

xCJK2uni 宏包

李清

sobenlee@gmail.com

2016/02/02 v0.5*

1 简介

xCJK2uni 是一个 L^AT_EX 宏包, 提供了将 CJK 文字转换成 Unicode 的功能。以宏包形式实现 gbk2uni 程序的功能, 使得使用 CJK 和 hyperref 宏包时, 可以得到正确的书签。并提供了 /ToUnicode 映射文件, 使得使用 GBK 等编码时, 配合 cmap 宏包, 用 pdfL^AT_EX 编译得到的 pdf 文件支持查找、复制和粘贴。

xCJK2uni 只适用于下一节说明的有效编码和以 pdfT_EX 作为编译引擎的工作方式。

xCJK2uni 依赖 l3kernel。应该与 CJK 或 CCT 等配合使用, 但 xCJK2uni 不会自动引入它们。

2 基本命令

`\useCJKencmap` `\useCJKencmap {⟨Bg5|Bg5+|GB|GBK|JIS|KS⟩}`

设置当前的编码环境。以上是 xCJK2uni 支持的文字编码, 缺省使用 GBK。xCJK2uni 会在 CJK 环境内自动更新支持的编码。

`\CJKchartouni` * `\CJKchartouni {⟨单个 CJK 文字⟩}`

将文字转换成 16 进制形式的 Unicode。例如 `\CJKchartouni{一}` 将得到 4E00。将它展开两次就可以得到结果。

`\CJKsfdtouni` * `\CJKsfdtouni {⟨plane⟩} {⟨slot⟩}`

`⟨plane⟩` 表示由 .sfd 文件定义的字体的编号, `⟨slot⟩` 表示在子字体中的位置, 这个命令将得到该坐标位置的 Unicode。将它展开两次就可以得到结果。如果在当前编码下, 该坐标位置不合法, 将展开为空。

3 代码实现

3.1 xCJK2uni

```
1 (*package)
2 <@@=CJKtu>
3 \msg_new:nnn { xCJK2uni } { 13-too-old }
4 {
5   Support~package~'expl3'~too~old. \\\
6   Please~update~an~up~to~date~version~of~the~bundles\\
7   'l3kernel'~and~'l3packages'\\
8   using~your~TeX~package~manager~or~from~CTAN.
9 }
10 \ifpackageafter { expl3 } { 2015/09/24 } { }
11 { \msg_error:nn { xCJK2uni } { 13-too-old } }
12 \sys_if_engine_pdfTeX:F
13 {
14   \msg_new:nnn { xCJK2uni } { pdflatex }
15   { The~xCJK2uni~package~is~only~supported~in~pdfTeX. }
```

*ctex-kit rev. 917f10e.

```

16 \msg_critical:nn { xCJK2uni } { pdflatex }
17 }
18 \tl_new:N \l__CJKtu_sfd_tl
19 \prop_new:N \c__CJKtu_sfd_prop
20 \seq_new:N \c__CJKtu_encoding_seq
21 \int_new:N \l__CJKtu_i_min_int
22 \int_new:N \l__CJKtu_i_max_int
23 \int_new:N \l__CJKtu_ii_min_int
24 \int_new:N \l__CJKtu_ii_max_int
25 \int_new:N \l__CJKtu_gap_begin_int
26 \int_new:N \l__CJKtu_gap_end_int
27 \int_new:N \l__CJKtu_plane_int
28 \int_set:Nn \l__CJKtu_i_max_int { "FE }
29 \int_set:Nn \l__CJKtu_ii_max_int { "FE }

```

__CJKtu_set_enc:nnnnnn

```

30 \cs_new_protected_nopar:Npn \__CJKtu_set_enc:nnnnnn #1#2#3#4#5#6
31 {
32   \seq_put_right:Nn \c__CJKtu_encoding_seq {#1}
33   \prop_gput:Nnn \c__CJKtu_sfd_prop {#1} {#2}
34   \int_const:cn { c__CJKtu_ #1 _i_min_int } {#3}
35   \int_const:cn { c__CJKtu_ #1 _ii_min_int } {#4}
36   \tl_if_empty:nTF {#5}
37   {
38     \int_const:cn { c__CJKtu_ #1 _gap_begin_int }
39     { \l__CJKtu_ii_max_int + \c_one }
40     \int_const:cn { c__CJKtu_ #1 _gap_end_int }
41     { \use:c { c__CJKtu_ #1 _gap_begin_int } }
42     \int_const:cn { c__CJKtu_ #1 _plane_int }
43     { \l__CJKtu_ii_max_int - (#4) + \c_one }
44   }
45   {
46     \int_const:cn { c__CJKtu_ #1 _gap_begin_int } { #5 + \c_one }
47     \int_const:cn { c__CJKtu_ #1 _gap_end_int } {#6}
48     \int_const:cn { c__CJKtu_ #1 _plane_int }
49     { (#5) - (#4) + \l__CJKtu_ii_max_int - (#6) + \c_two }
50   }
51   \cs_if_exist_use:NTF \CJKaddEncHook
52   { {#1} { \__CJKtu_enc_map:nn {#1} {#2} } }
53   {
54     \AtBeginDocument
55     {
56       \cs_if_exist_use:NT \CJKaddEncHook
57       { {#1} { \__CJKtu_enc_map:nn {#1} {#2} } }
58     }
59   }
60 }

```

(End definition for __CJKtu_set_enc:nnnnnn.)

__CJKtu_enc_map:nn

```

61 \cs_new_protected_nopar:Npn \__CJKtu_enc_map:nn #1#2
62 {
63   \int_set_eq:Nc \l__CJKtu_i_min_int { c__CJKtu_ #1 _i_min_int }
64   \int_set_eq:Nc \l__CJKtu_ii_min_int { c__CJKtu_ #1 _ii_min_int }
65   \int_set_eq:Nc \l__CJKtu_gap_begin_int { c__CJKtu_ #1 _gap_begin_int }
66   \int_set_eq:Nc \l__CJKtu_gap_end_int { c__CJKtu_ #1 _gap_end_int }
67   \int_set_eq:Nc \l__CJKtu_plane_int { c__CJKtu_ #1 _plane_int }
68   \cs_if_exist:cF { CJKtu_ #1 _sfd_map:nn }
69   { \group_begin: \file_input:n { xCJK2uni-#2.def } \group_end: }
70   \cs_set_eq:Nc \CJKtu_sfd_map:nn { CJKtu_ #1 _sfd_map:nn }
71 }

```

(End definition for __CJKtu_enc_map:nn.)

__CJKtu_use_enc_map:n

```

72 \cs_new_protected_nopar:Npn \__CJKtu_use_enc_map:n #1
73 {

```

```

74 \prop_get:NxNTF \c__CJKtu_sfd_prop {#1} \l__CJKtu_sfd_tl
75 { \__CJKtu_enc_map:nn {#1} { \l__CJKtu_sfd_tl } }
76 { \msg_error:nxx { xCJK2uni } { invalid-encoding } {#1} }
77 }
78 \msg_new:nnn { xCJK2uni } { invalid-encoding }
79 {
80 The~encoding~`#1'~is~invalid.\\
81 Only~\seq_use:Nnnn \c__CJKtu_encoding_seq { ~and~ } { ,~ } { ,~and~ } ~
82 are~supported.
83 }
84 \cs_generate_variant:Nn \prop_get:NnNTF { Nx }

```

(End definition for __CJKtu_use_enc_map:n.)

```

85 \__CJKtu_set_enc:nnnnnn { GB } { UGB } { "A1 } { "A1 } { } { }
86 \__CJKtu_set_enc:nnnnnn { JIS } { UJIS } { "A1 } { "A1 } { } { }
87 \__CJKtu_set_enc:nnnnnn { KS } { UKS } { "A1 } { "A1 } { } { }
88 \__CJKtu_set_enc:nnnnnn { Bg5 } { UBig5 } { "A1 } { "A0 } { "7E } { "A1 }
89 \__CJKtu_set_enc:nnnnnn { Bg5+ } { UBg5plus } { "81 } { "40 } { "7E } { "80 }
90 \__CJKtu_set_enc:nnnnnn { GBK } { UGBK } { "81 } { "40 } { "7E } { "80 }

```

__CJKtu_sfd_to_unicode:nn

```

91 \cs_new_nopar:Npn \__CJKtu_sfd_to_unicode:nn #1#2
92 {
93 \exp:w \exp_last_unbraced:Nf
94 \__CJKtu_get_unicode:w \CJKtu_sfd_map:nn {#1} {#2} \exp_end:
95 }

```

(End definition for __CJKtu_sfd_to_unicode:nn.)

__CJKtu_get_unicode:w

```

96 \cs_new_nopar:Npn \__CJKtu_get_unicode:w #1 \exp_end:
97 {
98 \tl_if_empty:nTF {#1}
99 { \exp_end: }
100 { \__CJKtu_get_unicode_aux:w #1 \q_stop }
101 }
102 \cs_new_nopar:Npn \__CJKtu_get_unicode_aux:w #1 \or: #2 \q_stop
103 { \or: #2 \exp_end: #1 }

```

(End definition for __CJKtu_get_unicode:w.)

__CJKtu_char_to_unicode:n

```

104 \group_begin:
105 \char_set_catcode_active:n { "7F }
106 \cs_new_nopar:Npn \__CJKtu_char_to_unicode:n #1
107 {
108 \exp:w
109 \int_compare:nNnTF { \tl_count:n {#1} } = \c_two
110 { \CJKtu_byte:NN #1 }
111 {
112 \tl_if_head_eq_meaning:nNTF {#1} ^^7f
113 { \__CJKtu_byte:wNnw #1 }
114 { \__CJKtu_encoding_error: }
115 }
116 }
117 \cs_new_nopar:Npn \__CJKtu_byte:wNnw ^^7f #1 ^^7f #2 ^^7f
118 { \CJKtu_byte_map:nn { `#1 } {#2} }
119 \cs_new_nopar:Npn \__CJKtu_byte:Nnw #1 ^^7f #2 ^^7f
120 { \CJKtu_byte_map:nn { `#1 } {#2} }

```

(End definition for __CJKtu_char_to_unicode:n.)

\CJKtu_bookmarks_hook:

```

121 \cs_new_protected_nopar:Npn \CJKtu_bookmarks_hook:
122 {
123 \cs_set_eq:NN ^^7f \__CJKtu_byte:Nnw
124 \cs_set_eq:NN \CJK@XX \CJKtu_byte:NN
125 \cs_set_eq:NN \CJKchar \__CJKtu_byte:wnn

```

```

126     \cs_set_eq:NN \CCTchar \CJKtu_byte:nn
127     \cs_set_eq:NN \@CCTSetChar \_CJKtu_CCT_byte:NNN
128     \cs_set_eq:NN \_CJKtu_get_unicode:n \_CJKtu_UTF_xvi_be:n
129   }
130 \group_end:

```

(End definition for \CJKtu_bookmarks_hook:.)

```

\CJKtu_byte:NN
\_CJKtu_CCT_byte:NNN
131 \cs_new_nopar:Npn \CJKtu_byte:NN #1#2
132 { \CJKtu_byte:nn { `#1 } { `#2 } }
133 \cs_new_nopar:Npn \_CJKtu_CCT_byte:NNN #1#2#3
134 { \CJKtu_byte:nn { `#1 } { `#3 } }

```

(End definition for \CJKtu_byte:NN and _CJKtu_CCT_byte:NNN.)

```

\_CJKtu_byte:wnn
135 \cs_new_nopar:Npn \_CJKtu_byte:wnn #1# { \CJKtu_byte:nn }

```

(End definition for _CJKtu_byte:wnn.)

```

\CJKtu_byte:nn
136 \cs_new_nopar:Npn \CJKtu_byte:nn #1#2
137 {
138   \int_compare:nTF { \l__CJKtu_i_min_int <= #1 <= \l__CJKtu_i_max_int }
139   {
140     \int_compare:nTF { \l__CJKtu_ii_min_int <= #2 < \l__CJKtu_gap_begin_int }
141     { \CJKtu_byte_map:nn {#1} {#2} }
142     {
143       \int_compare:nTF
144       { \l__CJKtu_gap_end_int <= #2 <= \l__CJKtu_ii_max_int }
145       { \CJKtu_byte_map:nn {#1} {#2} }
146       { \_CJKtu_encoding_error: }
147     }
148   }
149   { \_CJKtu_encoding_error: }
150 }
151 \cs_new_nopar:Npn \_CJKtu_encoding_error:
152 {
153   \msg_expandable_error:nn { xCJK2uni } { encoding-error }
154   \exp_end:
155 }
156 \msg_new:nnn { xCJK2uni } { encoding-error }
157 { Wrong~encoding~scheme. }

```

(End definition for \CJKtu_byte:nn.)

```

\CJKtu_byte_map:nn
158 \cs_new_nopar:Npn \CJKtu_byte_map:nn #1#2
159 {
160   \exp_args:Nf \_CJKtu_index_map:n
161   {
162     \int_eval:n
163     {
164       ( #1 - \l__CJKtu_i_min_int ) * \l__CJKtu_plane_int
165       + (#2) - \l__CJKtu_ii_min_int
166       \int_compare:nNnF {#2} < \l__CJKtu_gap_begin_int
167       { + \l__CJKtu_gap_begin_int - \l__CJKtu_gap_end_int }
168     }
169   }
170 }

```

(End definition for \CJKtu_byte_map:nn.)

```

\_CJKtu_index_map:n
171 \cs_new_nopar:Npn \_CJKtu_index_map:n #1
172 {
173   \_CJKtu_get_unicode:n
174   {

```

```

175     \CJKtu_sfd_map:nn
176     { \int_div_truncate:nn {#1} { \c_two_hundred_fifty_six } + \c_one }
177     { \int_mod:nn {#1} { \c_two_hundred_fifty_six } }
178   }
179 }

```

(End definition for _CJKtu_index_map:n)

_CJKtu_get_unicode:n

```

180 \cs_new_nopar:Npn \_CJKtu_get_unicode:n #1
181 { \exp_last_unbraced:Nf \_CJKtu_get_unicode:w #1 \exp_end: }

```

(End definition for _CJKtu_get_unicode:n)

_CJKtu_UTF_xvi_be:n

```

182 \cs_new_nopar:Npn \_CJKtu_UTF_xvi_be:n #1
183 { \exp_args:Nf \_CJKtu_UTF_xvi_aux:n { \int_eval:n { "#1" } } }
184 \cs_new_nopar:Npn \_CJKtu_UTF_xvi_aux:n #1
185 {
186   \HyPsd@DecimalToOctalFirst
187   { \int_div_truncate:nn {#1} { \c_two_hundred_fifty_six } }
188   \HyPsd@DecimalToOctalSecond
189   { \int_mod:nn {#1} { \c_two_hundred_fifty_six } }
190 }

```

(End definition for _CJKtu_UTF_xvi_be:n)

```

191 \cs_if_exist_use:NTF \hypersetup { { unicode , CJKbookmarks = false } }
192 { \PassOptionsToPackage { unicode , CJKbookmarks = false } { hyperref } }
193 \msg_new:nnn { xCJK2uni } { UTF-8-encoding }
194 { It~is~not~necessary~to~load~xCJK2uni~in~UTF-8~encoding. }
195 \tl_if_exist:NF \pdfstringdefPreHook { \tl_new:N \pdfstringdefPreHook }
196 \@ifpackageloaded { CJKutf8 }
197 { \msg_warning:nn { xCJK2uni } { UTF-8-encoding } }
198 {
199   \cs_new_eq:NN \CJKtu@bookmarks@hook \CJKtu_bookmarks_hook:
200   \tl_gput_right:Nn \pdfstringdefPreHook { \CJKtu@bookmarks@hook }
201   \AtBeginDocument
202   {
203     \@ifpackageloaded { CJKutf8 }
204     {
205       \cs_gset_eq:NN \CJKtu@bookmarks@hook \scan_stop:
206       \msg_warning:nn { xCJK2uni } { UTF-8-encoding }
207     }
208     { \cs_if_exist_use:NT \hypersetup { { CJKbookmarks = false } } }
209   }
210 }

```

\useCJKenomap

\CJKchartouni

\CJKsfdtouni

```

211 \cs_new_eq:NN \useCJKenomap \_CJKtu_use_enc_map:n
212 \cs_new_eq:NN \CJKchartouni \_CJKtu_char_to_unicode:n
213 \cs_new_eq:NN \CJKsfdtouni \_CJKtu_sfd_to_unicode:nn

```

(End definition for \useCJKenomap, \CJKchartouni, and \CJKsfdtouni. These functions are documented on page 1.)

_CJKtu_initial:N

```

214 \cs_new_protected_nopar:Npn \_CJKtu_initial:N #1
215 {
216   \tl_if_exist:NTF #1
217   {
218     \prop_get:NxNTF \c__CJKtu_sfd_prop {#1} \l__CJKtu_sfd_tl
219     { \_CJKtu_enc_map:nn {#1} { \l__CJKtu_sfd_tl } }
220     { \_CJKtu_use_enc_map:n { GBK } }
221   }
222   { \_CJKtu_use_enc_map:n { GBK } }
223 }

```

(End definition for _CJKtu_initial:N)

```

224 \exp_args:Nc \_CJKtu_initial:N { CJK @ @ @ enc }

```

<package>

3.2 xCJK2uni-make

```
226 <*make>
227 \msg_new:nnn { xCJK2uni } { file-not-found }
228 {
229   SFD file~`#2'~for~encoding~`#1'~not~found.\\
230   xCJK2uni~will~not~work!
231 }
232 \ior_new:N \g__CJKtu_sfd_ior
233 \str_new:N \g__CJKtu_path_str
234 \sys_if_engine luatex:TF
235 {
236   \str_gset:Nx \g__CJKtu_path_str
237   {
238     \lua_now_x:n
239     {
240       kpse.set_program_name("luatex") ~
241       local ~ sfd = kpse.find_file("UGBK.sfd", "subfont~definition~files") ~
242       if ~ sfd ~ then ~
243         tex.write(sfd) ~
244       end
245     }
246   }
247 }
248 {
249   \ior_open:NnTF \g__CJKtu_sfd_ior { |kpsewhich ~ UGBK.sfd }
250   {
251     \ior_get_str:NN \g__CJKtu_sfd_ior \g__CJKtu_path_str
252     \ior_close:N \g__CJKtu_sfd_ior
253   }
254   { \msg_fatal:nmmm { xCJK2uni } { file-not-found } { GBK } { UGBK.sfd } }
255 }
256 \tl_if_blank:oTF { \g__CJKtu_path_str }
257 { \msg_fatal:nmmm { xCJK2uni } { file-not-found } { GBK } { UGBK.sfd } }
258 {
259   \seq_set_split:NnV \l_tmpa_seq { / } \g__CJKtu_path_str
260   \seq_pop_right:NN \l_tmpa_seq \g__CJKtu_path_str
261   \str_gset:Nx \g__CJKtu_path_str { \seq_use:Nn \l_tmpa_seq { / } }
262 }
263 \str_const:Nx \c__CJKtu_indent_str
264 { \c_catcode_other_space_tl \c_catcode_other_space_tl }
265 \iow_new:N \g__CJKtu_sfd_map_iow
266 \iow_new:N \g__CJKtu_sfd_cmap_iow
267 \seq_new:N \l__CJKtu_sfd_plane_seq
268 \seq_new:N \l__CJKtu_sfd_line_seq
269 \tl_new:N \l__CJKtu_sfd_plane_tl
270 \tl_new:N \l__CJKtu_cmap_block_tl
271 \int_new:N \l__CJKtu_sfd_index_int

```

\c__CJKtu_sfd_map_prop

```
272 \prop_new:N \c__CJKtu_sfd_map_prop
273 \prop_gput:Nnn \c__CJKtu_sfd_map_prop { GB } { { 10 } { UGB } }
274 \prop_gput:Nnn \c__CJKtu_sfd_map_prop { JIS } { { 40 } { UJIS } }
275 \prop_gput:Nnn \c__CJKtu_sfd_map_prop { KS } { { 60 } { UKS } }
276 \prop_gput:Nnn \c__CJKtu_sfd_map_prop { Bg5 } { { 00 } { UBig5 } }
277 \prop_gput:Nnn \c__CJKtu_sfd_map_prop { Bg5+ } { { 09 } { UBg5plus } }
278 \prop_gput:Nnn \c__CJKtu_sfd_map_prop { GBK } { { 19 } { UGBK } }

```

(End definition for \c__CJKtu_sfd_map_prop.)

__CJKtu_write_file:nnn

```
279 \cs_new_protected_nopar:Npn \__CJKtu_write_file:nnn #1#2#3
280 {
281   \group_begin:
282   \file_path_include:n { \g__CJKtu_path_str / }
283   \ior_open:NnTF \g__CJKtu_sfd_ior { #3.sfd }
284   {
285     \group_end:

```

```

286     \seq_clear:N \l__CJktu_sfd_plane_seq
287     \__CJktu_write_sfd_map_header:n {#1}
288     \ior_str_map_inline:Nn \g__CJktu_sfd_ior
289     { \str_if_eq:nnT {##1} { 00 ~ 0x0000_0x00FF } { \ior_map_break: } }
290     \ior_str_map_inline:Nn \g__CJktu_sfd_ior
291     {
292         \tl_if_blank:nTF {##1}
293         {
294             \ior_map_break:n
295             {
296                 \__CJktu_write_sfd_map_trailer:n {#1}
297                 \__CJktu_write_cmap_file:n {#2}
298             }
299         }
300         { \__CJktu_read_sfd_line:nnn {##1} {#1} {#2} }
301     }
302     \iow_close:N \g__CJktu_sfd_cmap_iow
303     \ior_close:N \g__CJktu_sfd_ior
304     \seq_clear:N \l__CJktu_sfd_plane_seq
305     \seq_clear:N \l__CJktu_sfd_line_seq
306 }
307 {
308     \group_end:
309     \msg_critical:nxxx { xCJK2uni } { file-not-found } {#1} { #3.sfd }
310 }
311 }

```

(End definition for __CJktu_write_file:nnn.)

```

\__CJktu_read_sfd_line:n
\__CJktu_read_sfd_line:nn 312 \cs_new_protected_nopar:Npx \__CJktu_read_sfd_line:nnn #1#2#3
313 {
314     \exp_not:N \__CJktu_read_sfd_line:nwnn #1
315     \c_backslash_str \exp_not:N \q_nil \exp_not:N \q_stop {#2} {#3}
316 }
317 \use:x
318 {
319     \cs_new_protected_nopar:Npn \exp_not:N \__CJktu_read_sfd_line:nwnn
320     ##1 0 \token_to_str:N x ##2 \c_backslash_str ##3 \exp_not:N \q_stop
321     ##4##5
322     {
323         \exp_not:N \__CJktu_read_sfd_line:nnnnn
324         {##1} { 0 \token_to_str:N x } {##2} {##4} {##5}
325     }
326 }

```

(End definition for __CJktu_read_sfd_line:n and __CJktu_read_sfd_line:nn.)

```

\__CJktu_read_sfd_line:nnnnn
327 \cs_new_protected_nopar:Npn \__CJktu_read_sfd_line:nnnnn #1#2#3#4#5
328 {
329     \tl_if_blank:nF {#1}
330     {
331         \seq_if_empty:NF \l__CJktu_sfd_plane_seq
332         {
333             \iow_now:Nx \g__CJktu_sfd_map_iow
334             {
335                 \c__CJktu_indent_str \c__CJktu_indent_str
336                 \token_to_str:N \fi: \iow_newline:
337                 \c__CJktu_indent_str \c_right_brace_str
338             }
339             \__CJktu_write_cmap_file:n {#5}
340         }
341         \__CJktu_write_plane_map:nn {#1} {#4}
342         \tl_set:Nx \l__CJktu_sfd_plane_tl { \tl_trim_spaces:n {#1} }
343     }
344     \seq_set_split:Nnn \l__CJktu_sfd_line_seq {#2} {#3}
345     \seq_concat:NNN \l__CJktu_sfd_plane_seq
346     \l__CJktu_sfd_plane_seq \l__CJktu_sfd_line_seq

```

```

347 \iow_now:Nx \g__CJKtu_sfd_map_iow
348 {
349   \c__CJKtu_indent_str \c__CJKtu_indent_str \c__CJKtu_indent_str
350   \seq_use:Nn \l__CJKtu_sfd_line_seq { \or: }
351   \token_to_str:N \or:
352 }
353 }

```

(End definition for __CJKtu_read_sfd_line:nnnnn)

__CJKtu_write_plane_map:nn

```

354 \cs_new_protected_nopar:Npn \__CJKtu_write_plane_map:nn #1#2
355 {
356   \iow_now:Nx \g__CJKtu_sfd_map_iow
357   {
358     \cs_new_nopar:cpn { ~ __CJKtu_ #2 _sfd_map_ \int_eval:n {#1} :n ~ } ~
359     \c_hash_str 1 \iow_newline:
360     \c__CJKtu_indent_str \c_left_brace_str \iow_newline:
361     \c__CJKtu_indent_str \c__CJKtu_indent_str
362     \exp_not:N \if_case:w \etex_numexpr:D
363     \c_hash_str 1 ~ \token_to_str:N \scan_stop:
364   }
365 }

```

(End definition for __CJKtu_write_plane_map:nn)

__CJKtu_write_cmap_file:n

```

366 \cs_new_protected_nopar:Npn \__CJKtu_write_cmap_file:n #1
367 {
368   \tl_clear:N \l__CJKtu_cmap_block_tl
369   \int_zero:N \l__CJKtu_sfd_index_int
370   \iow_open:Nn \g__CJKtu_sfd_cmap_iow { c #1 \l__CJKtu_sfd_plane_tl .cmap }
371   \__CJKtu_write_cmap_header:nn { C #1 } { \l__CJKtu_sfd_plane_tl }
372   \iow_now:Nx \g__CJKtu_sfd_cmap_iow
373   {
374     \iow_newline:
375     1~begincodespacerange \iow_newline:
376     \c__CJKtu_indent_str < 00 > ~
377     <
378     \int_to_Hex:n
379     { \seq_count:N \l__CJKtu_sfd_plane_seq - \c_one }
380     > \iow_newline:
381     endcodespacerange
382   }
383   \seq_map_function:NN \l__CJKtu_sfd_plane_seq \__CJKtu_write_cmap_body:n
384   \tl_if_empty:NF \l__CJKtu_cmap_block_tl
385   {
386     \iow_now:Nx \g__CJKtu_sfd_cmap_iow
387     {
388       \iow_newline:
389       \int_mod:nn { \l__CJKtu_sfd_index_int } { \c_one_hundred } ~
390       beginbfchar \iow_newline:
391       \l__CJKtu_cmap_block_tl
392       endbfchar
393     }
394   }
395   \__CJKtu_write_cmap_trailer:
396   \seq_clear:N \l__CJKtu_sfd_plane_seq
397 }

```

(End definition for __CJKtu_write_cmap_file:n)

__CJKtu_write_cmap_body:n

```

398 \cs_new_protected_nopar:Npn \__CJKtu_write_cmap_body:n #1
399 {
400   \int_case:nnTF { \l__CJKtu_sfd_index_int }
401   { { 99 } { } { 199 } { } }
402   {
403     \iow_now:Nx \g__CJKtu_sfd_cmap_iow

```



```

404     {
405         \iow_newline:
406         100 ~ beginbfchar \iow_newline:
407         \l__CJKtu_cmap_block_tl
408         < \int_to_Hex:n { \l__CJKtu_sfd_index_int } > ~
409         <#1> \iow_newline:
410         endbfchar
411     }
412     \tl_clear:N \l__CJKtu_cmap_block_tl
413 }
414 {
415     \tl_put_right:Nx \l__CJKtu_cmap_block_tl
416     {
417         <
418         \int_compare:nNnF \l__CJKtu_sfd_index_int > \c_fifteen { 0 }
419         \int_to_Hex:n { \l__CJKtu_sfd_index_int }
420         > ~ <#1> \iow_newline:
421     }
422 }
423 \int_incr:N \l__CJKtu_sfd_index_int
424 }

```

(End definition for __CJKtu_write_cmap_body:n.)

__CJKtu_write_sfd_map_header:n

```

425 \group_begin:
426 \char_set_catcode_other:N \%
427 \cs_new_protected_nopar:Npn \__CJKtu_write_sfd_map_header:n #1
428 {
429     \iow_now:Nx \g__CJKtu_sfd_map_iow
430     {
431         %<#1> \iow_newline:
432         \cs_new_nopar:cpn { ~ CJKtu_#1_sfd_map:nn ~ } ~ \c_hash_str 1
433         \iow_newline: \c__CJKtu_indent_str \c_left_brace_str \iow_newline:
434         \c__CJKtu_indent_str \c__CJKtu_indent_str
435         \token_to_str:N \cs_if_exist_use:cF
436         \iow_newline: \c__CJKtu_indent_str \c__CJKtu_indent_str \c__CJKtu_indent_str
437         { ~ __CJKtu_#1_sfd_map_ ~ \exp_not:N \int_eval:n { \c_hash_str 1 } ~ :n ~ }
438         \iow_newline: \c__CJKtu_indent_str \c__CJKtu_indent_str \c__CJKtu_indent_str
439         { ~ \exp_not:N \use_none:n } \iow_newline:
440         \c__CJKtu_indent_str \c_right_brace_str
441     }
442 }

```

(End definition for __CJKtu_write_sfd_map_header:n.)

__CJKtu_write_sfd_map_trailer:n

```

443 \cs_new_protected_nopar:Npn \__CJKtu_write_sfd_map_trailer:n #1
444 {
445     \iow_now:Nx \g__CJKtu_sfd_map_iow
446     {
447         \c__CJKtu_indent_str \c__CJKtu_indent_str
448         \token_to_str:N \fi: \iow_newline:
449         \c__CJKtu_indent_str \c_right_brace_str \iow_newline:
450         %</#1>
451     }
452 }

```

(End definition for __CJKtu_write_sfd_map_trailer:n.)

__CJKtu_write_cmap_header:nn

```

453 \cs_new_protected_nopar:Npn \__CJKtu_write_cmap_header:nn #1#2
454 {
455     \iow_now:Nx \g__CJKtu_sfd_cmap_iow
456     {
457         %!PS-Adobe-3.0~Resource-CMap \iow_newline:
458         %%DocumentNeededResources:~ProcSet~(CIDInit) \iow_newline:
459         %%IncludeResource:~ProcSet~(CIDInit) \iow_newline:
460         %%BeginResource:~CMap~(TeX-#1-#2-0) \iow_newline:

```

```

461      %%Title:~(TeX-#1-#2-0~TeX~#1-#2~0) \iow_newline:
462      %%Version:~1.000 \iow_newline:
463      %%EndComments \iow_newline: \iow_newline:
464      /CIDInit~/ProcSet~findresource~begin \iow_newline: \iow_newline:
465      12~dict~begin \iow_newline: \iow_newline:
466      begincmap \iow_newline: \iow_newline:
467      /CIDSystemInfo~3~dict~dup~begin \iow_newline:
468      \c__CJKtu_indent_str /Registry~(TeX)~def \iow_newline:
469      \c__CJKtu_indent_str /Ordering~(#1-#2)~def \iow_newline:
470      \c__CJKtu_indent_str /Supplement~0~def \iow_newline:
471      end~def \iow_newline: \iow_newline:
472      /CMapName~/TeX-#1-#2-0~def \iow_newline:
473      /CMapVersion~1.000~def \iow_newline:
474      /CMapType~2~def \iow_newline: \iow_newline:
475      /WMode~0~def
476   }
477 }

```

(End definition for \c__CJKtu_write_cmap_header:nn.)

\c__CJKtu_write_cmap_trailer:

```

478 \cs_new_protected_nopar:Npx \c__CJKtu_write_cmap_trailer:
479 {
480   \iow_now:Nn \g__CJKtu_sfd_cmap_iow
481   {
482     \iow_newline:
483     endcmap \iow_newline:
484     CMapName~currentdict~/CMap~defineresource~pop \iow_newline:
485     end \iow_newline:
486     end \iow_newline: \iow_newline:
487     %%EndResource \iow_newline:
488     %%EOF
489   }
490 }
491 \group_end:

```

(End definition for \c__CJKtu_write_cmap_trailer:.)

```

492 \iow_open:Nn \g__CJKtu_sfd_map_iow { xCJK2uni-sfd.def }
493 \prop_map_inline:Nn \c__CJKtu_sfd_map_prop
494 { \c__CJKtu_write_file:nnn {#1} #2 }
495 \iow_close:N \g__CJKtu_sfd_map_iow
496 \cs_if_exist_use:NF \CJKtuend { \tex_end:D }
497 <make>

```

4 代码索引

斜体的数字表示对应项说明所在的页码,下划线的数字表示定义所在的代码行号,而直立的数字表示对应项使用时所在的行号。

Symbols	
\%	426
\\	5, 6, 7, 80, 229
A	
\AtBeginDocument	54, 201
B	
backslash commands:	
\c_backslash_str	315, 320
C	
catcode commands:	
\c_catcode_other_space_tl	264
\CCTchar	126
char commands:	
\char_set_catcode_active:n	105
\char_set_catcode_other:N	426
\CJKaddEncHook	51, 56
\CJKchar	125
\CJKchartouni	1, 1, 5, 212
\JKsfdtouni	1, 1, 5, 213
CJKtu commands:	
\CJKtu_bookmarks_hook:	3, 121, 199
\CJKtu_byte:NN	4, 110, 124, 131
\CJKtu_byte:nn	4, 126, 132, 134, 135, 136
__CJKtu_byte:Nwnw	119, 123
__CJKtu_byte:wnn	4, 125, 135
__CJKtu_byte:wNwnw	113, 117
\CJKtu_byte_map:nn	4, 118, 120, 141, 145, 158
__CJKtu_CCT_byte:NNN	4, 127, 133
__CJKtu_char_to_unicode:n	3, 106, 212
\l_CJKtu_cmap_block_tl	270, 368, 384, 391, 407, 412, 415
__CJKtu_enc_map:nn	2, 52, 57, 61, 75, 219
__CJKtu_encoding_error:	114, 146, 149, 151
\c_CJKtu_encoding_seq	20, 32, 81
\l_CJKtu_gap_begin_int	25, 65, 140, 166, 167
\l_CJKtu_gap_end_int	26, 66, 144, 167
__CJKtu_get_unicode:n	5, 128, 173, 180
__CJKtu_get_unicode:w	3, 94, 96, 181
__CJKtu_get_unicode_aux:w	100, 102
\l_CJKtu_i_max_int	22, 28, 138
\l_CJKtu_i_min_int	21, 63, 138, 164
\l_CJKtu_ii_max_int	24, 29, 39, 43, 49, 144
\l_CJKtu_ii_min_int	23, 64, 140, 165
\c_CJKtu_indent_str	263, 335, 337, 349, 360, 361, 376, 433, 434, 436, 438, 440, 447, 449, 468, 469, 470
__CJKtu_index_map:n	4, 160, 171
__CJKtu_initial:N	5, 214, 224
\g_CJKtu_path_str	233, 236, 251, 256, 259, 260, 261, 282
\l_CJKtu_plane_int	27, 67, 164
__CJKtu_read_sfd_line:n	7
__CJKtu_read_sfd_line:nn	7
__CJKtu_read_sfd_line:nnn	300, 312
__CJKtu_read_sfd_line:nnnn	7, 323, 327
__CJKtu_read_sfd_line:wnn	314, 319
__CJKtu_set_enc:nnnnn	2, 30, 85, 86, 87, 88, 89, 90
\g_CJKtu_sfd_cmap_iow	266, 302, 370, 372, 386, 403, 455, 480
\l_CJKtu_sfd_index_int	271, 369, 389, 400, 408, 418, 419, 423
\g_CJKtu_sfd_iior	232, 249, 251, 252, 283, 288, 290, 303
\l_CJKtu_sfd_line_seq	268, 305, 344, 346, 350
\CJKtu_sfd_map:nn	70, 94, 175
\g_CJKtu_sfd_map_iow	265, 333, 347, 356, 429, 445, 492, 495
\c_CJKtu_sfd_map_prop	6, 272, 273, 274, 275, 276, 277, 278, 493
\l_CJKtu_sfd_plane_seq	267, 286, 304, 331, 345, 346, 379, 383, 396
\l_CJKtu_sfd_plane_tl	269, 342, 370, 371
\c_CJKtu_sfd_prop	19, 33, 74, 218
\l_CJKtu_sfd_tl	18, 74, 75, 218, 219
__CJKtu_sfd_to_unicode:nn	3, 91, 213
__CJKtu_use_enc_map:n	2, 72, 211, 220, 222
__CJKtu_UTF_xvi_aux:n	183, 184
__CJKtu_UTF_xvi_be:n	5, 128, 182
__CJKtu_write_cmap_body:n	8, 383, 398
__CJKtu_write_cmap_file:n	8, 297, 339, 366
__CJKtu_write_cmap_header:nn	9, 371, 453
__CJKtu_write_cmap_trailer:	10, 395, 478
__CJKtu_write_file:nnn	6, 279, 494
__CJKtu_write_plane_map:nn	8, 341, 354
__CJKtu_write_sfd_map_header:n	9, 287, 427
__CJKtu_write_sfd_map_trailer:n	9, 296, 443
\CJKtuend	496
cs commands:	
\cs_generate_variant:Nn	84
\cs_gset_eq:NN	205
\cs_if_exist:cF	68
\cs_if_exist_use:cF	435
\cs_if_exist_use:Nf	496
\cs_if_exist_use:NT	56, 208
\cs_if_exist_use:NTF	51, 191
\cs_new_eq:NN	199, 211, 212, 213
\cs_new_nopar:cpn	358, 432
\cs_new_nopar:Npn	91, 96, 102, 106, 117, 119, 131, 133, 135, 136, 151, 158, 171, 180, 182, 184
\cs_new_protected_nopar:Npn	30, 61, 72, 121, 214, 279, 319, 327, 354, 366, 398, 427, 443, 453
\cs_new_protected_nopar:Npx	312, 478
\cs_set_eq:Nc	70
\cs_set_eq:NN	123, 124, 125, 126, 127, 128
E	
etex commands:	
\etex_numexpr:D	362
exp commands:	
\exp:w	93, 108
\exp_args:Nc	224
\exp_args:Nf	160, 183
\exp_end:	94, 96, 99, 103, 154, 181
\exp_last_unbraced:Nf	93, 181
\exp_not:N	314, 315, 319, 320, 323, 362, 437, 439

F		M	
fi commands:		msg commands:	
\fi:	336, 448	\msg_critical:nn	16
fifteen commands:		\msg_critical:nxxx	309
\c_fifteen	418	\msg_error:nn	11
file commands:		\msg_error:nxx	76
\file_input:n	69	\msg_expandable_error:nn	153
\file_path_include:n	282	\msg_fatal:nmnn	254, 257
G		\msg_new:nmn	3, 14, 78, 156, 193, 227
group commands:		\msg_warning:nn	197, 206
\group_begin:	69, 104, 281, 425	N	
\group_end:	69, 130, 285, 308, 491	nil commands:	
H		\q_nil	315
hash commands:		O	
\c_hash_str	359, 363, 432, 437	one commands:	
\hypersetup	191, 208	\c_one	39, 43, 46, 176, 379
I		\c_one_hundred	389
if commands:		or commands:	
\if_case:w	362	\or:	102, 103, 350, 351
int commands:		P	
\int_case:nnTF	400	\PassOptionsToPackage	192
\int_compare:nNnF	166, 418	\pdfstringdefPreHook	195, 200
\int_compare:nNnTF	109	prop commands:	
\int_compare:nTF	138, 140, 143	\prop_get:NnNTF	84
\int_const:cn	34, 35, 38, 40, 42, 46, 47, 48	\prop_get:NxNTF	74, 218
\int_div_truncate:nn	176, 187	\prop_gput:Nnn	33, 273, 274, 275, 276, 277, 278
\int_eval:n	162, 183, 358, 437	\prop_map_inline:Nn	493
\int_incr:N	423	\prop_new:N	19, 272
\int_mod:nn	177, 189, 389	R	
\int_new:N	21, 22, 23, 24, 25, 26, 27, 271	right commands:	
\int_set:Nn	28, 29	\c_right_brace_str	337, 440, 449
\int_set_eq:Nc	63, 64, 65, 66, 67	S	
\int_to_Hex:n	378, 408, 419	scan commands:	
\int_zero:N	369	\scan_stop:	205, 363
ior commands:		seq commands:	
\ior_close:N	252, 303	\seq_clear:N	286, 304, 305, 396
\ior_get_str:NN	251	\seq_concat:NNN	345
\ior_map_break:	289	\seq_count:N	379
\ior_map_break:n	294	\seq_if_empty:NF	331
\ior_new:N	232	\seq_map_function:NN	383
\ior_open:NnTF	249, 283	\seq_new:N	20, 267, 268
\ior_str_map_inline:Nn	288, 290	\seq_pop_right:NN	260
iow commands:		\seq_put_right:Nn	32
\iow_close:N	302, 495	\seq_set_split:Nnn	344
\iow_new:N	265, 266	\seq_set_split:NnV	259
\iow_newline: 336, 359, 360, 374, 375, 380, 388, 390, 405,		\seq_use:Nn	261, 350
406, 409, 420, 431, 433, 436, 438, 439, 448, 449, 457,		\seq_use:Nmnn	81
458, 459, 460, 461, 462, 463, 464, 465, 466, 467, 468,		stop commands:	
469, 470, 471, 472, 473, 474, 482, 483, 484, 485, 486, 487		\q_stop	100, 102, 315, 320
\iow_now:Nn	480	str commands:	
\iow_now:Nx ... 333, 347, 356, 372, 386, 403, 429, 445, 455		\str_const:Nx	263
\iow_open:Nn	370, 492	\str_gset:Nx	236, 261
L		\str_if_eq:nnT	289
left commands:		\str_new:N	233
\c_left_brace_str	360, 433	sys commands:	
lua commands:		\sys_if_engine luatex:TF	234
\lua_now_x:n	238	\sys_if_engine pdftex:F	12

T

TeX and L^AT_ε commands:

<code>\@CCTSetChar</code>	127
<code>\@ifpackagelater</code>	10
<code>\@ifpackageloaded</code>	196, 203
<code>\CJK@XX</code>	124
<code>\CJKtu@bookmarks@hook</code>	199, 200, 205
<code>\HyPsd@DecimalToOctalFirst</code>	186
<code>\HyPsd@DecimalToOctalSecond</code>	188

tex commands:

<code>\tex_end:D</code>	496
-------------------------------	-----

tl commands:

<code>\tl_clear:N</code>	368, 412
<code>\tl_count:n</code>	109
<code>\tl_gput_right:Nn</code>	200
<code>\tl_if_blank:nF</code>	329
<code>\tl_if_blank:nTF</code>	292
<code>\tl_if_blank:oTF</code>	256
<code>\tl_if_empty:NF</code>	384
<code>\tl_if_empty:nTF</code>	36, 98

<code>\tl_if_exist:NF</code>	195
<code>\tl_if_exist:NTF</code>	216
<code>\tl_if_head_eq_meaning:nNTF</code>	112
<code>\tl_new:N</code>	18, 195, 269, 270
<code>\tl_put_right:Nx</code>	415
<code>\tl_set:Nx</code>	342
<code>\tl_trim_spaces:n</code>	342

tmpa commands:

<code>\l_tmpa_seq</code>	259, 260, 261
--------------------------------	---------------

token commands:

<code>\token_to_str:N</code>	320, 324, 336, 351, 363, 435, 448
------------------------------------	-----------------------------------

two commands:

<code>\c_two</code>	49, 109
<code>\c_two_hundred_fifty_six</code>	176, 177, 187, 189

U

use commands:

<code>\use:c</code>	41
<code>\use:x</code>	317
<code>\use_none:n</code>	439
<code>\useCJKencmap</code>	1, 1, 5, 211