

字库编码与点阵字库生成工具

20250515

目 录

一. VGUS 串口屏字库编码设置	2
二. 两种格式字库文件比较 (DZK 与 ADZK)	4
三. 点阵字库生成工具使用说明	5
四. 制作 ASCII 码字库	6
五. 制作 GB2312 编码全角中文字库	7
六. 制作 Unicode 全字符字库	8
七. 制作 Unicode 自定义裁剪字库	9
八. 制作 Unicode 分区裁剪字库	11
九. Unicode 裁剪字库应用举例	12
十. Unicode 编码分区	14
十一. 常用字符编码方式	18
十二. 字库转换——DZK 转 ADZK	21



一. VGUS 串口屏字库编码设置

本节介绍 VGUS 串口屏中如何设置选择字库编码、半角/全角字符点阵、Y 方向点阵数等属性。

1.1 FONT0 字库设置

数据变量、数据录入、ASCII 录入、文本时钟、时钟变量、时钟设置控件，因为这些控件只用到数字和字母显示，所以这些控件只需要字库 FONT0。

必须正确设置字库相关属性，字库文件 FONT0 ID、Y 方向点阵数必须相互匹配，如图 1 所示，否则会导致字符不能正常显示。

变量属性	
名称定义	数据变量0
变量存储地址(0x)	0282
初始值	0
变量类型	int (2Byte)
显示格式	
整数位数	1
小数位数	0
文本颜色	255; 0; 0
FONT0 ID	0
X方向点阵数	16
Y方向点阵数	32
对齐方式	左对齐
无效位补零	<input type="checkbox"/>
小数点宽度调整	<input type="checkbox"/>
显示单位	
描述指针(0x)	FFFF

图 1 数据变量属性设置

变量属性	
名称定义	文本变量0
变量存储地址(0x)	01FC
初始值	
字长度	123
显示格式	
文本颜色	255; 0; 0
FONT0 ID	0
FONT1 ID	255
FONT1编码方式	0x05=UNICODE
Y方向点阵数	16
水平间隔	0
垂直间隔	0
字符间距自动调整	<input type="checkbox"/>
描述指针(0x)	FFFF

图 2 文本变量属性设置

FONT0 字库用于显示字符串的字母和数据，必须使用 ASCII 编码，不需要选择编码方式。

FONT0 字库点阵必须是半角字符，X 方向点阵数为 Y 方向点阵数的一半。

所有 VGUS 串口屏都预装有 0 号 ASCII 码字库，包含 4x8 到 64x128 点阵的所有字库。

1.2 FONT1 字库设置

文本变量、滚动文本、列表显示、文字标签、下拉列表、滚轮控件，这些控件既可以显示数字、字母，也可以显示中文汉字等全角字符，所以可以分别设置 FONT0 和 FONT1 字库。

必须正确设置字库相关属性，字库文件 FONT0/1 ID、字库编码和 Y 方向点阵数必须相互匹配，如图 2 所示，否则会导致字符不能正常显示。

FONT0 和 FONT1 所选字库的 Y 方向点阵数必须相同。

FONT0 字库用于显示 ASCII 码字符，同 2.1 节中介绍，必须是半角字符。

FONT1 字库用于显示全角字符，其 X 方向点阵数与 Y 方向点阵数相同。

FONT1 字库用于显示字符串中的全角字符，支持 GB2312、GBK、Big5、IJSI、Unicode、UTF8/UTF8-2 编码共 6 种编码方式，需要根据实际使用字库类型正确选择编码方式。

1.3 UTF8/UTF8-2 编码注意事项

当显示字符串使用 UTF8 编码时，字符编码值小于 0x80 使用 FONT0 字库，否则使用 FONT1 字库；当显示字符串使用 UTF8-2 编码时，无论字符编码是否小于 0x80，都使用 FONT1 字库。

当显示字符串使用 UTF8/UTF8-2 编码时，FONT1 字库必须使用 Unicode 字库文件。新版点阵字库生成工具支持 Unicode 字库裁剪，支持分区裁剪、以及提取指定字符点阵信息两种方式，能够有效解决 Unicode 字库文件过大、内存不足的问题。新版点阵字库还支持 Unicode 字库中的空格宽度设置。详细信息参考本文第七节。

1.4 常用字库文件获取

在 VGUS 开发工具的左侧字库文件列表中，点击鼠标右键、点击添加字库时，软件会打开常用字库文件夹，内置了部分常用宋体、黑体、以及数码管效果字库，可以根据需要选择添加。

当需要用到其它类型字库时，可以找技术支持索取、或参照本文后续章节自行生成。

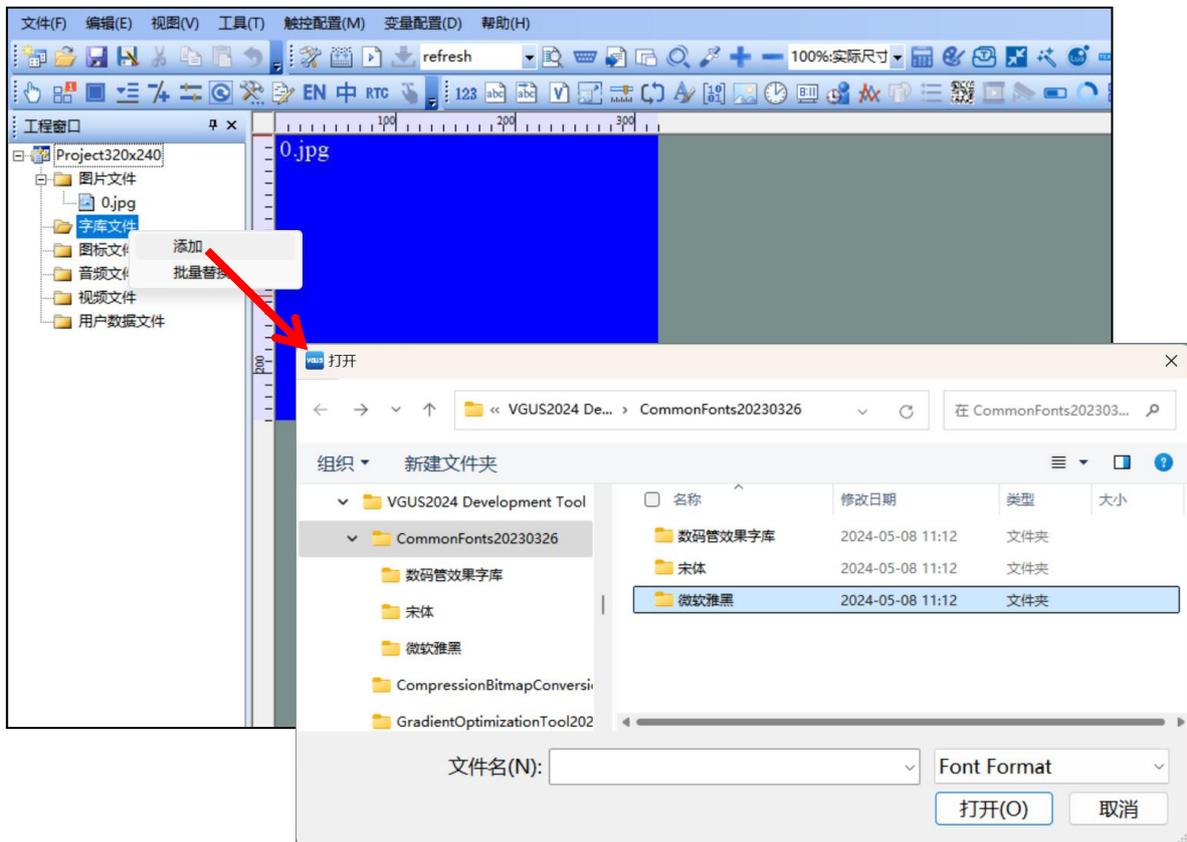


图 3 VGUS 开发工具中添加内置字库的方法

二. 两种格式字库文件比较 (DZK 与 ADZK)

VGUS 组态软件中的数据变量、文本变量、滚动文本、文本时钟、时钟显示、列表显示、文字标签、下拉列表、滚轮、数据录入、文本录入、RTC 录入等控件都有会使用到点阵字库。DZK 点阵字库采用一个像素对应一位二进制数的方式编码，该方式具有简单易用的特点。但是也有明显的不足：一是不支持抗锯齿；二是不支持文件压缩，导致字库文件大，占用较多内存。

为此，VGUS 组态软件创新采用了 ADZK 新字库文件格式，ADZK 字库文件从以下三个方面综合提升了应用性能：

- 1.支持生成抗锯齿字库，改善字符的显示效果。
- 2.支持字库文件压缩，提高存储效率，减少内存的占用。
- 3.支持定制裁剪版 Unicode 字库，减少内存的占用。

以前版本字库文件的后缀为“.DZK”，新版定义的字库文件后缀为“.ADZK”。

表 3: DZK 字库与 ADZK 字库比较

	支持的产品系列	字体抗锯齿	字库文件压缩	定制裁剪版 Unicode 字库
DZK	所有产品	不支持	不支持	不支持
ADZK	VTc、VTb、SDWb	支持	支持	支持

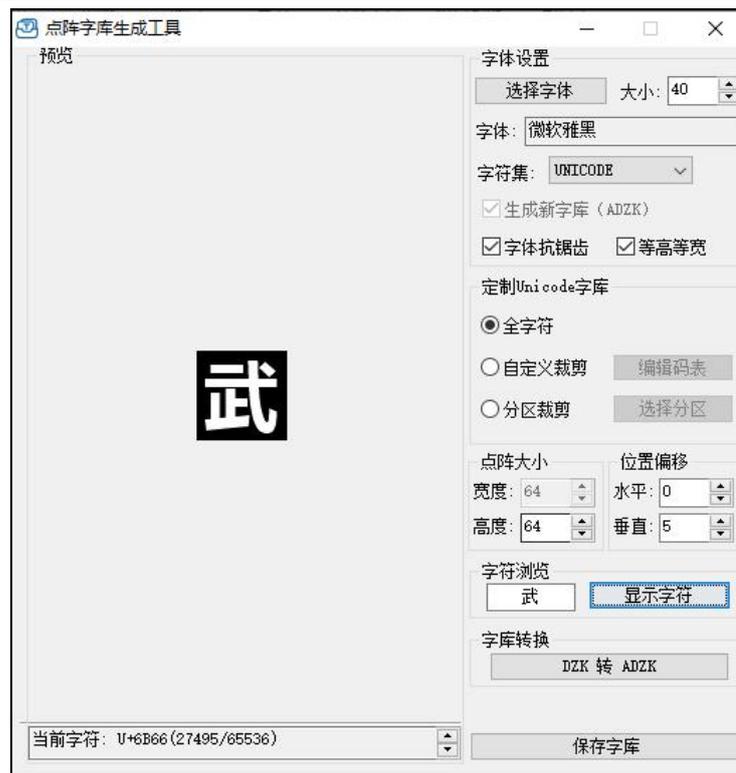


图 4 点阵字库生成工具

三. 点阵字库生成工具使用说明

在 VGUS2025 工具软件主界面下，在“工具”菜单栏下，选择“点阵字库生成工具”，可以打开点阵字库生成工具，主界面如图 4 所示。该工具既可以生成 ADZK 字库文件，也可以生成 DZK 字库文件，还可以将 DZK 字库文件转换为 ADZK 文件格式。

图 4 中各相关参数设置说明如表 4 所示。

表 4 各参数设置说明

序号	名称	说明
1	选择字体	选择需提取点阵字库的字体，如常见的宋体、黑体、楷体等等。
2	大小	设置点阵框中字的大小，点阵框中的字需要完整的显示在点阵框中央位置，并尽量充满点阵框。（字符浏览的时候选择宽度和高度都相对较大的字，例如英文字符“W”，汉字字符“武”）
3	字符集	选择所需字符集，即字库的编码方式。 可选 ASCII、BIG5、GB2312、GBK、SJIS、UNICODE。 对于仅需要部分字符的字库，用户可选用定制裁剪版的 Unicode 字库。
4	生成新字库 (ADZK)	勾选，将以 ADZK 格式保存字库数据，生成的字库文件的后缀为 ADZK。目前仅 VTc、VTb、SDWb 系列支持该格式字库。 不勾选，则会生成 DZK 格式字库。所有系列均支持 DZK 格式字库。
5	字体抗锯齿	在勾选“生成新字库 (ADZK)”的前提下，此选项才可以设置。 勾选则会生成字体抗锯齿字库，否则生成非抗锯齿点阵字库。 需要注意：勾选抗锯齿后，生成的字库文件会变大。
6	等高等宽	除 ASCII 码字符集，其他编码字库的字体都是等高等宽。
8	定制 Unicode 字库	字符集选择 UNICODE 后，可以定制裁剪版 Unicode 字库，此时会有三个选项： 1.全字符：基本上包含了世界上所有的语言字符，所以全字符 Unicode 字库文件非常大，字符编码为标准的 Unicode 编码。 2.自定义裁剪：可以只提取需要的字符，字符编码仍然是标准的 Unicode 编码。 3.分区裁剪：对不同种类的语言进行区分，可以只选择需要用到的语言分区，形成一个 Unicode 字库。字符编码是标准的 Unicode 编码。
9	点阵大小	宽度：字库 X 方向点阵数；高度：字库 Y 方向点阵数。
10	位置偏移	调整点阵框中字的水平位置偏移和垂直位置偏移，让预览的字符完全显示点阵框中央
11	字符浏览	如图，可预览字符“武”的效果。左下角显示当前预览字符的位置。点击上、下箭头，可以预览相邻字符的显示效果。
12	保存字库	点击即可开始生成字库。
13	字库转换	对于已有 DZK 格式的字库，想要在不改变字形的情况下，减小字库文件大小的用户可以尝试该功能，将 DZK 格式字库转换成 ADZK 格式的字库，详见下文（七）。

四. 制作 ASCII 码字库

制作 ASCII 码字库时点阵字库生成工具的设置如图 5 所示，先设置所需要的字体，字符集设置成 ASCII，生成新字库 ADZK 的选项根据需求选择是否勾选，目前只有 VTc、VTb、SDWb 系列的屏支持 ADZK 格式的字库。勾选“字体抗锯齿”可以生成显示效果更好的字库，但同时生成的字库文件会大一些。ASCII 码字库 X 方向点阵大小是 Y 方向点阵大小的一半，所以“等高等宽”一定不能勾选。

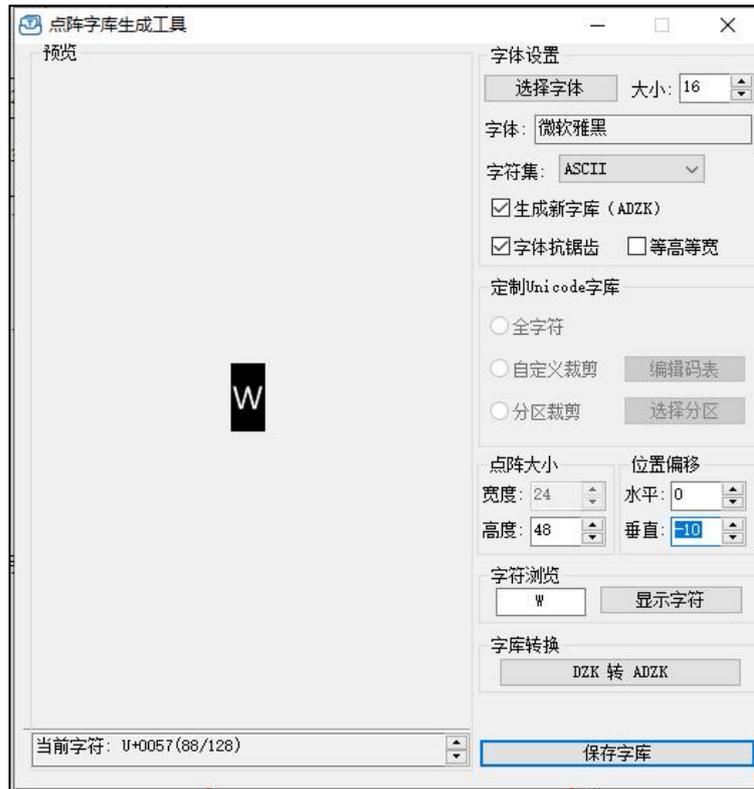


图 5 制作 ASCII 码字库点阵字库生成工具设置

例如点阵大小这里，高度（Y 方向大小）为 48，宽度（X 方向点阵大小自动设置为 24），字符浏览这里我们选择宽度和高度都比较大的字来预览避免出现有些字符显示不全的情况，这里选择大写的“W”进行预览。然后调整“水平和垂直位置偏移”让预览的文字处在点阵框中央。点击“保存字库”，即可生成一个 24x48 点阵的 ASCII 码字库。

五. 制作 GB2312 编码全角中文字库

制作 GB2312 编码全角中文字库时，点阵字库工具的设置如图 6 所示。除了 ASCII 码字库，其他编码字库都为全角字符，点阵大小的宽和高是一致的。

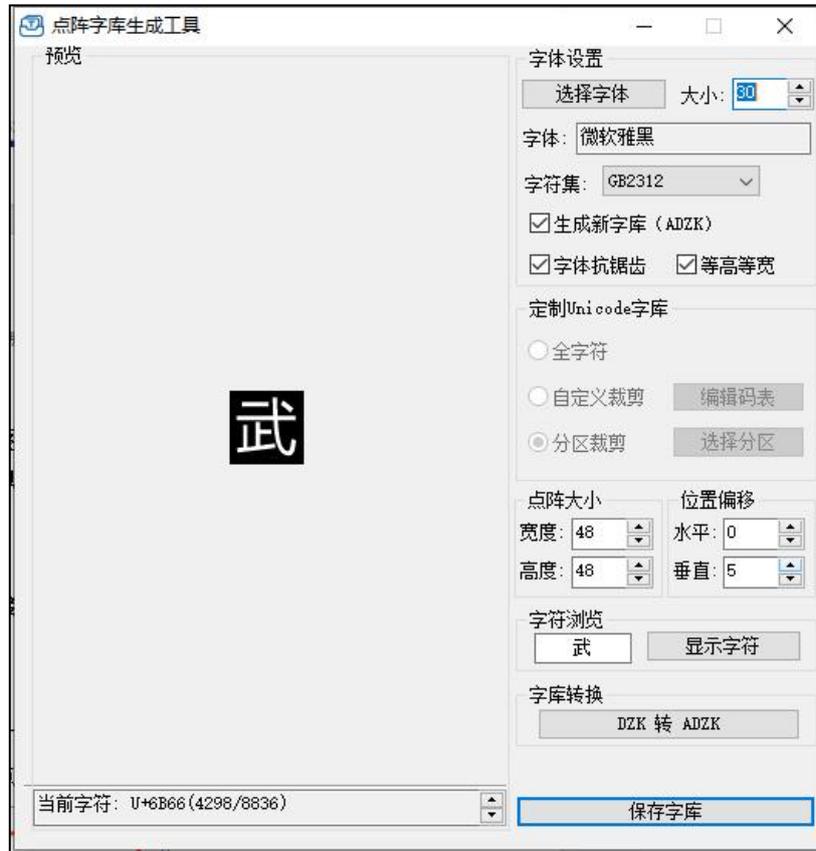


图 6 制作标准编码字库点阵工具设置

这里以 GB2312 编码为例，勾选“等高等宽”，点阵大小的宽度和高度都为 48，字符浏览这里填入汉字“武”，调整“大小”以及“水平垂直位置偏移”使字体刚好完整显示在点阵框中央，设置好后点击“保存字库”就生成了一个 48x48 点阵的汉字字库。

六. 制作 Unicode 全字符字库

制作 Unicode 全字符字库时，点阵字库生成工具的设置方式如图 7 所示，字符集选择 Unicode，在定制 Unicode 字库这里设置选择“全字符”，其他设置与上面制作 GB2312 编码字库设置一致，点击“保存字库”就可以生成一个 48x48 微软雅黑字体的 Unicode 编码全字符字库。



图 7.制作 Unicode 全字符字库点阵字库生成工具设置

按照这个设置最后生成的字库文件大小为 17.3M 字节，很容易就超出屏幕可用内存大小，所以一般建议使用自定义裁剪或者分区裁剪的 Unicode 字库。

七. 制作 Unicode 自定义裁剪字库

自定义裁剪 Unicode 字库，适用于只使用某些特定字符的场合。将实际使用到的字符点阵信息汇总到一起，形成一个容量超小的自定义字库。裁剪后的字符还是按照标准 Unicode 编码，用户继续使用标准 Unicode 编码引用这些字符。

制作自定义裁剪字库时，点阵字库生成工具的设置如图 8 所示，选中“自定义裁剪”，其他属性与制作 Unicode 全字符时一致，点击“编辑码表”，会出现一个自定义裁剪的弹出窗口，如图 9。

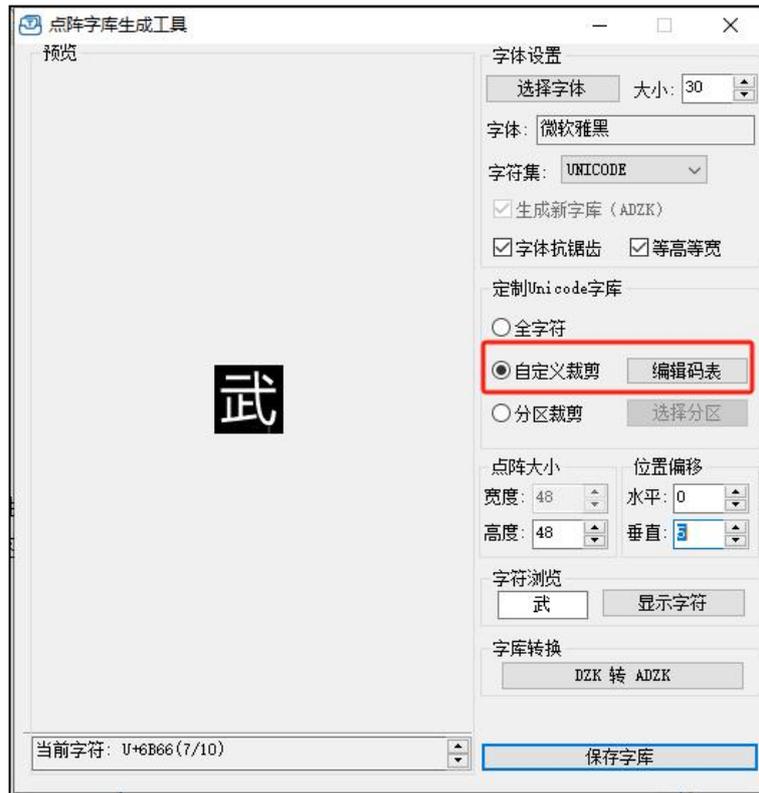


图 8.制作 Unicode 全字符字库点阵字库生成工具设置



图 9 自定义裁剪弹出窗口

将需要用到的字符全部列举在自定义码表的字符输入框里，点击“完成”后再点击保存字库即可生成一个自定义裁剪的 Unicode 编码字库。例如，在自定义码表里输入“武汉中显科技有限公司”，点击“完成”，图 9 是输入字符点击完成后再次打开编辑码表后的显示状态，字符会按照 Unicode 编码先后顺序自动排序，点击“保存字库”会生成一个包含“武汉中显科技有限公司”这 10 个汉字的自定义裁剪字库。按照该设置生成的文件大小只有 2KB，相比全字符的 17.3M 极大的减小了字库的大小。

除此之外，还可以选择自定义范围，我们可以在文档第十一节“Unicode 编码分布”中查询到韩文的编码范围为 0xAC00-0xD7AF，“开始字符”这里我们填入 0xAC00，“结束字符”这里我们填入 0xD7AF,然后点击“导入字符”，如图 10 所示，自定义码表里会生成这些编码对应的韩文字符。点击“完成”后生成的字库大小为 3.3M，也远远小于全字符的 17.3M。



图 10 自定义裁剪范围设置

编辑码表这里的载入文件这里就是可以导入 txt 文件，将 txt 文件上的字导入进编辑码表，然后再生成字库，这个方式和在编辑码表中直接输入字符，最后生成的字库实际上是相同的效果。

八. 制作 Unicode 分区裁剪字库

Unicode 字符按照语言分成了多个编码区间，如本文第十一节。分区裁剪可以生成特定语言的 Unicode 点阵字库。**裁剪后字库，用户继续使用标准 Unicode 编码引用这些字符。**

制作分区裁剪字库时，点阵字库生成工具的设置如图 11 所示。字符集选择 Unicode 后，在定制 Unicode 字库这里选择分区裁剪。

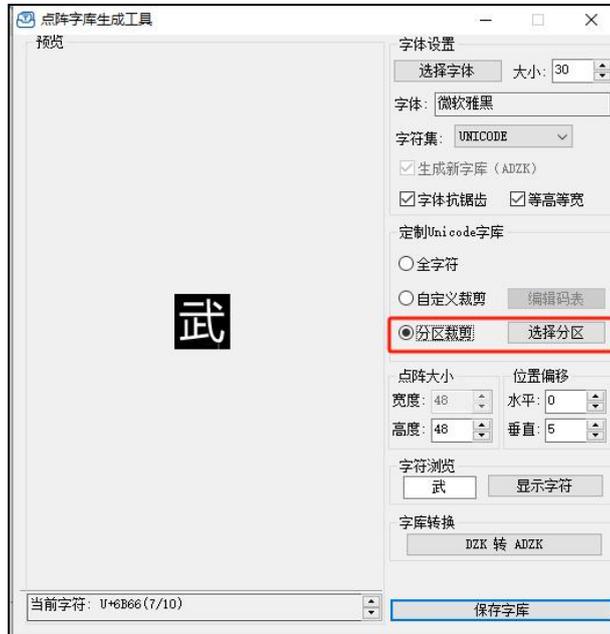


图 11.制作 Unicode 分区裁剪字库点阵字库生成工具设置

然后点击“选择分区”，会出现一个如图 12 所示的语言选择分区的弹出框，每个语言分区都会有一个复选框，可以选中不同范围，支持多选。

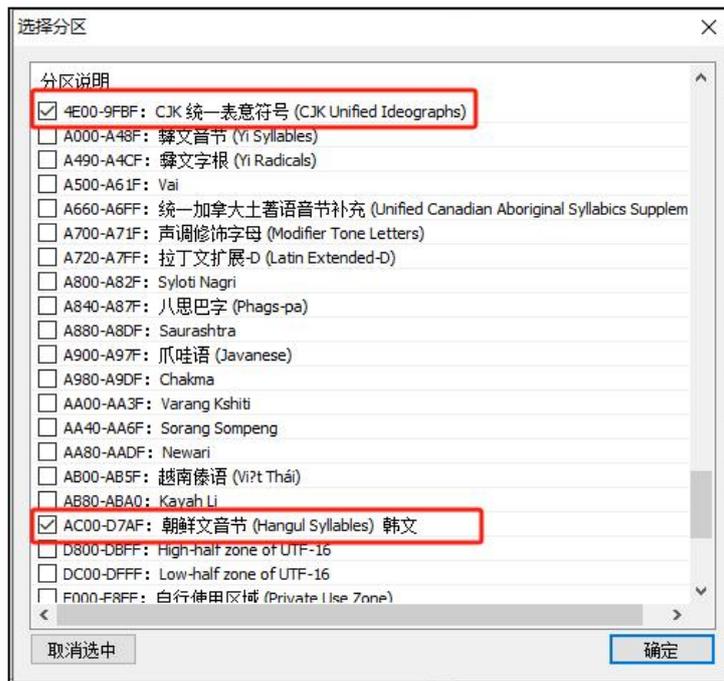


图 12. 语言分区弹出框

例如这里我们选择“4E00-9FBF: CJK 统一表意符号 (CJK Unified Ideographs)”再加上“A C00-D7AF: 朝鲜文音节 (Hangul Syllables) 韩文”。然后点击确定，再点击“保存字库”，就可以生成一个中文加上韩文的 48x48 点阵微软雅黑字体的 Unicode 编码的字库，其大小为 10M，比全字符的 17.3M 也缩小了很多。

九. Unicode 裁剪字库应用举例

如图 13 所示，在工程窗口里右键点击字库文件，将按照第八、九节设置生成的定制 Unicode 字库添加进来，“1_48x48 微软雅黑 Unicode 自定义裁剪字库”，包含了“武汉中显科技有限公司”十个汉字的 48x48 点阵的字库信息。“2_48x48 微软雅黑 Unicode 分区裁剪字库”包含了中文加韩文的 48x48 点阵的字库信息。

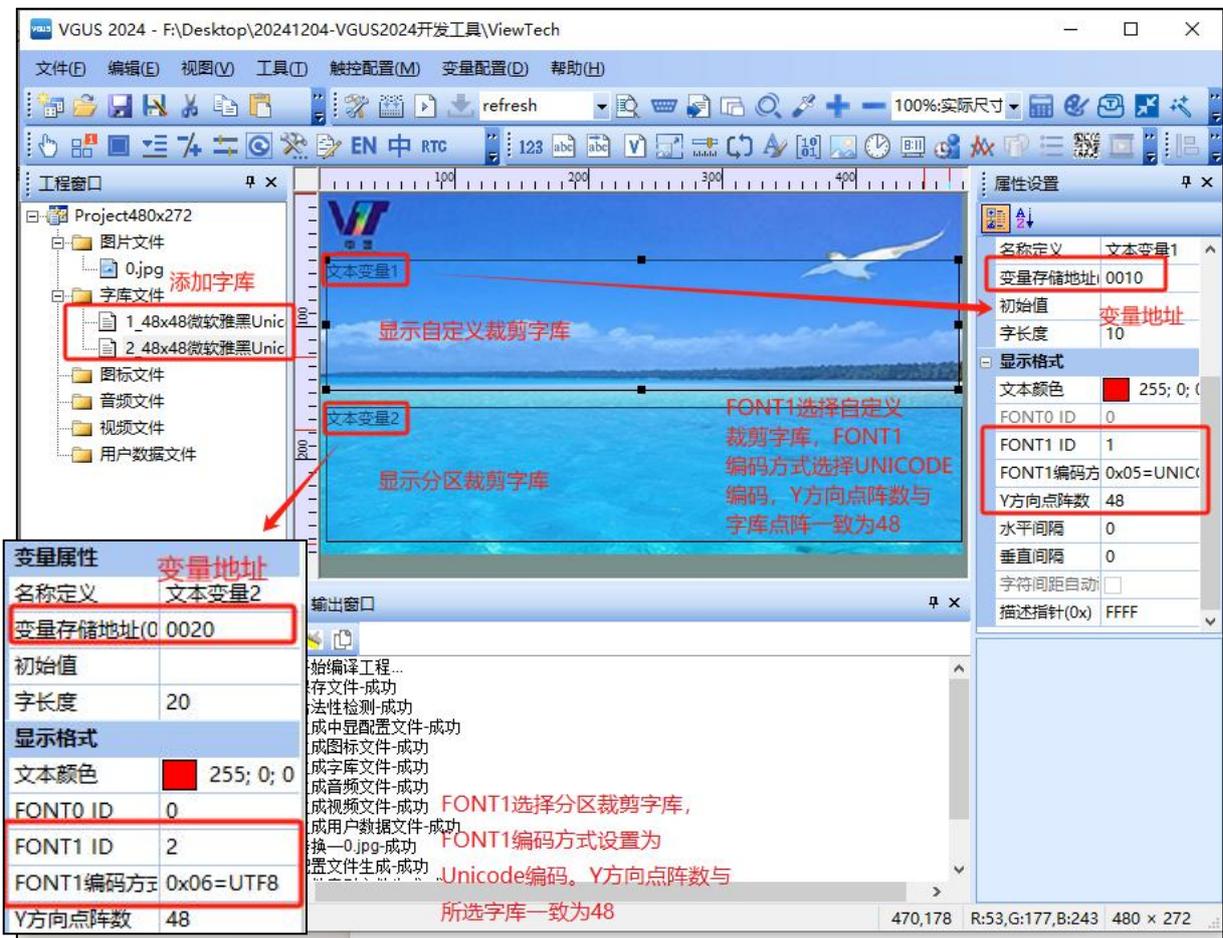


图 13. 定制 Unicode 字库对应控件设置

在 0 号图片上添加一个文本控件“文本变量 1”，用于测试自定义裁剪字库，属性设置如图 10 右侧所示。属性设置中的变量地址设置为 0x0010，编码方式为 UNICODE 编码，因此，当向串口屏中发送下面指令时，文本框位置将显示“武汉中显科技有限公司”。

A5 5A 19 82 00 10 6B 66 6C 49 4E 2D 66 3E 79 D1 62 80 67 09 96 50 51 6C 53 F8 FF FF

A5 5A: 帧头

19 : 指令长度 25 字节

00 10: 变量存储地址

6B 66 6C 49 4E 2D 66 3E 79 D1 62 80 67 09 96 50 51 6C 53 F8: Unicode 字符编码“武汉中显科技有限公司”

FF FF; 结束符

在 0 号图片添加控件“文本变量 2”，用于测试分区裁剪字库，属性设置与“文本变量 1”类似，但是编码方式用的是 UTF-8 编码，UTF-8 编码同样只能使用 Unicode 编码字库，如图 10 左下角所示。变量存储地址设置为 0x0020，UTF-8 编码方式每个韩文对应的是三个字节，所以字长度要大一些，这里设置为 20。

因此，当向串口屏中发送下面指令时，文本框位置将显示“武汉中显科技有限公司”对应的韩文字符“우한중현과학기술유한회사”。

A5 5A 29 82 00 20 EC 9A B0 ED 95 9C EC A4 91 ED 98 84 EA B3 BC ED 95 99 EA B8 B0 EC 88 A0 EC 9C A0 ED 95 9C ED 9A 8C EC 82 AC FF FF

A5 5A: 帧头

29 : 指令长度 41 字节

00 20: 变量存储地址

EC 9A B0 ED 95 9C EC A4 91 ED 98 84 EA B3 BC ED 95 99 EA B8 B0 EC 88 A0 EC 9C A0 ED 95 9C ED 9A 8C EC 82 AC :UTF-8 字符编码“우한중현과학기술유한회사”

FF FF; 结束符

通过指令助手，给实际屏发送上述两条指令后，实际屏显示效果如图 14 所示。



图 14 实际屏测试结果

十. Unicode 编码分区

注：中文范围 4E00-9FA5：CJK 统一表意符号 (CJK Unified Ideographs)

- 0000-007F: C0 控制符及基本拉丁文 (C0 Control and Basic Latin)
- 0080-00FF: C1 控制符及拉丁文补充-1 (C1 Control and Latin 1 Supplement)
- 0100-017F: 拉丁文扩展-A (Latin Extended-A)
- 0180-024F: 拉丁文扩展-B (Latin Extended-B)
- 0250-02AF: 国际音标扩展 (IPA Extensions)
- 02B0-02FF: 空白修饰字母 (Spacing Modifiers)
- 0300-036F: 结合用读音符号 (Combining Diacritics Marks)
- 0370-03FF: 希腊文及科普特文 (Greek and Coptic)
- 0400-04FF: 西里尔字母(Cyrillic) 俄文
- 0500-052F: 西里尔字母补充 (Cyrillic Supplement)
- 0530-058F: 亚美尼亚语 (Armenian)
- 0590-05FF: 希伯来文 (Hebrew)
- 0600-06FF: 阿拉伯文 (Arabic)
- 0700-074F: 叙利亚文 (Syriac)
- 0750-077F: 阿拉伯文补充 (Arabic Supplement)
- 0780-07BF: 马尔代夫语 (Thaana)
- 07C0-07FF: 西非书面语言 (N'Ko)
- 0800-085F: 阿维斯塔语及巴列维语(Avestan and Pahlavi)
- 0860-087F: Mandaic
- 0880-08AF: 撒马利亚语 (Samaritan)
- 0900-097F: 天城文书 (Devanagari)
- 0980-09FF: 孟加拉语 (Bengali)
- 0A00-0A7F: 锡克教文 (Gurmukhi)
- 0A80-0AFF: 古吉拉特文 (Gujarati)
- 0B00-0B7F: 奥里亚文 (Oriya)
- 0B80-0BFF: 泰米尔文 (Tamil)
- 0C00-0C7F: 泰卢固文 (Telugu)
- 0C80-0CFF: 卡纳达文 (Kannada)
- 0D00-0D7F: 德拉维族语 (Malayalam)
- 0D80-0DFF: 僧伽罗语 (Sinhala)
- 0E00-0E7F: 泰文 (Thai)
- 0E80-0EFF: 老挝文 (Lao)
- 0F00-0FFF: 藏文 (Tibetan)
- 1000-109F: 缅甸语 (Myanmar)
- 10A0-10FF: 格鲁吉亚语(Georgian)
- 1100-11FF: 朝鲜文 (Hangul Jamo)
- 1200-137F: 埃塞俄比亚语 (Ethiopic)
- 1380-139F: 埃塞俄比亚语补充 (Ethiopic Supplement)
- 13A0-13FF: 切罗基语 (Cherokee)
- 1400-167F: 统一加拿大土著语音节 (Unified Canadian Aboriginal Syllabics)
- 1680-169F: 欧甘字母 (Ogham)
- 16A0-16FF: 如尼文(Runic)



1700-171F: 塔加拉语 (Tagalog)
 1720-173F: Hanunóo
 1740-175F: Buhid
 1760-177F: 塔格班瓦文(Tagbanwa)
 1780-17FF: 高棉语 (Khmer)
 1800-18AF: 蒙古文 (Mongolian)
 18B0-18FF: Cham
 1900-194F: Limbu
 1950-197F: 德宏泰语 (Tai Le)
 1980-19DF: 新傣仂语 (New Tai Lue)
 19E0-19FF: 高棉语记号 (Kmer Symbols)
 1A00-1A1F: Buginese
 1A20-1A5F: Batak
 1A80-1AEF: Lanna
 1B00-1B7F: 巴厘语 (Balinese)
 1B80-1BB0: 巽他语 (Sundanese)
 1BC0-1BFF: Pahawh Hmong
 1C00-1C4F: 雷布查语(Lepcha)
 1C50-1C7F: 桑塔利文(OI Chiki)
 1C80-1CDF: 曼尼普尔语(Meithei/Manipuri)
 1D00-1D7F: 语音学扩展 (Phonetic Extensions)
 1D80-1DBF: 语音学扩展补充 (Phonetic Extensions Supplement)
 1DC0-1DFF: 结合用读音符号补充 (Combining Diacritics Marks Supplement)
 1E00-1EFF: 拉丁文扩充附加 (Latin Extended Additional)
 1F00-1FFF: 希腊语扩充 (Greek Extended)
 2000-206F: 常用标点(General Punctuation)
 2070-209F: 上标及下标 (Superscripts and Subscripts)
 20A0-20CF: 货币符号 (Currency Symbols)
 20D0-20FF: 组合用记号 (Combining Diacritics Marks for Symbols)
 2100-214F: 字母式符号 (Letterlike Symbols)
 2150-218F: 数字形式 (Number Form)
 2190-21FF: 箭头 (Arrows)
 2200-22FF: 数学运算符 (Mathematical Operator)
 2300-23FF: 杂项工业符号 (Miscellaneous Technical)
 2400-243F: 控制图片 (Control Pictures)
 2440-245F: 光学识别符 (Optical Character Recognition)
 2460-24FF: 封闭式字母数字 (Enclosed Alphanumerics)
 2500-257F: 制表符 (Box Drawing)
 2580-259F: 方块元素 (Block Element)
 25A0-25FF: 几何图形 (Geometric Shapes)
 2600-26FF: 杂项符号 (Miscellaneous Symbols)
 2700-27BF: 印刷符号 (Dingbats)
 27C0-27EF: 杂项数学符号-A (Miscellaneous Mathematical Symbols-A)
 27F0-27FF: 追加箭头-A (Supplemental Arrows-A)
 2800-28FF: 盲文点字模型 (Braille Patterns)



2900-297F: 追加箭头-B (Supplemental Arrows-B)
 2980-29FF: 杂项数学符号-B (Miscellaneous Mathematical Symbols-B)
 2A00-2AFF: 追加数学运算符 (Supplemental Mathematical Operator)
 2B00-2BFF: 杂项符号和箭头 (Miscellaneous Symbols and Arrows)
 2C00-2C5F: 格拉哥里字母(Glagolitic)
 2C60-2C7F: 拉丁文扩展-C (Latin Extended-C)
 2C80-2CFF: 古埃及语 (Coptic)
 2D00-2D2F: 格鲁吉亚语补充 (Georgian Supplement)
 2D30-2D7F: 提非纳文 (Tifinagh)
 2D80-2DDF: 埃塞俄比亚语扩展 (Ethiopic Extended)
 2E00-2E7F: 追加标点 (Supplemental Punctuation)
 2E80-2EFF: CJK 部首补充 (CJK Radicals Supplement)
 2F00-2FDF: 康熙字典部首 (Kangxi Radicals)
 2FF0-2FFF: 表意文字描述符 (Ideographic Description Characters)
 3000-303F: CJK 符号和标点 (CJK Symbols and Punctuation)
 3040-309F: 日文平假名 (Hiragana)
 30A0-30FF: 日文片假名 (Katakana)
 3100-312F: 注音字母 (Bopomofo)
 3130-318F: 朝鲜文兼容字母 (Hangul Compatibility Jamo)
 3190-319F: 象形字注释标志 (Kanbun)
 31A0-31BF: 注音字母扩展 (Bopomofo Extended)
 31C0-31EF: CJK 笔画 (CJK Strokes)
 31F0-31FF: 日文片假名语音扩展 (Katakana Phonetic Extensions)
 3200-32FF: 封闭式 CJK 文字和月份 (Enclosed CJK Letters and Months)
 3300-33FF: CJK 兼容 (CJK Compatibility)
 3400-4DBF: CJK 统一表意符号扩展 A (CJK Unified Ideographs Extension A)
 4DC0-4DFF: 易经六十四卦符号 (Yijing Hexagrams Symbols)
 4E00-9FBF: CJK 统一表意符号 (CJK Unified Ideographs)
 A000-A48F: 彝文音节 (Yi Syllables)
 A490-A4CF: 彝文字根 (Yi Radicals)
 A500-A61F: Vai
 A660-A6FF: 统一加拿大土著语音节补充 (Unified Canadian Aboriginal Syllabics Supplement)
 A700-A71F: 声调修饰字母 (Modifier Tone Letters)
 A720-A7FF: 拉丁文扩展-D (Latin Extended-D)
 A800-A82F: Syloti Nagri
 A840-A87F: 八思巴字 (Phags-pa)
 A880-A8DF: Saurashtra
 A900-A97F: 爪哇语 (Javanese)
 A980-A9DF: Chakma
 AA00-AA3F: Varang Kshiti
 AA40-AA6F: Sorang Sompeng
 AA80-AADF: Newari
 AB00-AB5F: 越南傣语 (Vít Thái)
 AB80-ABA0: Kayah Li
 AC00-D7AF: 朝鲜文音节 (Hangul Syllables) 韩文



D800-DBFF: High-half zone of UTF-16
DC00-DFFF: Low-half zone of UTF-16
E000-F8FF: 自行使用区域 (Private Use Zone)
F900-FAFF: CJK 兼容象形文字 (CJK Compatibility Ideographs)
FB00-FB4F: 字母表达形式 (Alphabetic Presentation Form)
FB50-FDFF: 阿拉伯表达形式 A ([Arabic Presentation Form-A](#))
FE00-FE0F: 变量选择符 (Variation Selector)
FE10-FE1F: 竖排形式 (Vertical Forms)
FE20-FE2F: 组合用半符号 (Combining Half Marks)
FE30-FE4F: CJK 兼容形式 (CJK Compatibility Forms)
FE50-FE6F: 小型变体形式 (Small Form Variants)
FE70-FEFF: 阿拉伯表达形式 B ([Arabic Presentation Form-B](#))
FF00-FFEF: 半型及全型形式 (Halfwidth and Fullwidth Form)
FFF0-FFFF: 特殊 (Specials)

十一. 常用字符编码方式

显示不同大小的字符、不同字体的字符、不同语言的字符，就需要选择不同的字库文件。所有字库文件都会涉及到编码方式问题，同一个字符在不同编码方式中编号是不同的。编码方式定义了如何按照一个字符编号从字库文件中找到该字符点阵数据存储位置的方法。下文将对常用字库编码方式进行介绍。

12.1 ASCII 编码

ASCII 码用于英文字母和阿拉伯数字显示，包括所有英文字符和阿拉伯数字等。ASCII 码用单字节数编码，共有 128 个字符或符号，所有字符采用顺序编号从 0 到 127。

ASCII 码是“American standard code for information interchange”的简称，原是美国标准，由于使用广泛，早已被确定为国际标准，成为计算机领域中最重要字符编码方式。

表 1 常用字符与 ASCII 代码对照表

编码	字符	编码	字符	编码	字符	编码	字符
0	NUL	32	Space	64	@	96	`
1	SOH	33	!	65	A	97	a
2	STX	34	"	66	B	98	b
3	ETX	35	#	67	C	99	c
4	EOT	36	\$	68	D	100	d
5	ENQ	37	%	69	E	101	e
6	ACK	38	&	70	F	102	f
7	BEL	39	'	71	G	103	g
8	BS	40	(72	H	104	h
9	TAB	41)	73	I	105	i
10	LF	42	*	74	J	106	j
11	VT	43	+	75	K	107	k
12	FF	44	,	76	L	108	l
13	CR	45	-	77	M	109	m
14	SO	46	.	78	N	110	n
15	SI	47	/	79	O	111	o
16	DLE	48	0	80	P	112	p
17	DC1	49	1	81	Q	113	q
18	DC2	50	2	82	R	114	r
19	DC3	51	3	83	S	115	s
20	DC4	52	4	84	T	116	t
21	NAK	53	5	85	U	117	u
22	SYN	54	6	86	V	118	v
23	ETB	55	7	87	W	119	w
24	CAN	56	8	88	X	120	x
25	EM	57	9	89	Y	121	y
26	SUB	58	:	90	Z	122	z
27	ESC	59	;	91	[123	{
28	FS	60	<	92	\	124	
29	GS	61	=	93]	125	}
30	RS	62	>	94	^	126	~
31	US	63	?	95	_	127	DEL

12.2 GB2312 中文编码（简体）

GB2312 编码用于简体中文显示，俗称为机内码，包含 6763 个中文简体汉字（一级汉字 3755 个，二级汉字 3008 个），也包含部分符号、字母、日文假名等共 7445 个图形字符。GB2312 编码采用双字节编号，高字节范围 0xa1—0xfe，低字节范围也是 0xa1—0xfe。GB2312 编码不是顺序编



码的，汉字编号从 0xb0a1 开始，结束于 0xf7fe，中间有大量空闲无用编号。

GB2312 编码是中华人民共和国国家汉字信息交换用编码，全称《信息交换用汉字编码字符集--基本集》，由国家标准总局发布，1981 年 5 月 1 日实施，通行于大陆，新加坡等地也使用此编码。中国大陆几乎所有的中文系统和国际化的软件都支持 GB 2312。

GB2312 的出现，基本满足了汉字的计算机处理需要，它所收录的汉字已经覆盖中国大陆 99.75% 的使用频率。对于人名、古汉语等方面出现的罕用字，GB2312 不能处理，这导致了后来 GBK 汉字字符集的出现。

表 2 GB2312 内码与国标码、区位码对照表

内码	A1		A2		A3		...		AA		AB		...		AF		B0		B1		...		F9		FA		FB		FC		FD		FE			
	国标		01		02		03		...		10		11		...		15		16		17		...		89		90		91		92		93		94	
	区位		01		02		03		...		10		11		...		15		16		17		...		89		90		91		92		93		94	
A1	21	01		、	。	...	—	~	...	“	”	...	*	→	←	↑	↓	≡																		
A2	22	02	i	ii	iii	...	x		1.	IX	X	XI	XII																			
A3	23	03	!	"	#	...	*	+	...	/	0	1	...	y	z	{		}	—																	
...	
A8	28	08	ā	á	ǎ	...	í	î	...	ó	ô	ū	...																							
A9	29	09				...	!	!	...	!	!	...	!	!	...	!	!	...																		
...	空区	
B0	30	16	啊	阿	埃	...	蔼	矮	...	隘	鞍	氨	...	谤	苞	胞	包	褒	剥																	
B1	31	17	薄	雹	保	...	豹	鲍	...	悲	卑	北	...	冰	柄	丙	秉	饼	炳																	
...
D6	56	54	帧	症	郑	...	知	肢	...	织	职	直	...	柱	助	蛀	贮	铸	筑																	
D7	57	55	住	注	祝	...	转	撰	...	庄	装	妆	...	座																						
D8	58	56	丁	汀	兀	...	鬲	霏	...	丿	匕	毛	...	伫	佞	侏	攸	佚	侑																	
D9	59	57	佟	佗	佻	...	侏	侑	...	侬	侔	侗	...	羸	羸	𠃉	沅	洌	洌																	
...
F6	76	86	鲛	鲧	鲧	...	霏	霏	...	霭	霭	霭	...	鳃	鳃	鳃	鳃	鳃	鳃																	
F7	77	87	鲞	鳍	鲟	...	蟹	蟹	...	鳢	鳢	鳢	...	鳔	鳔	鳔	鳔	鳔	鳔																	
...	空区
FE	7E	94	空区																																	

12.3 GBK 中文编码（简体+繁体）

GBK 编码用于中文罕见字、繁体字显示，共收录汉字 21003 个、符号 883 个，并提供 1894 个造字码位，简体、繁体中文融于一库。GBK 编码采用双字节编号，高字节范围 0x81~0xFE，低字节分两部分 0x40~0x7E 和 0x80~0xFE，其中和 GB2312 同的区域，字符完全相同。显然 GBK 编码和 GB2312 编码一样，也不是顺序编码的。

GBK 即汉字内码扩展规范，是对 GB2312 的扩展，K 为扩展的汉语拼音中“扩”字的声母，GBK 编码标准兼容 GB2312。

12.4 Unicode 通用字符编码（统一码、万国码、单一码）

Unicode 编码用于多国语言显示，包含了世界上所有的语言字符，它也就成为了全世界一种通用的编码，包含符号 6811 个、汉字 20902 个、韩文拼音 11172 个、造字区 6400 个、保留 20249 个，共计 65534 个。



Unicode 编码采用双字节进行编号，可编 65536 字符，基本上包含了世界上所有的语言字符，它也就成为了全世界一种通用的编码，而且用十六进制双字节表示一个编码，非常简结直观，为大多数开发者所接受。

Unicode 字库点阵信息统一为全角方式，点阵数据按照字符编码顺序排放。例如，一个 16x16 点阵的 Unicode 点阵字库文件，每个字符的点阵信息数据为 32 字节，编号为 n（编号范围 0~65535）的字符点阵数据在该字库文件中的偏移量为 32*n。任意一个字符，只要查询到该字符的 Unicode 编号，就可以找到该字符的点阵数据。

12.5 BIG-5 繁体中文编码

BIG-5 码用于繁体中文显示，是通行于台湾、香港地区的一个繁体字编码方案，俗称“大五码”。BIG5 每个字由两个字节组成，其第一字节编码范围为 0xA1~0xF9，第二字节编码范围为 0x40~0x7E 与 0xA1~0xFE，总计收入 13868 个字(包括 5401 个常用字、7652 个次常用字、7 个扩充字、以及 808 个各式符号)。BIG5 编码属于中文内码，其编码方式与 GB2312、GBK 编码方式类似。

BIG-5 码是使用繁体中文社群中最常用的电脑汉字字符集标准，共收录 13,060 个中文字，其中有二字为重覆编码。Big5 虽普及于中国的台湾、香港与澳门等繁体中文通行区，但长期以来并非当地的国家标准，而只是业界标准。

12.6 Shift-JIS 日文编码

Shift-JIS 编码（即 S-JIS 码）用于日文显示，是在 windows 系统中比较常用的一种日文编码。Shift-JIS 编码也是由两个字节组成。第一个字节是从 0x81-0x84、0x87-0x9F、0xE0-0xEA、0xED-0xEE、0xFA-0xFC，第二个字节是从 0x40-0xFC。

12.7 UTF-8 通用字符编码

UTF-8（8 位元，Universal Character Set/Unicode Transformation Format）是针对 Unicode 的一种可变长度字符编码。它可以用来表示 Unicode 标准中的任何字符，而且其编码中的第一个字节仍与 ASCII 相容，使得原来处理 ASCII 字符的软件无需或只进行少部分修改后，便可继续使用。因此，它逐渐成为电子邮件、网页及其他存储或传送文字的应用中，优先采用的编码。

UTF-8 使用 1~4 字节为每个字符编码：

- 一个 US-ASCII 字符只需 1 字节编码（Unicode 范围由 U+0000~U+007F）。
- 带有变音符号的拉丁文、希腊文、西里尔字母、亚美尼亚语、希伯来文、阿拉伯文、叙利亚文等字母则需要 2 字节编码（Unicode 范围由 U+0080~U+07FF）。
- 其他语言的字符（包括中日韩文字、东南亚文字、中东文字等）包含了大部分常用字，使用 3 字节编码。
- 其他极少使用的语言字符使用 4 字节编码。

十二. 字库转换——DZK 转 ADZK

DZK 转 ADZK 功能界面如图 15 所示。需要注意的是，要正确填写待转换的 DZK 字库的 X 方向点阵数和 Y 方向点阵数。选择 DZK 字库所在的路径，点击转换即可。

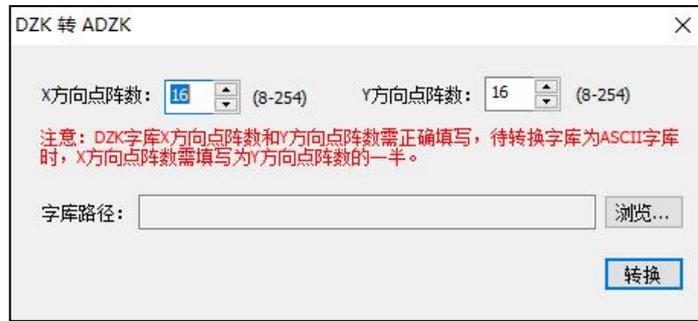


图 15 DZK 转 ADZK