{#2}\sumlimits}}%\@
90 \cs_set_protected:Npn \math_bcsym_DeL:Nn #1#2{\@syntype\mathopen{\OrdSymbol{#2}}}%\@symDeL
91 \cs_set_protected:Npn \math_bcsym_DeR:Nn #1#2{\@syntype\mathclose{\OrdSymbol{#2}}}%\@symDeR
92 \cs_set_protected:Npn \math_bcsym_DeB:Nn #1#2{\@syntype\mathord{\OrdSymbol{#2}}}%\@symDeB
93 \cs_set_protected:Npn \math_bcsym_DeA:Nn #1#2{\@syntype\mathrel{\OrdSymbol{#2}}}%\@symDeA
94 \cs_set_protected:Npn \math_bcsym_Acc:Nn {\@sym}%\@symAcc FIX!
95 % These three?
96 \cs_set_protected:Npn \math_bcsym_Ope:Nn #1#2{\@syntype\mathopen{\OrdSymbol{#2}}}%\@symVar
97 \cs_set_protected:Npn \math_bcsym_Clo:Nn #1#2{\@syntype\mathclose{\OrdSymbol{#2}}}%\@symVar
98 \cs_set_protected:Npn \math_bcsym_Inn:Nn #1#2{\@syntype\mathinner{\OrdSymbol{#2}}}%\@symVar
99
100 \let\@syntype\@firstofone
101 \let\sym\global\global

```

The inline variants, using the basic operations. Currently we do not do anything to inline math.

```

102 \cs_new:Npn \math_isym_Ord:Nn {\math_bsym_Ord:Nn }%\m@Ord
103 \cs_new:Npn \math_isym_Var:Nn {\math_bsym_Var:Nn }%\m@Var
104 \cs_new:Npn \math_isym_Bin:Nn {\math_bsym_Bin:Nn }%\m@Bin
105 \cs_new:Npn \math_isym_Rel:Nn {\math_bsym_Rel:Nn }%\m@Bin
106 \cs_new:Npn \math_isym_Pun:Nn {\math_bsym_Pun:Nn }%\m@Pun
107 \cs_new:Npn \math_isym_COs:Nn {\math_bsym_COs:Nn }%\m@COs
108 \cs_new:Npn \math_isym_COi:Nn {\math_bsym_COi:Nn }%\m@COi
109 \cs_new:Npn \math_isym_DeL:Nn {\math_bsym_DeL:Nn }%\m@DeL
110 \cs_new:Npn \math_isym_DeR:Nn {\math_bsym_DeR:Nn }%\m@DeR
111 \cs_new:Npn \math_isym_DeB:Nn {\math_bsym_DeB:Nn }%\m@DeB
112 \cs_new:Npn \math_isym_DeA:Nn {\math_bsym_DeA:Nn }%\m@DeA
113 \cs_new:Npn \math_isym_Rad:Nn {\math_bsym_Rad:Nn }%\m@Rad
114 \cs_new:Npn \math_isym_Acc:Nn {\math_bsym_DeL:Nn }% name is wrong
115 % inline compound
116 \cs_set_protected:Npn \math_icsym_Ord:Nn {\math_bcsym_Ord:Nn}
117 \cs_set_protected:Npn \math_icsym_Var:Nn {\math_bcsym_Var:Nn}
118 \cs_set_protected:Npn \math_icsym_Bin:Nn {\math_bcsym_Bin:Nn}
119 \cs_set_protected:Npn \math_icsym_Rel:Nn {\math_bcsym_Rel:Nn}
120 \cs_set_protected:Npn \math_icsym_Pun:Nn {\math_bcsym_Pun:Nn}
121 \cs_set_protected:Npn \math_icsym_COi:Nn {\math_bcsym_COi:Nn}
122 \cs_set_protected:Npn \math_icsym_COs:Nn {\math_bcsym_COs:Nn}
123 \cs_set_protected:Npn \math_icsym_DeL:Nn {\math_bcsym_DeL:Nn}
124 \cs_set_protected:Npn \math_icsym_DeR:Nn {\math_bcsym_DeR:Nn}
125 \cs_set_protected:Npn \math_icsym_DeB:Nn {\math_bcsym_DeB:Nn}
126 \cs_set_protected:Npn \math_icsym_DeA:Nn {\math_bcsym_DeA:Nn}
127 \cs_set_protected:Npn \math_icsym_Acc:Nn {\math_bcsym_Acc:Nn}
128 \cs_set_protected:Npn \math_icsym_Ope:Nn {\math_bcsym_Ope:Nn}
129 \cs_set_protected:Npn \math_icsym_Clo:Nn {\math_bcsym_Clo:Nn}
130 \cs_set_protected:Npn \math_icsym_Inn:Nn {\math_bcsym_Inn:Nn}

```

The display variants, using the basic operations. Currently we do not do anything to inline math.

```

131 \cs_new:Npn \math_dsym_Ord:Nn {\math_bsym_Ord:Nn }

```

```

132 \cs_new:Npn \math_dsym_Var:Nn {\math_bsym_Var:Nn }
133 \cs_new:Npn \math_dsym_Bin:Nn {\math_bsym_Bin:Nn}
134 \cs_new:Npn \math_dsym_Rel:Nn {\math_bsym_Rel:Nn }
135 \cs_new:Npn \math_dsym_Pun:Nn {\math_bsym_Pun:Nn }
136 \cs_new:Npn \math_dsym_COs:Nn {\math_bsym_COs:Nn}
137 \cs_new:Npn \math_dsym_COi:Nn {\math_bsym_COi:Nn}
138 \cs_new:Npn \math_dsym_DeL:Nn {\math_bsym_DeL:Nn}
139 \cs_new:Npn \math_dsym_DeR:Nn{\math_bsym_DeR:Nn}
140 \cs_new:Npn \math_dsym_DeB:Nn{ \math_bsym_DeB:Nn }
141 \cs_new:Npn \math_dsym_DeA:Nn { \math_bsym_DeA:Nn }
142 \cs_new:Npn \math_dsym_Rad:Nn { \math_bsym_Rad:Nn}
143 \cs_new:Npn \math_dsym_Acc:Nn {\math_bsym_DeL:Nn}
144 % inline compound
145 \cs_set_protected:Npn \math_dcsym_Ord:Nn {\math_bcsym_Ord:Nn}
146 \cs_set_protected:Npn \math_dcsym_Var:Nn {\math_bcsym_Var:Nn}
147 \cs_set_protected:Npn \math_dcsym_Bin:Nn {\math_bcsym_Bin:Nn}
148 \cs_set_protected:Npn \math_dcsym_Rel:Nn {\math_bcsym_Rel:Nn}
149 \cs_set_protected:Npn \math_dcsym_Pun:Nn {\math_bcsym_Pun:Nn}
150 \cs_set_protected:Npn \math_dcsym_COi:Nn {\math_bcsym_COi:Nn}
151 \cs_set_protected:Npn \math_dcsym_COs:Nn {\math_bcsym_COs:Nn}
152 \cs_set_protected:Npn \math_dcsym_DeL:Nn {\math_bcsym_DeL:Nn}
153 \cs_set_protected:Npn \math_dcsym_DeR:Nn {\math_bcsym_DeR:Nn}
154 \cs_set_protected:Npn \math_dcsym_DeB:Nn {\math_bcsym_DeB:Nn}
155 \cs_set_protected:Npn \math_dcsym_DeA:Nn {\math_bcsym_DeA:Nn}
156 \cs_set_protected:Npn \math_dcsym_Acc:Nn {\math_bcsym_Acc:Nn}
157 \cs_set_protected:Npn \math_dcsym_Ope:Nn {\math_bcsym_Ope:Nn}
158 \cs_set_protected:Npn \math_dcsym_Clo:Nn {\math_bcsym_Clo:Nn}
159 \cs_set_protected:Npn \math_dcsym_Inn:Nn {\math_bcsym_Inn:Nn}

```

Almost ready now! Now just need two commands to initialize these settings.

```

160 \cs_set:Npn \math_setup_inline_symbols:{
161   \cs_set_eq:NN \math_sym_Ord:Nn \math_isym_Ord:Nn
162   \cs_set_eq:NN \math_sym_Var:Nn \math_isym_Var:Nn
163   \cs_set_eq:NN \math_sym_Bin:Nn \math_isym_Bin:Nn
164   \cs_set_eq:NN \math_sym_Rel:Nn \math_isym_Rel:Nn
165   \cs_set_eq:NN \math_sym_Pun:Nn \math_isym_Pun:Nn
166   \cs_set_eq:NN \math_sym_COs:Nn \math_isym_COs:Nn
167   \cs_set_eq:NN \math_sym_COi:Nn \math_isym_COi:Nn
168   \cs_set_eq:NN \math_sym_DeL:Nn \math_isym_DeL:Nn
169   \cs_set_eq:NN \math_sym_DeR:Nn \math_isym_DeR:Nn
170   \cs_set_eq:NN \math_sym_DeB:Nn \math_isym_DeL:Nn
171   \cs_set_eq:NN \math_sym_DeA:Nn \math_isym_DeA:Nn
172   \cs_set_eq:NN \math_sym_Rad:Nn \math_isym_Rad:Nn
173   \cs_set_eq:NN \math_sym_Acc:Nn \math_isym_DeL:Nn
174   \cs_set_eq:NN \math_csym_Ord:Nn \math_icsym_Ord:Nn
175   \cs_set_eq:NN \math_csym_Var:Nn \math_icsym_Var:Nn
176   \cs_set_eq:NN \math_csym_Bin:Nn \math_icsym_Bin:Nn
177   \cs_set_eq:NN \math_csym_Rel:Nn \math_icsym_Rel:Nn
178   \cs_set_eq:NN \math_csym_Pun:Nn \math_icsym_Pun:Nn
179   \cs_set_eq:NN \math_csym_COi:Nn \math_icsym_COi:Nn

```

```

180 \cs_set_eq:NN \math_csym_COs:Nn \math_icsym_COs:Nn
181 \cs_set_eq:NN \math_csym_DeL:Nn \math_icsym_DeL:Nn
182 \cs_set_eq:NN \math_csym_DeR:Nn \math_icsym_DeR:Nn
183 \cs_set_eq:NN \math_csym_DeB:Nn \math_icsym_DeB:Nn
184 \cs_set_eq:NN \math_csym_DeA:Nn \math_icsym_DeA:Nn
185 \cs_set_eq:NN \math_csym_Acc:Nn \math_icsym_Acc:Nn
186 \cs_set_eq:NN \math_csym_Ope:Nn \math_icsym_Ope:Nn
187 \cs_set_eq:NN \math_csym_Clo:Nn \math_icsym_Clo:Nn
188 \cs_set_eq:NN \math_csym_Inn:Nn \math_icsym_Inn:Nn
189 }
190
191 \cs_set:Npn \math_setup_display_symbols:{
192   \cs_set_eq:NN \math_sym_Ord:Nn \math_dsym_Ord:Nn
193   \cs_set_eq:NN \math_sym_Var:Nn \math_dsym_Var:Nn
194   \cs_set_eq:NN \math_sym_Bin:Nn \math_dsym_Bin:Nn
195   \cs_set_eq:NN \math_sym_Rel:Nn \math_dsym_Rel:Nn
196   \cs_set_eq:NN \math_sym_Pun:Nn \math_dsym_Pun:Nn
197   \cs_set_eq:NN \math_sym_COs:Nn \math_dsym_COs:Nn
198   \cs_set_eq:NN \math_sym_COi:Nn \math_dsym_COi:Nn
199   \cs_set_eq:NN \math_sym_DeL:Nn \math_dsym_DeL:Nn
200   \cs_set_eq:NN \math_sym_DeR:Nn \math_dsym_DeR:Nn
201   \cs_set_eq:NN \math_sym_DeB:Nn \math_dsym_DeL:Nn
202   \cs_set_eq:NN \math_sym_DeA:Nn \math_dsym_DeA:Nn
203   \cs_set_eq:NN \math_sym_Rad:Nn \math_dsym_Rad:Nn
204   \cs_set_eq:NN \math_sym_Acc:Nn \math_dsym_DeL:Nn
205   \cs_set_eq:NN \math_csym_Ord:Nn \math_dcsym_Ord:Nn
206   \cs_set_eq:NN \math_csym_Var:Nn \math_dcsym_Var:Nn
207   \cs_set_eq:NN \math_csym_Bin:Nn \math_dcsym_Bin:Nn
208   \cs_set_eq:NN \math_csym_Rel:Nn \math_dcsym_Rel:Nn
209   \cs_set_eq:NN \math_csym_Pun:Nn \math_dcsym_Pun:Nn
210   \cs_set_eq:NN \math_csym_COi:Nn \math_dcsym_COi:Nn
211   \cs_set_eq:NN \math_csym_COs:Nn \math_dcsym_COs:Nn
212   \cs_set_eq:NN \math_csym_DeL:Nn \math_dcsym_DeL:Nn
213   \cs_set_eq:NN \math_csym_DeR:Nn \math_dcsym_DeR:Nn
214   \cs_set_eq:NN \math_csym_DeB:Nn \math_dcsym_DeB:Nn
215   \cs_set_eq:NN \math_csym_DeA:Nn \math_dcsym_DeA:Nn
216   \cs_set_eq:NN \math_csym_Acc:Nn \math_dcsym_Acc:Nn
217   \cs_set_eq:NN \math_csym_Ope:Nn \math_dcsym_Ope:Nn
218   \cs_set_eq:NN \math_csym_Clo:Nn \math_dcsym_Clo:Nn
219   \cs_set_eq:NN \math_csym_Inn:Nn \math_dcsym_Inn:Nn
220 }

```

Phew, that was it.

Well, almost. We need to set them up for use properly. Should they be added to `\everymath`? Probably, for math within displays. However, this is a lot of extra processing which we could tackle in the display setup.

```
221 \math_setup_inline_symbols:
```

Need an active character for a second. Don't rely on `~` being active!

```
222 \edef\tmp{\catcode\z@=\the\catcode\z@}
```

```

223 \catcode\z@=\active
224 \def\DeclareFlexSymbol#1#2#3#4{%
225   \begingroup
226   \cs_set_protected:Npx\@tempb{
227     \exp_not:N\@sym\exp_not:N#1\exp_not:c{math_sym_#2:Nn}
228     \exp_not:c{mg@#3}{#4}
229   }
230   \ifcat\exp_not:N#1\relax
231     \sym@global\let#1\@tempb
232   \else
233     \sym@global\mathcode'#1="8000\relax
234     \lccode\z@='#1\relax
235     \lowercase{\sym@global\let^^@\@tempb}% zero char
236   \fi
237   \endgroup
238 }
239 \tmp % restore catcode
240 \cs_set:Npn \DeclareFlexDelimiter #1#2#3#4#5#6{
241   \DeclareFlexSymbol{#1}{#2}{#3}{#4}
242   \cs_gset:cpx{sd@use:c{mg@#3}{#4}{\exp_not:c{mg@#5}{#6}}}
243 }
244
\DeclareFlexCompoundSymbol{\cdots}{Inn}{\cdotp\cdotp\cdotp} \def\@symInn#1#2{\@symtyp
\@symtype \mathinner{\OrdSymbol{\cdotp\cdotp\cdotp}}}
245 \def\DeclareFlexCompoundSymbol#1#2#3{%
246   \exp_args:NNo \DeclareRobustCommand#1{\csname math_csym_#2:Nn\endcsname#1{#3}}%
247   \sym@global\let#1#1\relax
248 }
249 \DeclareRobustCommand\textchar{\text@char\textfont}
250 \DeclareRobustCommand\scriptchar{\text@char\scriptfont}%

```

Simplified the next bit because now the slot is read as one argument so no afteras-  
signment and what have you. Just drop the char directly.

```

251 \def\text@char@sym#1#2#3#4{% #3=fam, #4=slot
252   \begingroup
253     \cs_set_eq:NN \@sym \prg_do_nothing: % defense against infinite loops
254     \the\text@script@char#3%
255     \char"#4\endgroup
256 }
257 \edef\tmp{\catcode\z@=\the\catcode\z@}
258 \catcode\z@=\active
259 \def\text@char#1#2{\begingroup
260   \check@mathfonts
261   \cs_set_eq:NN \text@script@char#1
262   \cs_set_eq:NN \@sym\text@char@sym
263   \cs_set_eq:NN \@symtype\use_ii:nn
264   \cs_set_eq:NN\OrdSymbol\use:n
265   \cs_set_eq:NN\ifmmode\iftrue

```



```

266 \everymath{$\use_none:n}%$
267 \def\mkern{\mskip\z@}
268 \cs_set_eq:NN\mskip\mkern
269 \ifcat\relax\noexpand#2% true if #2 is a cs.
270   #2%
271 \else
272   \lccode\z@=\expandafter'\string#2\relax
273   \lowercase{^^@}%
274 \fi
275 \endgroup
276 }
277 \tmp % restore catcode
278 \providecommand\textprime{}
279 \DeclareRobustCommand\textprime{\leavevmode
280   \raise.8ex\hbox{\text@char\scriptfont\prime}%
281 }
282 \ifundefined{resetMathstrut@}{\{}{\%
283   \def\resetMathstrut@{%
284     \setbox\z@\hbox{\textchar\vert}%
285     \ht\Mathstrutbox@ \ht\z@ \dp\Mathstrutbox@ \dp\z@
286   }%
287 }

```

Arrow fills. changed to 7mu as in amsmath

```

288 \ifundefined{rightarrowfill@}{\{}{\%
289   \def\rightarrowfill@#1{\m@th\setboxz@h{${#1\relbar$}\ht\z@\z@
290     ${#1\copy\z@\mkern-7mu\cleaders
291       \hbox{${#1\mkern-2mu\box\z@\mkern-2mu$}\hfill
292       \mkern-6mu\OrdSymbol{\rightarrow}$}
293   \def\leftarrowfill@#1{\m@th\setboxz@h{${#1\relbar$}\ht\z@\z@
294     ${#1\OrdSymbol{\leftarrow}\mkern-6mu\cleaders
295       \hbox{${#1\mkern-2mu\copy\z@\mkern-2mu$}\hfill
296       \mkern-7mu\box\z@$}
297   \def\leftrightharrowfill@#1{\m@th\setboxz@h{${#1\relbar$}\ht\z@\z@
298     ${#1\OrdSymbol{\leftarrow}\mkern-6mu\cleaders
299       \hbox{${#1\mkern-2mu\box\z@\mkern-2mu$}\hfill
300       \mkern-6mu\OrdSymbol{\rightarrow}$}
301 }

```

hey, this looks like a simple case switch...

```

302 \def\binrel@sym#1#2#3#4{%
303   \xdef\binrel@#1{%
304     \ifx\math_sym_Ord:Nn #2 \math_csym_Ord:Nn
305     \else\ifx\math_sym_Var:Nn#2 \math_csym_Var:Nn
306     \else\ifx\math_sym_COs:Nn#2 \math_csym_COs:Nn
307     \else\ifx\math_sym_COi:Nn#2 \math_csym_COi:Nn
308     \else\ifx\math_sym_Bin:Nn#2 \math_csym_Bin:Nn
309     \else\ifx\math_sym_Rel:Nn#2 \math_csym_Rel:Nn
310     \else\ifx\math_sym_Pun:Nn#2 \math_csym_Pun:Nn
311     \else\exp_not:N\@symErr \fi\fi\fi\fi\fi\fi\fi
312     ?{\exp_not:N\OrdSymbol{##1}}}%

```

```

313 }
314
315 \def\binrel@a{%
316   \def\math_sym_Ord:Nn##1##2{\gdef\binrel@@###1{\math_sym_Ord:Nn##1{\OrdSymbol{####1}}}}%
317   \def\math_sym_Var:Nn##1##2{\gdef\binrel@@###1{\math_sym_Var:Nn##1{\OrdSymbol{####1}}}}%
318   \def\math_sym_COs:Nn##1##2{\gdef\binrel@@###1{\math_sym_COs:Nn##1{\OrdSymbol{####1}}}}%
319   \def\math_sym_COi:Nn##1##2{\gdef\binrel@@###1{\math_sym_COi:Nn##1{\OrdSymbol{####1}}}}%
320   \def\math_sym_Bin:Nn##1##2{\gdef\binrel@@###1{\math_sym_Bin:Nn##1{\OrdSymbol{####1}}}}%
321   \def\math_sym_Rel:Nn##1##2{\gdef\binrel@@###1{\math_sym_Rel:Nn##1{\OrdSymbol{####1}}}}%
322   \def\math_sym_Pun:Nn##1##2{\gdef\binrel@@###1{\math_sym_Pun:Nn##1{\OrdSymbol{####1}}}}%
323 }
324 \def\binrel@#1{%
325   \setbox\z@\hbox{${}%
326     \let\mathchoice\@gobblethree
327     \let\@sym\binrel@sym \binrel@a
328     #1$}%
329 }
330 \def\@symextension{sym}
331 \newcommand\usesymbols[1]{%
332   \clist_map_variable:Nn{#1}\@tempb{%
333     \exp_args:No\@onefilewithoptions{\@tempb} [] []\@symextension
334   }%
335 }
336 % Need to introduce \ProvidesExplFile somehow
337 \newcommand\ProvidesSymbols[1]{\ProvidesFile{#1.sym}}
338 \DeclareRobustCommand{\not}[1]{\math_csym_Rel:Nn\not{\OrdSymbol{\notRel#1}}}
339 \DeclareRobustCommand{\OrdSymbol}[1]{%
340   \begingroup\mathchars@reset#1\endgroup
341 }
342 \def\mathchars@reset{\let\@sym\@sym@ord \let\@symtype\@symtype@ord
343   \let\OrdSymbol\relax}
344 \def\@symtype@ord#1#2{% a strange sort of \@gobble
345 \def\@sym@ord#1#2{\exp_after:wN\@sym@ord@a\string#2\@nil}%
346 \begingroup
347 \lccode'\.='\_ \lowercase{\endgroup
348 \def\@sym@ord@a#1.#2.}{#3#4\@nil#5#6}%
349 \if D#3
350   %\math_ord_delim_aux:Nn #5{#6}
351   \math_sd_del_aux:Nnn 0 #5{#6}% check if this works!
352 \else
353   \math_char:Nnn 0 #5{#6}
354 \fi

```

Read delimited argument here. We want to find first character of DeA, Bin, etc. and the control sequence checked against is `\m@DeL`, `\m@Pun`, etc. The `lccode` trick makes the `.` into an `@` with catcode 12. This is what results when the code is called with `\string`. Beware of this when we change internal names for math groups! If a Delimiter is found, insert it with class 0 but use the smallest version available. Otherwise just insert math char of class 0. The code here is not pretty and it indicates it should be tackled differently!

```

355 }
356 \cs_set:Nn \math_ord_delim_aux:Nn {
357   \math_sd_aux:nn { \math_char:NNn 0 } {#1{#2}}
358 }

```

Before declaring any math characters active, we have to take care of a small problem with `amsmath v2.x`, if it is loaded before `flexisym`. `\std@minus` and `\std@equal` are defined as

```

\mathchardef\std@minus\mathcode'\- \relax
\mathchardef\std@equal\mathcode'\= \relax

```

in `amsmath.sty` and again `\AtBeginDocument`. The latter is because

In case some alternative math fonts are loaded later. [`amsmath.dtx`]

The problem arises because `flexisym` sets the mathcode of all symbols to 32768 which is illegal for a `\mathchardef`.

We have to remove the assignments from the `\AtBeginDocument` hook as they will cause an error there.

```

359 \@ifpackageloaded{amsmath}{%
360   \begingroup

```

Split the contents of `\@begindocumenthook` by reading what we search for as a delimited argument and ensure these two assignments do not take place. It is questionable if anything reasonable can be done to them. In the case of a package such as `mathpazo` which defines

```

\DeclareMathSymbol{=}{\mathrel}{upright}{"3D}

```

the `\Relbar` will look wrong if we don't use the correct symbol. The way to solve this is define additional `.sym` files which contain the definition of `\relbar` and `\Relbar` needed. We need those additional files anyway for things like `\joinord`.

```

361   \long\def\next#1\mathchardef\std@minus\mathcode'\- \relax
362           \mathchardef\std@equal\mathcode'\= \relax#2\flexi@stop{%
363     \toks@{#1#2}%
364     \xdef\@begindocumenthook{\the\toks@}%
365   }%
366   \expandafter\next\@begindocumenthook\flexi@stop
367   \endgroup
368 }{}

```

There is problem when using `\DeclareMathOperator` as the operators defined call a command `\newmcodes@` which relies on the mathcode of `-` being less than 32768. We delay the definition `\AtBeginDocument` in case `amssymb` hasn't been loaded yet.

```

369 \AtBeginDocument{%
370   \def\newmcodes@{%
371     \mathcode '\ '39\space

```

```

372 \mathcode '\*42\space
373 \mathcode '\."613A\space
374 \ifnum\mathcode'\-=45\space
375 \else
    The extra check. Don't do anything if - is math active.
376 \ifnum\mathcode'\-=32768\space
377 \else
378 \mathchardef \std@minus \mathcode '\-\relax
379 \fi
380 \fi
381 \mathcode '\-45\space
382 \mathcode '\-/47\space
383 \mathcode '\:"603A\space\relax
384 }%
385 }

```

And we then continue with the options.

```

386 \DeclareOption{mathstyleoff}{%
387 \PassOptionsToPackage{noactivechars}{mathstyle}}
388 \DeclareOption{cmbase}{\usesymbols{cmbase}}
389 \DeclareOption{mathpazo}{\usesymbols{mathpazo}}
390 \DeclareOption{mathptmx}{\usesymbols{mathptmx}}
391 \ExecuteOptions{cmbase}
392 \ProcessOptions\relax
393 \renewcommand{\lnot}{\neg}
394 \renewcommand{\land}{\wedge}
395 \renewcommand{\lor}{\vee}
396 \renewcommand{\le}{\leq}
397 \renewcommand{\ge}{\geq}
398 \renewcommand{\ne}{\neq}
399 \renewcommand{\owns}{\ni}
400 \renewcommand{\gets}{\leftarrow}
401 \renewcommand{\to}{\rightarrow}
402 \renewcommand{\|}{\Vert}
403 \RequirePackage{mathstyle}
404 \endinput

```

## 2 cmbase, mathpazo, mathptmx

For each math font package we define a corresponding symbol file with extension `sym`. The Computer Modern base is called `cmbase` and `mathpazo` and `mathptmx` corresponds to the packages. The definitions are almost identical as they mostly concern the positions in the math font encodings. Look for differences in `\joinord`, `\relbar` and `\Relbar`. If you inspect the source code, you'll see that the support for `mathptmx` didn't require any work but I thought it better to create a `sym` file to maintain a uniform interface.

**Open question on ! and ?:** maybe they should have type 'Pun' instead of 'DeR'. Need to search for uses in math in

AMS archives. Or, maybe add a special ‘Clo’ type for them:  
non-extensible closing delimiter.

Default mathgroup setup.

```

405 <*cmbase | mathpazo | mathptmx>
406 <cmbase> \ProvidesSymbols{cmbase}[2007/12/19 v0.92]
407 <mathpazo> \ProvidesSymbols{mathpazo}[2010/07/11 v0.3]
408 <mathptmx> \ProvidesSymbols{mathptmx}[2010/07/11 v0.3]
409 \ExplSyntaxOn
410 \cs_gset:cpx{mg@OT1}{\hexnumber@\symoperators}
411 \cs_gset:cpx{mg@OML}{\hexnumber@\symletters}
412 \cs_gset:cpx{mg@OMS}{\hexnumber@\symsymbols}
413 \cs_gset:cpx{mg@OMX}{\hexnumber@\symlargesymbols}
414 \cs_gset:Npx\mg@bin{\mg@OMS}
415 \cs_gset:Npx\mg@del{\mg@OMX}
416 \cs_gset:Npx\mg@digit{\exp_not:c{mg@OT1}}
417 \cs_gset:Npn\mg@latin{\mg@OML}
418 \cs_gset_eq:NN\mg@Latin\mg@latin
419 \cs_gset_eq:NN\mg@greek\mg@latin
420 <cmbase | mathptmx> \cs_gset_eq:NN\mg@greek\mg@digit

```

Mathpazo takes the upper case greeks from the letter font if slantedGreek is in effect, but from *upright* if not. Mathptmx also takes the slanted greek from the letter font.

```

421 <mathpazo> \@ifpackagewith{mathpazo}{slantedGreek}{%
422 <mathpazo> \cs_gset_eq:NN\mg@greek\mg@latin
423 <mathpazo> }{%
424 <mathpazo> \cs_gset:cpx{mg@greek}{\hexnumber@\symupright}
425 <mathpazo> }
426 <mathptmx> \@ifpackagewith{mathptmx}{slantedGreek}{%
427 <mathptmx> \cs_gset_eq:NN\mg@greek\mg@latin
428 <mathptmx> }{}
429 \cs_gset_eq:NN\mg@rel\mg@bin
430 \cs_gset_eq:NN\mg@ord\mg@bin
431 \cs_gset_eq:NN\mg@cop\mg@del

```

Symbols from the 128-character cmr encoding. Paren and square bracket delimiters from this encoding are covered by the definitions in the cmex section, however.

```

432 \DeclareFlexSymbol{!} {Pun}{OT1}{21}
433 \DeclareFlexSymbol{+} {Bin}{OT1}{2B}
434 \DeclareFlexSymbol{:} {Rel}{OT1}{3A}
435 \DeclareFlexSymbol{\colon} {Pun}{OT1}{3A}
436 \DeclareFlexSymbol{;} {Pun}{OT1}{3B}
437 \DeclareFlexSymbol{=} {Rel}{OT1}{3D}
438 \DeclareFlexSymbol{?} {Pun}{OT1}{3F}

```

$\mathcal{M}\mathcal{S}\mathcal{T}_{\mathrm{E}}\mathrm{X}$ , and therefore the amsmath package, make the uppercase Greek letters class 0 (nonvariable) instead of 7 (variable), to eliminate the glaring inconsistency with lowercase Greek. (In plain  $\mathrm{T}_{\mathrm{E}}\mathrm{X}$ ,  $\{\mathrm{bf}\Delta\}$  works, while  $\{\mathrm{bf}\delta\}$

doesn't.) Let us try to make them both variable (fonts permitting) instead of nonvariable.

```

439 \DeclareFlexSymbol{\Gamma} {Var}{Greek}{00}
440 \DeclareFlexSymbol{\Delta} {Var}{Greek}{01}
441 \DeclareFlexSymbol{\Theta} {Var}{Greek}{02}
442 \DeclareFlexSymbol{\Lambda} {Var}{Greek}{03}
443 \DeclareFlexSymbol{\Xi} {Var}{Greek}{04}
444 \DeclareFlexSymbol{\Pi} {Var}{Greek}{05}
445 \DeclareFlexSymbol{\Sigma} {Var}{Greek}{06}
446 \DeclareFlexSymbol{\Upsilon} {Var}{Greek}{07}
447 \DeclareFlexSymbol{\Phi} {Var}{Greek}{08}
448 \DeclareFlexSymbol{\Psi} {Var}{Greek}{09}
449 \DeclareFlexSymbol{\Omega} {Var}{Greek}{0A}

```

Decimal digits.

```

450 \DeclareFlexSymbol{0}{Var}{digit}{30}
451 \DeclareFlexSymbol{1}{Var}{digit}{31}
452 \DeclareFlexSymbol{2}{Var}{digit}{32}
453 \DeclareFlexSymbol{3}{Var}{digit}{33}
454 \DeclareFlexSymbol{4}{Var}{digit}{34}
455 \DeclareFlexSymbol{5}{Var}{digit}{35}
456 \DeclareFlexSymbol{6}{Var}{digit}{36}
457 \DeclareFlexSymbol{7}{Var}{digit}{37}
458 \DeclareFlexSymbol{8}{Var}{digit}{38}
459 \DeclareFlexSymbol{9}{Var}{digit}{39}

```

Symbols from the 128-character cmmt encoding.

```

460 \DeclareFlexSymbol{,}{Pun}{OML}{3B}
461 \DeclareFlexSymbol{.}{Ord}{OML}{3A}
462 \DeclareFlexSymbol{/}{Ord}{OML}{3D}
463 \DeclareFlexSymbol{<}{Rel}{OML}{3C}
464 \DeclareFlexSymbol{>}{Rel}{OML}{3E}

```

To do: make the Var property of lc Greek work properly.

```

465 \DeclareFlexSymbol{\alpha}{Var}{greek}{0B}
466 \DeclareFlexSymbol{\beta}{Var}{greek}{0C}
467 \DeclareFlexSymbol{\gamma}{Var}{greek}{0D}
468 \DeclareFlexSymbol{\delta}{Var}{greek}{0E}
469 \DeclareFlexSymbol{\epsilon}{Var}{greek}{0F}
470 \DeclareFlexSymbol{\zeta}{Var}{greek}{10}
471 \DeclareFlexSymbol{\eta}{Var}{greek}{11}
472 \DeclareFlexSymbol{\theta}{Var}{greek}{12}
473 \DeclareFlexSymbol{\iota}{Var}{greek}{13}
474 \DeclareFlexSymbol{\kappa}{Var}{greek}{14}
475 \DeclareFlexSymbol{\lambda}{Var}{greek}{15}
476 \DeclareFlexSymbol{\mu}{Var}{greek}{16}
477 \DeclareFlexSymbol{\nu}{Var}{greek}{17}
478 \DeclareFlexSymbol{\xi}{Var}{greek}{18}
479 \DeclareFlexSymbol{\pi}{Var}{greek}{19}
480 \DeclareFlexSymbol{\rho}{Var}{greek}{1A}
481 \DeclareFlexSymbol{\sigma}{Var}{greek}{1B}

```

```

482 \DeclareFlexSymbol{\tau}{Var}{greek}{1C}
483 \DeclareFlexSymbol{\upsilon}{Var}{greek}{1D}
484 \DeclareFlexSymbol{\phi}{Var}{greek}{1E}
485 \DeclareFlexSymbol{\chi}{Var}{greek}{1F}
486 \DeclareFlexSymbol{\psi}{Var}{greek}{20}
487 \DeclareFlexSymbol{\omega}{Var}{greek}{21}
488 \DeclareFlexSymbol{\varepsilon}{Var}{greek}{22}
489 \DeclareFlexSymbol{\vartheta}{Var}{greek}{23}
490 \DeclareFlexSymbol{\varpi}{Var}{greek}{24}
491 \DeclareFlexSymbol{\varrho}{Var}{greek}{25}
492 \DeclareFlexSymbol{\varsigma}{Var}{greek}{26}
493 \DeclareFlexSymbol{\varphi}{Var}{greek}{27}

```

Note that in plain  $\TeX$  `\imath` and `\jmath` are not variable-font. But if a `j` changes font to, let's say, sans serif or calligraphic, a dotless `j` in the same context should change font in the same way.

```

494 \DeclareFlexSymbol{\imath}{Var}{OML}{7B}
495 \DeclareFlexSymbol{\jmath}{Var}{OML}{7C}
496 \DeclareFlexSymbol{\ell}{Ord}{OML}{60}
497 \DeclareFlexSymbol{\wp}{Ord}{OML}{7D}
498 \DeclareFlexSymbol{\partial}{Ord}{OML}{40}
499 \DeclareFlexSymbol{\flat}{Ord}{OML}{5B}
500 \DeclareFlexSymbol{\natural}{Ord}{OML}{5C}
501 \DeclareFlexSymbol{\sharp}{Ord}{OML}{5D}
502 \DeclareFlexSymbol{\triangleleft}{Bin}{OML}{2F}
503 \DeclareFlexSymbol{\triangleright}{Bin}{OML}{2E}
504 \DeclareFlexSymbol{\star}{Bin}{OML}{3F}
505 \DeclareFlexSymbol{\smile}{Rel}{OML}{5E}
506 \DeclareFlexSymbol{\frown}{Rel}{OML}{5F}
507 \DeclareFlexSymbol{\leftharpoonup}{Rel}{OML}{28}
508 \DeclareFlexSymbol{\leftharpoondown}{Rel}{OML}{29}
509 \DeclareFlexSymbol{\rightharpoonup}{Rel}{OML}{2A}
510 \DeclareFlexSymbol{\rightharpoondown}{Rel}{OML}{2B}
511 \DeclareFlexSymbol{a}{Var}{latin}{61}
512 \DeclareFlexSymbol{b}{Var}{latin}{62}
513 \DeclareFlexSymbol{c}{Var}{latin}{63}
514 \DeclareFlexSymbol{d}{Var}{latin}{64}
515 \DeclareFlexSymbol{e}{Var}{latin}{65}
516 \DeclareFlexSymbol{f}{Var}{latin}{66}
517 \DeclareFlexSymbol{g}{Var}{latin}{67}
518 \DeclareFlexSymbol{h}{Var}{latin}{68}
519 \DeclareFlexSymbol{i}{Var}{latin}{69}
520 \DeclareFlexSymbol{j}{Var}{latin}{6A}
521 \DeclareFlexSymbol{k}{Var}{latin}{6B}
522 \DeclareFlexSymbol{l}{Var}{latin}{6C}
523 \DeclareFlexSymbol{m}{Var}{latin}{6D}
524 \DeclareFlexSymbol{n}{Var}{latin}{6E}
525 \DeclareFlexSymbol{o}{Var}{latin}{6F}
526 \DeclareFlexSymbol{p}{Var}{latin}{70}
527 \DeclareFlexSymbol{q}{Var}{latin}{71}

```

```

528 \DeclareFlexSymbol{r}{Var}{latin}{72}
529 \DeclareFlexSymbol{s}{Var}{latin}{73}
530 \DeclareFlexSymbol{t}{Var}{latin}{74}
531 \DeclareFlexSymbol{u}{Var}{latin}{75}
532 \DeclareFlexSymbol{v}{Var}{latin}{76}
533 \DeclareFlexSymbol{w}{Var}{latin}{77}
534 \DeclareFlexSymbol{x}{Var}{latin}{78}
535 \DeclareFlexSymbol{y}{Var}{latin}{79}
536 \DeclareFlexSymbol{z}{Var}{latin}{7A}
537 \DeclareFlexSymbol{A}{Var}{Latin}{41}
538 \DeclareFlexSymbol{B}{Var}{Latin}{42}
539 \DeclareFlexSymbol{C}{Var}{Latin}{43}
540 \DeclareFlexSymbol{D}{Var}{Latin}{44}
541 \DeclareFlexSymbol{E}{Var}{Latin}{45}
542 \DeclareFlexSymbol{F}{Var}{Latin}{46}
543 \DeclareFlexSymbol{G}{Var}{Latin}{47}
544 \DeclareFlexSymbol{H}{Var}{Latin}{48}
545 \DeclareFlexSymbol{I}{Var}{Latin}{49}
546 \DeclareFlexSymbol{J}{Var}{Latin}{4A}
547 \DeclareFlexSymbol{K}{Var}{Latin}{4B}
548 \DeclareFlexSymbol{L}{Var}{Latin}{4C}
549 \DeclareFlexSymbol{M}{Var}{Latin}{4D}
550 \DeclareFlexSymbol{N}{Var}{Latin}{4E}
551 \DeclareFlexSymbol{O}{Var}{Latin}{4F}
552 \DeclareFlexSymbol{P}{Var}{Latin}{50}
553 \DeclareFlexSymbol{Q}{Var}{Latin}{51}
554 \DeclareFlexSymbol{R}{Var}{Latin}{52}
555 \DeclareFlexSymbol{S}{Var}{Latin}{53}
556 \DeclareFlexSymbol{T}{Var}{Latin}{54}
557 \DeclareFlexSymbol{U}{Var}{Latin}{55}
558 \DeclareFlexSymbol{V}{Var}{Latin}{56}
559 \DeclareFlexSymbol{W}{Var}{Latin}{57}
560 \DeclareFlexSymbol{X}{Var}{Latin}{58}
561 \DeclareFlexSymbol{Y}{Var}{Latin}{59}
562 \DeclareFlexSymbol{Z}{Var}{Latin}{5A}

```

The `\ldotPun` glyph is used in constructing the `\ldots` symbol. It is just a period with a different math symbol class. `\lhookRel` and `\rhookRel` are used in a similar way for building hooked arrow symbols.

```

563 \DeclareFlexSymbol{\ldotPun}{Pun}{OML}{3A}
564 \def\ldotp{\ldotPun}
565 \DeclareFlexSymbol{\lhookRel}{Rel}{OML}{2C}
566 \DeclareFlexSymbol{\rhookRel}{Rel}{OML}{2D}

```

Symbols from the 128-character `cmsy` encoding.

```

567 \DeclareFlexSymbol{*}{Bin}{bin}{03} % \ast
568 \DeclareFlexSymbol{-}{Bin}{bin}{00}
569 \DeclareFlexSymbol{|}{Ord}{OMS}{6A}
570 \DeclareFlexSymbol{\aleph}{Ord}{ord}{40}
571 \DeclareFlexSymbol{\Re}{Ord}{ord}{3C}

```



```

572 \DeclareFlexSymbol{\Im}{Ord}{ord}{3D}
573 \DeclareFlexSymbol{\infty}{Ord}{ord}{31}
574 \DeclareFlexSymbol{\prime}{Ord}{ord}{30}
575 \DeclareFlexSymbol{\emptyset}{Ord}{ord}{3B}
576 \DeclareFlexSymbol{\nabla}{Ord}{ord}{72}
577 \DeclareFlexSymbol{\top}{Ord}{ord}{3E}
578 \DeclareFlexSymbol{\bot}{Ord}{ord}{3F}
579 \DeclareFlexSymbol{\triangle}{Ord}{ord}{34}
580 \DeclareFlexSymbol{\forall}{Ord}{ord}{38}
581 \DeclareFlexSymbol{\exists}{Ord}{ord}{39}
582 \DeclareFlexSymbol{\neg}{Ord}{ord}{3A}
583 \DeclareFlexSymbol{\clubsuit}{Ord}{ord}{7C}
584 \DeclareFlexSymbol{\diamondsuit}{Ord}{ord}{7D}
585 \DeclareFlexSymbol{\heartsuit}{Ord}{ord}{7E}
586 \DeclareFlexSymbol{\spadesuit}{Ord}{ord}{7F}
587 \DeclareFlexSymbol{\smallint}{COs}{OMS}{73}

```

Binary operators.

```

588 \DeclareFlexSymbol{\bigtriangleup}{Bin}{bin}{34}
589 \DeclareFlexSymbol{\bigtriangledown}{Bin}{bin}{35}
590 \DeclareFlexSymbol{\wedge}{Bin}{bin}{5E}
591 \DeclareFlexSymbol{\vee}{Bin}{bin}{5F}
592 \DeclareFlexSymbol{\cap}{Bin}{bin}{5C}
593 \DeclareFlexSymbol{\cup}{Bin}{bin}{5B}
594 \DeclareFlexSymbol{\dagger}{Bin}{bin}{7A}
595 \DeclareFlexSymbol{\dagger}{Bin}{bin}{79}
596 \DeclareFlexSymbol{\sqcap}{Bin}{bin}{75}
597 \DeclareFlexSymbol{\sqcup}{Bin}{bin}{74}
598 \DeclareFlexSymbol{\uplus}{Bin}{bin}{5D}
599 \DeclareFlexSymbol{\amalg}{Bin}{bin}{71}
600 \DeclareFlexSymbol{\diamond}{Bin}{bin}{05}
601 \DeclareFlexSymbol{\bullet}{Bin}{bin}{0F}
602 \DeclareFlexSymbol{\wr}{Bin}{bin}{6F}
603 \DeclareFlexSymbol{\div}{Bin}{bin}{04}
604 \DeclareFlexSymbol{\odot}{Bin}{bin}{0C}
605 \DeclareFlexSymbol{\oslash}{Bin}{bin}{0B}
606 \DeclareFlexSymbol{\otimes}{Bin}{bin}{0A}
607 \DeclareFlexSymbol{\ominus}{Bin}{bin}{09}
608 \DeclareFlexSymbol{\oplus}{Bin}{bin}{08}
609 \DeclareFlexSymbol{\mp}{Bin}{bin}{07}
610 \DeclareFlexSymbol{\pm}{Bin}{bin}{06}
611 \DeclareFlexSymbol{\circ}{Bin}{bin}{0E}
612 \DeclareFlexSymbol{\bigcirc}{Bin}{bin}{0D}
613 \DeclareFlexSymbol{\setminus}{Bin}{bin}{6E}
614 \DeclareFlexSymbol{\cdot}{Bin}{bin}{01}
615 \DeclareFlexSymbol{\ast}{Bin}{bin}{03}
616 \DeclareFlexSymbol{\times}{Bin}{bin}{02}

```

Relation symbols.

```

617 \DeclareFlexSymbol{\propto}{Rel}{rel}{2F}
618 \DeclareFlexSymbol{\sqsubseteq}{Rel}{rel}{76}

```

```

619 \DeclareFlexSymbol{\sqsupseteq}{Rel}{rel}{77}
620 \DeclareFlexSymbol{\parallel}{Rel}{rel}{6B}
621 \DeclareFlexSymbol{\mid}{Rel}{rel}{6A}
622 \DeclareFlexSymbol{\dashv}{Rel}{rel}{61}
623 \DeclareFlexSymbol{\vdash}{Rel}{rel}{60}
624 \DeclareFlexSymbol{\nearrow}{Rel}{rel}{25}
625 \DeclareFlexSymbol{\searrow}{Rel}{rel}{26}
626 \DeclareFlexSymbol{\nrightarrow}{Rel}{rel}{2D}
627 \DeclareFlexSymbol{\swarrow}{Rel}{rel}{2E}
628 \DeclareFlexSymbol{\Leftrightarrow}{Rel}{rel}{2C}
629 \DeclareFlexSymbol{\Leftarrow}{Rel}{rel}{28}
630 \DeclareFlexSymbol{\Rightarrow}{Rel}{rel}{29}
631 \DeclareFlexSymbol{\leq}{Rel}{rel}{14}
632 \DeclareFlexSymbol{\geq}{Rel}{rel}{15}
633 \DeclareFlexSymbol{\succ}{Rel}{rel}{1F}
634 \DeclareFlexSymbol{\prec}{Rel}{rel}{1E}
635 \DeclareFlexSymbol{\approx}{Rel}{rel}{19}
636 \DeclareFlexSymbol{\succeq}{Rel}{rel}{17}
637 \DeclareFlexSymbol{\preceq}{Rel}{rel}{16}
638 \DeclareFlexSymbol{\supset}{Rel}{rel}{1B}
639 \DeclareFlexSymbol{\subset}{Rel}{rel}{1A}
640 \DeclareFlexSymbol{\supseteq}{Rel}{rel}{13}
641 \DeclareFlexSymbol{\subseteq}{Rel}{rel}{12}
642 \DeclareFlexSymbol{\in}{Rel}{rel}{32}
643 \DeclareFlexSymbol{\ni}{Rel}{rel}{33}
644 \DeclareFlexSymbol{\gg}{Rel}{rel}{1D}
645 \DeclareFlexSymbol{\ll}{Rel}{rel}{1C}
646 \DeclareFlexSymbol{\leftrightharpoonup}{Rel}{rel}{24}
647 \DeclareFlexSymbol{\leftarrow}{Rel}{rel}{20}
648 \DeclareFlexSymbol{\rightarrow}{Rel}{rel}{21}
649 \DeclareFlexSymbol{\sim}{Rel}{rel}{18}
650 \DeclareFlexSymbol{\simeq}{Rel}{rel}{27}
651 \DeclareFlexSymbol{\perp}{Rel}{rel}{3F}
652 \DeclareFlexSymbol{\equiv}{Rel}{rel}{11}
653 \DeclareFlexSymbol{\asymp}{Rel}{rel}{10}

```

The `\notRel` glyph is a special zero-width glyph intended only for use in constructing negated symbols. `\mapstoRel` and `\cdotPun` have similar but more restricted applications.

```

654 \DeclareFlexSymbol{\notRel}{Rel}{rel}{36}
655 \DeclareFlexSymbol{\mapstoOrd}{Ord}{OMS}{37}
656 \DeclareFlexSymbol{\cdotOrd}{Ord}{OMS}{01}
657 \cs_set:Npn\cdotp{\mathpunct{\cdotOrd}}

```

Symbols from the 128-character `cmex` encoding. `COs` stands for ‘cumulative operator (sum-like)’. `COi` stands for ‘cumulative operator (integral-like)’. These typically differ only in the default placement of limits. `cop` stands for ‘cumulative operator math group’.

```

658 \DeclareFlexSymbol{\coprod}{COs}{cop}{60}
659 \DeclareFlexSymbol{\bigvee}{COs}{cop}{57}

```

```

660 \DeclareFlexSymbol{\bigwedge}{C0s}{cop}{56}
661 \DeclareFlexSymbol{\biguplus}{C0s}{cop}{55}
662 \DeclareFlexSymbol{\bigcap}{C0s}{cop}{54}
663 \DeclareFlexSymbol{\bigcup}{C0s}{cop}{53}
664 \DeclareFlexSymbol{\int}{C0i}{cop}{52}
665 \DeclareFlexSymbol{\prod}{C0s}{cop}{51}
666 \DeclareFlexSymbol{\sum}{C0s}{cop}{50}
667 \DeclareFlexSymbol{\bigotimes}{C0s}{cop}{4E}
668 \DeclareFlexSymbol{\bigoplus}{C0s}{cop}{4C}
669 \DeclareFlexSymbol{\bigodot}{C0s}{cop}{4A}
670 \DeclareFlexSymbol{\oint}{C0i}{cop}{48}
671 \DeclareFlexSymbol{\bigsqcup}{C0s}{cop}{46}

```

Delimiter symbols. DeL stands for ‘delimiter (left)’. DeR stands for ‘delimiter (right)’. DeB stands for ‘delimiter (bidirectional)’. The principal encoding point for an extensible delimiter is the first link in the list of linked sizes as specified in the font metric information. For a math encoding such as OT1/OML/OMS/OMX where not all sizes of a given delimiter reside in a given font, the extra encoding point for the smallest delimiter must be supplied by defining

```
\sd@GXX
```

where G is the mathgroup and XX is the hexadecimal glyph position. `\DeclareFlexDelimiter` does that for us.

```

672 \DeclareFlexDelimiter{\rangle}{DeR}{del}{0B}{OMS}{69}
673 \DeclareFlexDelimiter{\langle}{DeL}{del}{0A}{OMS}{68}
674 \DeclareFlexDelimiter{\rbrace}{DeR}{del}{09}{OMS}{67}
675 \DeclareFlexDelimiter{\lbrace}{DeL}{del}{08}{OMS}{66}
676 \DeclareFlexDelimiter{\rceil}{DeR}{del}{07}{OMS}{65}
677 \DeclareFlexDelimiter{\lceil}{DeL}{del}{06}{OMS}{64}
678 \DeclareFlexDelimiter{\rfloor}{DeR}{del}{05}{OMS}{63}
679 \DeclareFlexDelimiter{\lfloor}{DeL}{del}{04}{OMS}{62}
680 \DeclareFlexDelimiter{()}{DeL}{del}{00}{OT1}{28}
681 \DeclareFlexDelimiter{}{DeR}{del}{01}{OT1}{29}
682 \DeclareFlexDelimiter{[]}{DeL}{del}{02}{OT1}{5B}
683 \DeclareFlexDelimiter{}{DeR}{del}{03}{OT1}{5D}
684 \DeclareFlexDelimiter{\lVert}{DeL}{del}{0D}{OMS}{6B}
685 \DeclareFlexDelimiter{\rVert}{DeR}{del}{0D}{OMS}{6B}
686 \DeclareFlexDelimiter{\lvert}{DeL}{del}{0C}{OMS}{6A}
687 \DeclareFlexDelimiter{\rvert}{DeR}{del}{0C}{OMS}{6A}
688 \DeclareFlexDelimiter{\Vert}{DeB}{del}{0D}{OMS}{6B}
689 \DeclareFlexDelimiter{\vert}{DeB}{del}{0C}{OMS}{6A}

```

Maybe make the vert bars mathord instead of delimiter, to discourage poor usage.

```

690 \DeclareFlexDelimiter{|}{DeB}{del}{0C}{OMS}{6A}
691 \DeclareFlexDelimiter{/}{DeB}{del}{0E}{OML}{3D}

```

These wacky delimiters need to be supported I guess for compabitility reasons. The DeA delimiter type is a special case used only for these arrows.

```
692 \DeclareFlexDelimiter{\lmoustache}{DeL}{del}{40}{del}{7A}
```

```

693 \DeclareFlexDelimiter{\rmoustache}{DeR}{del}{41}{del}{7B}
694 \DeclareFlexDelimiter{\lgroup}{DeL}{del}{3A}{del}{3A}
695 \DeclareFlexDelimiter{\rgroup}{DeR}{del}{3B}{del}{3B}
696 \DeclareFlexDelimiter{\bracevert}{DeB}{del}{3E}{del}{3E}
697 \DeclareFlexDelimiter{\arrowvert}{DeB}{del}{3C}{OMS}{6A}
698 \DeclareFlexDelimiter{\Arrowvert}{DeB}{del}{3D}{OMS}{6B}
699 \DeclareFlexDelimiter{\uparrow}{DeA}{del}{78}{OMS}{22}
700 \DeclareFlexDelimiter{\downarrow}{DeA}{del}{79}{OMS}{23}
701 \DeclareFlexDelimiter{\updownarrow}{DeA}{del}{3F}{OMS}{6C}
702 \DeclareFlexDelimiter{\Uparrow}{DeA}{del}{7E}{OMS}{2A}
703 \DeclareFlexDelimiter{\Downarrow}{DeA}{del}{7F}{OMS}{2B}
704 \DeclareFlexDelimiter{\Updownarrow}{DeA}{del}{77}{OMS}{6D}
705 \DeclareFlexDelimiter{\backslash}{DeB}{del}{0F}{OMS}{6E}

```

### 3 Some compound symbols

The following symbols are not robust in standard L<sup>A</sup>T<sub>E</sub>X because they use # or \mathpalette (which is not robust and contains a # in its expansion): \angle, \cong, \notin, \rightleftharpoons.

In this definition of \hbar, the symbol is cobbled together from a math italic h and the cmr overbar accent glyph.

```

706 \DeclareFlexSymbol{\hbarOrd}{Ord}{OT1}{16}
707 \DeclareFlexCompoundSymbol{\hbar}{Ord}{\hbarOrd\mkern-9mu h}

```

For \surd, the interior symbol gets math class 1 (cumulative operator) to make the glyph vertically centered on the math axis, but the desired horizontal spacing is the spacing for a mathord. (Couldn't it just be class mathopen, though?)

```

708 \DeclareFlexSymbol{\surdOrd}{Ord}{OMS}{70}
709 \DeclareFlexCompoundSymbol{\surd}{Ord}{\mathop{\surdOrd}}

```

As shown in this definition of \angle, rule dimens are not allowed to use math-units, unfortunately.

```

710 \DeclareFlexCompoundSymbol{\angle}{Ord}{%
711   \vbox{\ialign{%
712     $\m@th\scriptstyle##$\crcr
713     \notRel\mathrel{\mkern14mu}\crcr
714     \noalign{\nointerlineskip}%
715     \mkern2.5mu\leaders\hrule \@height.34pt\hfill\mkern2.5mu\crcr
716   }}%
717 }

```

The \not function, which is defined in the flexisym package, requires a suitably defined \notRel symbol.

```

718 \DeclareFlexCompoundSymbol{\neq}{Rel}{\not{=}}
.
719 \DeclareFlexCompoundSymbol{\mapsto}{Rel}{\mapstoOrd\rightharpoonup}

```

The `\@vereq` function ends by centering the whole construction on the math axis, unlike `\buildrel` where the base symbol remains at its normal altitude. Furthermore, `\@vereq` leaves the math style of the top symbol as given instead of downsizing to scriptstyle.

```
720 \DeclareFlexCompoundSymbol{\cong}{Rel}{\mathpalette\@vereq\sim}
```

The `\m@th` in the `fontmath.ltx` definition of `\notin` is superfluous unless `\c@ncel` doesn't include it (which was perhaps true in an older version of `plain.tex`).

```
721 \providecommand*\joinord{}
722 \cmbase | mathptmx \renewcommand*\joinord{\mkern-3mu }
723 \mathpazo \renewcommand*\joinord{\mkern-3.45mu }
724 \DeclareFlexCompoundSymbol{\notin}{Rel}{\mathpalette\c@ncel\in}
725 \DeclareFlexCompoundSymbol{\rightleftharpoons}{Rel}{\mathpalette\rlh@{}}
726 \DeclareFlexCompoundSymbol{\doteq}{Rel}{\buildrel\textstyle.\over= }
727 \DeclareFlexCompoundSymbol{\hookrightarrow}{Rel}{\lhookRel\joinord\rightarrow}
728 \DeclareFlexCompoundSymbol{\hookleftarrow}{Rel}{\leftarrow\joinord\rhookRel}
729 \DeclareFlexCompoundSymbol{\bowtie}{Rel}{\triangleright\joinord\triangleleft}
730 \DeclareFlexCompoundSymbol{\models}{Rel}{\vert\joinord=}
731 \DeclareFlexCompoundSymbol{\Longrightarrow}{Rel}{\Relbar\joinord\rightarrow}
732 \DeclareFlexCompoundSymbol{\longrightarrow}{Rel}{\relbar\joinord\rightarrow}
733 \DeclareFlexCompoundSymbol{\Longleftarrow}{Rel}{\Leftarrow\joinord\Relbar}
734 \DeclareFlexCompoundSymbol{\longleftarrow}{Rel}{\leftarrow\joinord\relbar}
735 \DeclareFlexCompoundSymbol{\longmapsto}{Rel}{\mapstochar\longrightarrow}
736 \DeclareFlexCompoundSymbol{\longleftarrowtriarrow}{Rel}{\leftarrow\joinord\rightarrow}
737 \DeclareFlexCompoundSymbol{\Longleftarrowtriarrow}{Rel}{\Leftarrow\joinord\rightarrow}
```

Here is what you get from the old definition of `\iff`.

```
\glue 2.77771 plus 2.77771
\glue(\thickmuskip) 2.77771 plus 2.77771
\OMS/cmsy/m/n/10 (
\hbox(0.0+0.0)x-1.66663
.\kern -1.66663
\OMS/cmsy/m/n/10 )
\penalty 500
\glue 2.77771 plus 2.77771
\glue(\thickmuskip) 2.77771 plus 2.77771
```

Looks like it could be simplified slightly. But it's not so easy as it looks to do it without screwing up the line breaking possibilities.

```
738 \renewcommand*\iff{%
739 \mskip\thickmuskip\Longleftarrowtriarrow\mskip\thickmuskip
740 }
```

Some dotly symbols.

```
741 \DeclareFlexCompoundSymbol{\cdots}{Inn}{\cdotp\cdotp\cdotp}%
742 \DeclareFlexCompoundSymbol{\vdots}{Ord}{%
743 \vbox{\baselineskip4\p@ \lineskiplimit\z@
744 \kern6\p@\hbox{.}\hbox{.}\hbox{.}}}
745 \DeclareFlexCompoundSymbol{\ddots}{Inn}{%}
```

```

746 \mkern1mu\raise7\p@
747 \vbox{\kern7\p@\hbox{.}}\mkern2mu%
748 \raise4\p@\hbox{.}\mkern2mu\raise\p@\hbox{.}\mkern1mu%
749 }
.
750 \def\relbar{\begingroup \def\smash@{tb}% in case amsmath is loaded
751 \mathpalette\mathsm@sh{\mathchar"200 }\endgroup}
For \Relbar we take an equal sign of class 0 (Ord) from the operator family. For
cmr and mathptmx we know this is family 0.
752 <cmbase | mathptmx>\def\Relbar{\mathchar"3D }
For the mathpazo setup we need to use the equal sign from cmr and so must insert
class 0 and use the symbol from the upright symbols.
753 <mathpazo>\edef\Relbar{\mathchar\string\hexnumber@\symupright3D }
Done.
754 \ExplSyntaxOff
755 </cmbase | mathpazo | mathptmx>
Various synonyms such as \le for \leq and \to for \rightarrow are defined in
flexisym with \def instead of \let, for slower execution speed but smaller chance
of synchronization problems.
756 <*msabm>
757 \ProvidesSymbols{msabm}[2001/09/08 v0.91]
758 \ExplSyntaxOn
759 \RequirePackage{amsfonts}\relax
760 \cs_gset:cpx{mg@MSA}{\hexnumber@\symAMSa}%
761 \cs_gset:cpx{mg@MSB}{\hexnumber@\symAMSb}%
762 \DeclareFlexSymbol{\boxdot} {Bin}{MSA}{00}
763 \DeclareFlexSymbol{\boxplus} {Bin}{MSA}{01}
764 \DeclareFlexSymbol{\boxtimes} {Bin}{MSA}{02}
765 \DeclareFlexSymbol{\square} {Ord}{MSA}{03}
766 \DeclareFlexSymbol{\blacksquare} {Ord}{MSA}{04}
767 \DeclareFlexSymbol{\centerdot} {Bin}{MSA}{05}
768 \DeclareFlexSymbol{\lozenge} {Ord}{MSA}{06}
769 \DeclareFlexSymbol{\blacklozenge} {Ord}{MSA}{07}
770 \DeclareFlexSymbol{\circlearrowright} {Rel}{MSA}{08}
771 \DeclareFlexSymbol{\circlearrowleft} {Rel}{MSA}{09}
In amsfonts.sty:
772 %\DeclareFlexSymbol{\rightleftharpoons}{Rel}{MSA}{0A}
773 \DeclareFlexSymbol{\leftrightharpoons} {Rel}{MSA}{0B}
774 \DeclareFlexSymbol{\boxminus} {Bin}{MSA}{0C}
775 \DeclareFlexSymbol{\Vdash} {Rel}{MSA}{0D}
776 \DeclareFlexSymbol{\Vvdash} {Rel}{MSA}{0E}
777 \DeclareFlexSymbol{\vDash} {Rel}{MSA}{0F}
778 \DeclareFlexSymbol{\twoheadrightarrow} {Rel}{MSA}{10}
779 \DeclareFlexSymbol{\twoheadleftarrow} {Rel}{MSA}{11}

```

```

780 \DeclareFlexSymbol{\leftleftarrows} {Rel}{MSA}{12}
781 \DeclareFlexSymbol{\rightrightarrows} {Rel}{MSA}{13}
782 \DeclareFlexSymbol{\upuparrows} {Rel}{MSA}{14}
783 \DeclareFlexSymbol{\downdownarrows} {Rel}{MSA}{15}
784 \DeclareFlexSymbol{\upharpoonright} {Rel}{MSA}{16}
785 \let\restriction\upharpoonright
786 \DeclareFlexSymbol{\downharpoonright} {Rel}{MSA}{17}
787 \DeclareFlexSymbol{\upharpoonleft} {Rel}{MSA}{18}
788 \DeclareFlexSymbol{\downharpoonleft} {Rel}{MSA}{19}
789 \DeclareFlexSymbol{\rightarrowtail} {Rel}{MSA}{1A}
790 \DeclareFlexSymbol{\leftarrowtail} {Rel}{MSA}{1B}
791 \DeclareFlexSymbol{\leftrightarrows} {Rel}{MSA}{1C}
792 \DeclareFlexSymbol{\rightleftarrows} {Rel}{MSA}{1D}
793 \DeclareFlexSymbol{\Lsh} {Rel}{MSA}{1E}
794 \DeclareFlexSymbol{\Rsh} {Rel}{MSA}{1F}
795 \DeclareFlexSymbol{\rightsquigarrow} {Rel}{MSA}{20}
796 \DeclareFlexSymbol{\leftrightsquigarrow} {Rel}{MSA}{21}
797 \DeclareFlexSymbol{\looparrowleft} {Rel}{MSA}{22}
798 \DeclareFlexSymbol{\looparrowright} {Rel}{MSA}{23}
799 \DeclareFlexSymbol{\circeq} {Rel}{MSA}{24}
800 \DeclareFlexSymbol{\succsim} {Rel}{MSA}{25}
801 \DeclareFlexSymbol{\gtrsim} {Rel}{MSA}{26}
802 \DeclareFlexSymbol{\gtrapprox} {Rel}{MSA}{27}
803 \DeclareFlexSymbol{\multimap} {Rel}{MSA}{28}
804 \DeclareFlexSymbol{\therefore} {Rel}{MSA}{29}
805 \DeclareFlexSymbol{\because} {Rel}{MSA}{2A}
806 \DeclareFlexSymbol{\doteqdot} {Rel}{MSA}{2B}
807 \let\Doteq\doteqdot
808 \DeclareFlexSymbol{\triangleq} {Rel}{MSA}{2C}
809 \DeclareFlexSymbol{\precsim} {Rel}{MSA}{2D}
810 \DeclareFlexSymbol{\lessim} {Rel}{MSA}{2E}
811 \DeclareFlexSymbol{\lessapprox} {Rel}{MSA}{2F}
812 \DeclareFlexSymbol{\eqslantless} {Rel}{MSA}{30}
813 \DeclareFlexSymbol{\eqslantgtr} {Rel}{MSA}{31}
814 \DeclareFlexSymbol{\curlyeqprec} {Rel}{MSA}{32}
815 \DeclareFlexSymbol{\curlyeqsucc} {Rel}{MSA}{33}
816 \DeclareFlexSymbol{\preccurlyeq} {Rel}{MSA}{34}
817 \DeclareFlexSymbol{\leqq} {Rel}{MSA}{35}
818 \DeclareFlexSymbol{\leqslant} {Rel}{MSA}{36}
819 \DeclareFlexSymbol{\lessgtr} {Rel}{MSA}{37}
820 \DeclareFlexSymbol{\backprime} {Ord}{MSA}{38}
821 \DeclareFlexSymbol{\risingdotseq} {Rel}{MSA}{3A}
822 \DeclareFlexSymbol{\fallingdotseq} {Rel}{MSA}{3B}
823 \DeclareFlexSymbol{\succcurlyeq} {Rel}{MSA}{3C}
824 \DeclareFlexSymbol{\geqq} {Rel}{MSA}{3D}
825 \DeclareFlexSymbol{\geqslant} {Rel}{MSA}{3E}
826 \DeclareFlexSymbol{\gtrless} {Rel}{MSA}{3F}

in amsfonts.sty
827 %% \DeclareFlexSymbol{\sqsubset} {Rel}{MSA}{40}

```

```

828 %% \DeclareFlexSymbol{\sqsupset} {Rel}{MSA}{41}
829 \DeclareFlexSymbol{\vartriangleright}{Rel}{MSA}{42}
830 \DeclareFlexSymbol{\vartriangleleft} {Rel}{MSA}{43}
831 \DeclareFlexSymbol{\trianglerighteq} {Rel}{MSA}{44}
832 \DeclareFlexSymbol{\trianglelefteq} {Rel}{MSA}{45}
833 \DeclareFlexSymbol{\bigstar} {Ord}{MSA}{46}
834 \DeclareFlexSymbol{\between} {Rel}{MSA}{47}
835 \DeclareFlexSymbol{\blacktriangledown} {Ord}{MSA}{48}
836 \DeclareFlexSymbol{\blacktriangleright} {Rel}{MSA}{49}
837 \DeclareFlexSymbol{\blacktriangleleft} {Rel}{MSA}{4A}
838 \DeclareFlexSymbol{\vartriangle} {Rel}{MSA}{4D}
839 \DeclareFlexSymbol{\blacktriangle} {Ord}{MSA}{4E}
840 \DeclareFlexSymbol{\triangledown} {Ord}{MSA}{4F}
841 \DeclareFlexSymbol{\eqcirc} {Rel}{MSA}{50}
842 \DeclareFlexSymbol{\lesseqgtr} {Rel}{MSA}{51}
843 \DeclareFlexSymbol{\gtreqless} {Rel}{MSA}{52}
844 \DeclareFlexSymbol{\lesseqgtr} {Rel}{MSA}{53}
845 \DeclareFlexSymbol{\gtreqless} {Rel}{MSA}{54}
846 \DeclareFlexSymbol{\Rrightarrow} {Rel}{MSA}{56}
847 \DeclareFlexSymbol{\Lleftarrow} {Rel}{MSA}{57}
848 \DeclareFlexSymbol{\veebar} {Bin}{MSA}{59}
849 \DeclareFlexSymbol{\barwedge} {Bin}{MSA}{5A}
850 \DeclareFlexSymbol{\doublebarwedge} {Bin}{MSA}{5B}

In amsfonts.sty
851 %%\DeclareFlexSymbol{\angle} {Ord}{MSA}{5C}
852 \DeclareFlexSymbol{\measuredangle} {Ord}{MSA}{5D}
853 \DeclareFlexSymbol{\sphericalangle} {Ord}{MSA}{5E}
854 \DeclareFlexSymbol{\varpropto} {Rel}{MSA}{5F}
855 \DeclareFlexSymbol{\smallsmile} {Rel}{MSA}{60}
856 \DeclareFlexSymbol{\smallfrown} {Rel}{MSA}{61}
857 \DeclareFlexSymbol{\Subset} {Rel}{MSA}{62}
858 \DeclareFlexSymbol{\Supset} {Rel}{MSA}{63}
859 \DeclareFlexSymbol{\Cup} {Bin}{MSA}{64}
860 \let\doublecup\Cup
861 \DeclareFlexSymbol{\Cap} {Bin}{MSA}{65}
862 \let\doublecap\Cap
863 \DeclareFlexSymbol{\curlywedge} {Bin}{MSA}{66}
864 \DeclareFlexSymbol{\curlyvee} {Bin}{MSA}{67}
865 \DeclareFlexSymbol{\leftthreetimes} {Bin}{MSA}{68}
866 \DeclareFlexSymbol{\rightthreetimes} {Bin}{MSA}{69}
867 \DeclareFlexSymbol{\subteqq} {Rel}{MSA}{6A}
868 \DeclareFlexSymbol{\supseteqq} {Rel}{MSA}{6B}
869 \DeclareFlexSymbol{\bumpeq} {Rel}{MSA}{6C}
870 \DeclareFlexSymbol{\Bumpeq} {Rel}{MSA}{6D}
871 \DeclareFlexSymbol{\lll} {Rel}{MSA}{6E}
872 \let\llless\lll
873 \DeclareFlexSymbol{\ggg} {Rel}{MSA}{6F}
874 \let\gggtr\ggg
875 \DeclareFlexSymbol{\circledS} {Ord}{MSA}{73}

```



876	<code>\DeclareFlexSymbol{\pitchfork}</code>	<code>{Rel}{MSA}{74}</code>
877	<code>\DeclareFlexSymbol{\dotplus}</code>	<code>{Bin}{MSA}{75}</code>
878	<code>\DeclareFlexSymbol{\backsim}</code>	<code>{Rel}{MSA}{76}</code>
879	<code>\DeclareFlexSymbol{\backsimeq}</code>	<code>{Rel}{MSA}{77}</code>
880	<code>\DeclareFlexSymbol{\complement}</code>	<code>{Ord}{MSA}{7B}</code>
881	<code>\DeclareFlexSymbol{\intercal}</code>	<code>{Bin}{MSA}{7C}</code>
882	<code>\DeclareFlexSymbol{\circledcirc}</code>	<code>{Bin}{MSA}{7D}</code>
883	<code>\DeclareFlexSymbol{\circledast}</code>	<code>{Bin}{MSA}{7E}</code>
884	<code>\DeclareFlexSymbol{\circleddash}</code>	<code>{Bin}{MSA}{7F}</code>

Begin AMSb declarations

885	<code>\DeclareFlexSymbol{\lvertneqq}</code>	<code>{Rel}{MSB}{00}</code>
886	<code>\DeclareFlexSymbol{\gvertneqq}</code>	<code>{Rel}{MSB}{01}</code>
887	<code>\DeclareFlexSymbol{\nleq}</code>	<code>{Rel}{MSB}{02}</code>
888	<code>\DeclareFlexSymbol{\ngeq}</code>	<code>{Rel}{MSB}{03}</code>
889	<code>\DeclareFlexSymbol{\nless}</code>	<code>{Rel}{MSB}{04}</code>
890	<code>\DeclareFlexSymbol{\ngtr}</code>	<code>{Rel}{MSB}{05}</code>
891	<code>\DeclareFlexSymbol{\nprec}</code>	<code>{Rel}{MSB}{06}</code>
892	<code>\DeclareFlexSymbol{\nsucc}</code>	<code>{Rel}{MSB}{07}</code>
893	<code>\DeclareFlexSymbol{\lneqq}</code>	<code>{Rel}{MSB}{08}</code>
894	<code>\DeclareFlexSymbol{\gneqq}</code>	<code>{Rel}{MSB}{09}</code>
895	<code>\DeclareFlexSymbol{\nleqslant}</code>	<code>{Rel}{MSB}{0A}</code>
896	<code>\DeclareFlexSymbol{\ngeqslant}</code>	<code>{Rel}{MSB}{0B}</code>
897	<code>\DeclareFlexSymbol{\lneq}</code>	<code>{Rel}{MSB}{0C}</code>
898	<code>\DeclareFlexSymbol{\gneq}</code>	<code>{Rel}{MSB}{0D}</code>
899	<code>\DeclareFlexSymbol{\npreceq}</code>	<code>{Rel}{MSB}{0E}</code>
900	<code>\DeclareFlexSymbol{\nsucceq}</code>	<code>{Rel}{MSB}{0F}</code>
901	<code>\DeclareFlexSymbol{\precnsim}</code>	<code>{Rel}{MSB}{10}</code>
902	<code>\DeclareFlexSymbol{\succnsim}</code>	<code>{Rel}{MSB}{11}</code>
903	<code>\DeclareFlexSymbol{\lnsim}</code>	<code>{Rel}{MSB}{12}</code>
904	<code>\DeclareFlexSymbol{\gnsim}</code>	<code>{Rel}{MSB}{13}</code>
905	<code>\DeclareFlexSymbol{\nleqq}</code>	<code>{Rel}{MSB}{14}</code>
906	<code>\DeclareFlexSymbol{\ngeqq}</code>	<code>{Rel}{MSB}{15}</code>
907	<code>\DeclareFlexSymbol{\precneqq}</code>	<code>{Rel}{MSB}{16}</code>
908	<code>\DeclareFlexSymbol{\succneqq}</code>	<code>{Rel}{MSB}{17}</code>
909	<code>\DeclareFlexSymbol{\precnapprox}</code>	<code>{Rel}{MSB}{18}</code>
910	<code>\DeclareFlexSymbol{\succnapprox}</code>	<code>{Rel}{MSB}{19}</code>
911	<code>\DeclareFlexSymbol{\lnapprox}</code>	<code>{Rel}{MSB}{1A}</code>
912	<code>\DeclareFlexSymbol{\gnapprox}</code>	<code>{Rel}{MSB}{1B}</code>
913	<code>\DeclareFlexSymbol{\nsim}</code>	<code>{Rel}{MSB}{1C}</code>
914	<code>\DeclareFlexSymbol{\ncong}</code>	<code>{Rel}{MSB}{1D}</code>
915	<code>\DeclareFlexSymbol{\diagup}</code>	<code>{Ord}{MSB}{1E}</code>
916	<code>\DeclareFlexSymbol{\diagdown}</code>	<code>{Ord}{MSB}{1F}</code>
917	<code>\DeclareFlexSymbol{\varsubsetneq}</code>	<code>{Rel}{MSB}{20}</code>
918	<code>\DeclareFlexSymbol{\varsupsetneq}</code>	<code>{Rel}{MSB}{21}</code>
919	<code>\DeclareFlexSymbol{\nsubseteq}</code>	<code>{Rel}{MSB}{22}</code>
920	<code>\DeclareFlexSymbol{\nsupseteq}</code>	<code>{Rel}{MSB}{23}</code>
921	<code>\DeclareFlexSymbol{\subsetneqq}</code>	<code>{Rel}{MSB}{24}</code>
922	<code>\DeclareFlexSymbol{\supsetneqq}</code>	<code>{Rel}{MSB}{25}</code>
923	<code>\DeclareFlexSymbol{\varsubsetneqq}</code>	<code>{Rel}{MSB}{26}</code>

```

924 \DeclareFlexSymbol{\varsupsetneqq} {Rel}{MSB}{27}
925 \DeclareFlexSymbol{\subsetneq} {Rel}{MSB}{28}
926 \DeclareFlexSymbol{\supsetneq} {Rel}{MSB}{29}
927 \DeclareFlexSymbol{\nsubseteq} {Rel}{MSB}{2A}
928 \DeclareFlexSymbol{\nsupseteq} {Rel}{MSB}{2B}
929 \DeclareFlexSymbol{\nparallel} {Rel}{MSB}{2C}
930 \DeclareFlexSymbol{\nmid} {Rel}{MSB}{2D}
931 \DeclareFlexSymbol{\nshortmid} {Rel}{MSB}{2E}
932 \DeclareFlexSymbol{\nshortparallel} {Rel}{MSB}{2F}
933 \DeclareFlexSymbol{\nvdash} {Rel}{MSB}{30}
934 \DeclareFlexSymbol{\nVdash} {Rel}{MSB}{31}
935 \DeclareFlexSymbol{\nvDash} {Rel}{MSB}{32}
936 \DeclareFlexSymbol{\nVDash} {Rel}{MSB}{33}
937 \DeclareFlexSymbol{\ntrianglerighteq} {Rel}{MSB}{34}
938 \DeclareFlexSymbol{\ntrianglelefteq} {Rel}{MSB}{35}
939 \DeclareFlexSymbol{\ntriangleleft} {Rel}{MSB}{36}
940 \DeclareFlexSymbol{\ntriangleright} {Rel}{MSB}{37}
941 \DeclareFlexSymbol{\nleftarrow} {Rel}{MSB}{38}
942 \DeclareFlexSymbol{\nrightarrow} {Rel}{MSB}{39}
943 \DeclareFlexSymbol{\nLeftarrow} {Rel}{MSB}{3A}
944 \DeclareFlexSymbol{\nRightarrow} {Rel}{MSB}{3B}
945 \DeclareFlexSymbol{\nLeftrightarrow} {Rel}{MSB}{3C}
946 \DeclareFlexSymbol{\nleqtrightharpoonup} {Rel}{MSB}{3D}
947 \DeclareFlexSymbol{\divideontimes} {Bin}{MSB}{3E}
948 \DeclareFlexSymbol{\varnothing} {Ord}{MSB}{3F}
949 \DeclareFlexSymbol{\nexists} {Ord}{MSB}{40}
950 \DeclareFlexSymbol{\Finv} {Ord}{MSB}{60}
951 \DeclareFlexSymbol{\Game} {Ord}{MSB}{61}

```

In amsfonts.sty:

```

952 %\DeclareFlexSymbol{\mho} {Ord}{MSB}{66}
953 \DeclareFlexSymbol{\eth} {Ord}{MSB}{67}
954 \DeclareFlexSymbol{\eqsim} {Rel}{MSB}{68}
955 \DeclareFlexSymbol{\beth} {Ord}{MSB}{69}
956 \DeclareFlexSymbol{\gimel} {Ord}{MSB}{6A}
957 \DeclareFlexSymbol{\daleth} {Ord}{MSB}{6B}
958 \DeclareFlexSymbol{\lessdot} {Bin}{MSB}{6C}
959 \DeclareFlexSymbol{\gtrdot} {Bin}{MSB}{6D}
960 \DeclareFlexSymbol{\ltimes} {Bin}{MSB}{6E}
961 \DeclareFlexSymbol{\rtimes} {Bin}{MSB}{6F}
962 \DeclareFlexSymbol{\shortmid} {Rel}{MSB}{70}
963 \DeclareFlexSymbol{\shortparallel} {Rel}{MSB}{71}
964 \DeclareFlexSymbol{\smallsetminus} {Bin}{MSB}{72}
965 \DeclareFlexSymbol{\thicksim} {Rel}{MSB}{73}
966 \DeclareFlexSymbol{\thickapprox} {Rel}{MSB}{74}
967 \DeclareFlexSymbol{\approxeq} {Rel}{MSB}{75}
968 \DeclareFlexSymbol{\succapprox} {Rel}{MSB}{76}
969 \DeclareFlexSymbol{\precapprox} {Rel}{MSB}{77}
970 \DeclareFlexSymbol{\curvearrowleft} {Rel}{MSB}{78}
971 \DeclareFlexSymbol{\curvearrowright} {Rel}{MSB}{79}

```

```

972 \DeclareFlexSymbol{\digamma}      {Ord}{MSB}{7A}
973 \DeclareFlexSymbol{\varkappa}     {Ord}{MSB}{7B}
974 \DeclareFlexSymbol{\Bbbk}          {Ord}{MSB}{7C}
975 \DeclareFlexSymbol{\hslash}        {Ord}{MSB}{7D}

  In amsfonts.sty:
976 %%\DeclareFlexSymbol{\hbar}        {Ord}{MSB}{7E}
977 \DeclareFlexSymbol{\backepsilon}   {Rel}{MSB}{7F}
978 \ExplSyntaxOff
979 </msabm>

```