在 RT1050 LittleVgl GUI 中嵌入中文输入法框架

Calvin Ji 2021.07.08

North China CAS Team

LittleVgl 作为一款开源免费的嵌入式 GUI 得到越来越多工程师的厚爱，我们可以看到很多小型 HMI 项目或者一些开源社区都在使用它作为 GUI 的框架，同时也受益于用户群的不断扩大以及一些半导体原厂的青睐（通俗点就是说有赞助有钱儿了），LittleVgl 本身也在快速的不断更新迭代，易用的组件和相关的辅助开发工具在不断的增加，而 RT1050/1060/1170 系列作为一款带有 LCD 控制器的平台，自然成为了 LittleVgl 最佳的载体之一了。

LittleVgl 本身的组件已经很丰富了，但是遗憾的是一直没有加入对中文输入法 Keyboard 的支持（看了下它在 Github 上的 Contributor List 没有华人），这让它在我们国内的应用有了一些限制（注意在某组件上显示中文和真正的中文输入法是不同的概念），所以本项目旨在解决该问题，即把一个简单轻量的中文输入法框架嵌入到 LittleVgl 并跑在 RT1050 平台上，并把它开源开放出来，所以不要小看了我的“公益心”，哈哈。

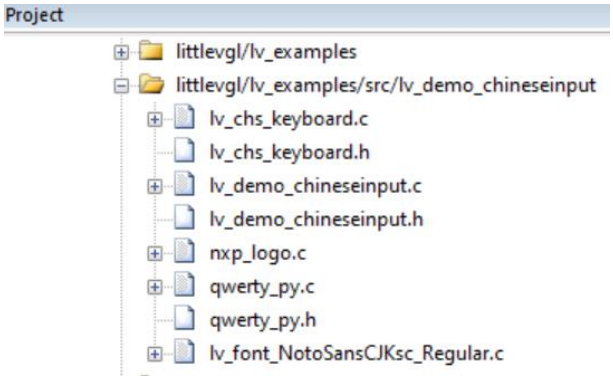
下面进入正题，首先把测试环境给出来，方便有兴趣有能力的朋友可以自行搭建（当然应一部分偷懒的强烈需求，我随本文档也附赠了完整的移植好的工程），然后我再一步一步地给出如何移植这套框架到用户自己的工程里，当然我已经把代码本身做了很多优化，尽量减小环境依赖，力求最少步骤的移植过程，理论上来讲不太会出现移植后编译出一堆 Error 的问题，咳咳。。。下面我们赶紧开整吧：

测试环境：

SDK 版本：SDK_v2.9.1
SDK 参考例程：boards\evkbimxrt1050\littlevgl_examples\littlevgl_demo_widgets
LittleVgl 版本：v7.4.0
IDE 工具：Keil_v5.31
开发板：MIMXRT1050-EVK + 480*272 RGB LCD 屏

软件说明：

我们先看下这套中文输入法所需的几个文件，如下图所示，.c 和.h 文件加起来一共 7 个，其中 nxp_logo.c 只是我额外加的一个 NXP 的官方 logo 图标转成的 C 数组文件供 littleVgl 调用显示，属于锦上添花的东西，可有可无，真正跟输入法相关的是剩下的 6 个文件，下面我们逐一介绍下这几个文件的作用：



1. qwerty_py.c/.h，实际上这两个文件才是这套全键盘拼音中文输入法的核心框架，实现了对输入的拼音字母进行索引匹配对应的汉字候选列表，这部分我是移植了如下链接中网友分享的代码，所以这两个文件我的角色只是一个大自然搬运工，不过说实话我是很感激该网友的无私分享的（这也是我一直推崇开源分享精神的源动力），之前对平时使用的各种输入法里面的算法原理一直充满好奇，直到看了这篇文章后才豁然开朗，“So that is what it is!”，让我获益匪浅（可能人的学习曲线和知识体系就是这样一点一滴的积累吧），

而且更关键的是，如果让我继续往下开发诸如拼音联想和多汉字输入等功能的话，我更多关心的可能只是逻辑搭建的工作量问题，而不是纠结于 Yes or No 的问题了，因为咱已经了解了其最底层的工作原理了，所以很多复杂的事情，我们如果能抽丝剥茧的找到其最底层的本质（虽然这真的很难），那很多让人抓耳挠腮的问题很快就可以理清思路。说到这里我思维又发散了，呵呵，我想起让 Linus Torvalds 等一波老大神们一直头疼的 Linux 内核维护后继无人的问题，其实我的个人理解有很大一部分原因是如今的 Linux 太庞大了以至于几乎没有后辈的人对 Linux 的理解能赶上这些老辈大神，而这些老辈大神的最大优势是他们创建了 Linux 最早期的底层框架而且难能可贵的是一直在 follow Linux 每个版本的历史。总之，推荐大家看看如下这篇文章吧（实际上主要内容也都是代码），希望能各有所获；
https://www.amobbs.com/thread-5668320-1-1.html?_dsign=0939dcbd

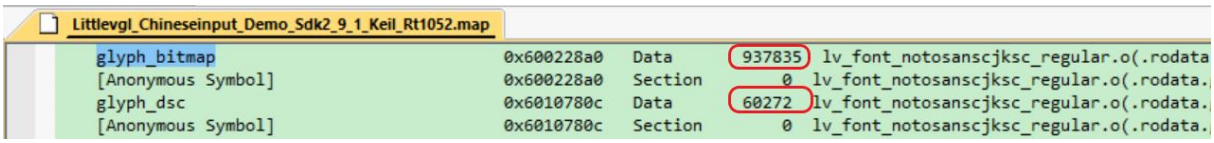
2. lv_chs_keyboard.c/.c 文件，这部分就是我的工作了（咱也不能啥都搬运…，这是体现咱的 value 的东西不是），我把它当作 littleVgl 的一个补充组件来写的，里面的大多数 API 参考官方 littlevgl 的 lv_keyboard.c，所谓的文章开头的嵌入中文输入法到 LittleVgl GUI 环境中实际上就是这两个文件干的活，即将上面提到 qwerty.c/.h 实现的拼音输入法与 LittleVgl 框架结合到一块，起到一个桥梁的作用，所以如果你想把这套中文输入法嵌入到其他 GUI 环境中的话（比如 emWin，GUIX，TouchGFX 等），那主要的工作就是参考这两个文件的内容了；

3. lv_font_NotoSansCJKsc_Regular.c 字体文件，虽然 littleVgl 官方源码包里自带了一个中文字体文件（\lvgl\src\lv_font\lv_font_simsun_16_cjk.c），但是它只包含了 1000 个左右最常用的字，我实际体验了下很多我们想用的字都找不到，所以这个时候就需要自己去做一个更全一点的字体库了。这里面涉及到两个问题需要考虑，第一是很多我们常见的中文字体是收费的（咱 PC 机的 Microsoft Office 套件里的中文字体都是微软付费买的，所以咱也理解下早年正版 Windows 为啥辣么贵了，那你问为啥现在便宜了？因为人家现在不靠这个赚钱了呗），第二个是字体转换工具的问题，我们网上找到的字体都是 TTF 或者 OTF 格式的，但 littleVgl 是不认的，需要转换成它支持的字体格式。

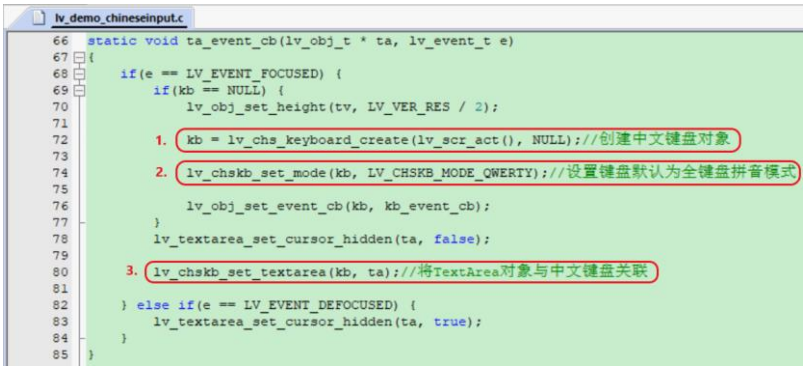
对于第一个问题，我网上搜了好久最终选择了目前用的比较多的 Google 开源免费的字体，Google 真乃金主也，它维护的网站里面字体各种各样啥都有且是开源免费的，如下链接，我选择的是 NotoSansCJKsc 字体（最后面的 sc 表示 simplified Chinese，简体中文），然后它里面又包含了各种字形（regular, bold, light 等），可以根据需要自行选择，整个包很大（100 多 MB），拆分成不同字形的就小了（每个 14~16MB 左右）；
https://www.google.com/get/noto/

对于第二个字体转换工具的问题，LittleVgl 官方自带了一个字体转换工具（online font converter）,我个人觉着不太好用（对 OTF 字体支持的不行），这里推荐阿里大神自己做的一个 LittleVgl 字体转换工具（LvglFontTool），非常方便好用，且支持加入 Awesome 图标；
http://www.lfly.xyz/forum.php?mod=viewthread&tid=24&extra=page%3D1

关于字体这部分我需要再补充个问题，就是它占用的 memory 大小，毕竟我们是在嵌入式 MCU 平台 Flash 和 RAM 的资源是受限的，如下图所示，该字体文件占用大概 1Mbytes 的 rodata 空间（即可寻址的 Flash 空间，当然该大小可以通过在上图转换工具中增减一些文字来调 整），所以在移植本套输入法之前需要预留足够的 Flash 空间，当然对 RT 平台来说这部分还好，毕竟其本身就外扩至少几 MB 空间的 QSPI Flash 作为存储空间的。



4. lv_demo_chineseinput.c/.h 文件，这两个文件属于应用层实现了，主要关注该文件中下图的 ta_event_cb 函数（即 textarea 事件的 callback，点击文本框的输入时回调），在里面我们需要按照 1，2，3 去调用即可（这三步的 API 均在 lv_chs_keyboard.c/h 文件里实现）；



至此，这套全键盘拼音中文输入法框架所需的几个文件就介绍完了，用户只需要把这几个文件放到自己的工程设置好文件搜索路径，并参考随本文档附带的代码工程示例，再结合自己产品的 GUI 样式，把这套中文输入法嵌入到自己应用当中。