

基于 SDL 库的嵌入式平台中文显示技术研究^①

Chinese Display Technology on Embedded Platform Using SDL

王园园 高明煜 曾 毓 (杭州电子科技大学 电子信息学院 浙江 杭州 310018)

摘 要: 随着国内基于 Windows CE 的嵌入式应用蓬勃发展,跨平台多媒体应用软件开发中涉及的中文显示技术尤其值得关注。该文介绍了利用 Simple DirectMedia Layer(SDL)图形库开发支持中文显示应用软件的方法和技巧,着重研究运用 SDL 显示 GBK 点阵汉字及 TrueType 矢量字体的方法,并比较各方法的优缺点,最后给出各自的适用场合。

关键词: SDL GBK 点阵汉字 TrueType 字体 嵌入式平台

1 引言

目前,嵌入式系统的字库还是以点阵字为主,随着嵌入式系统在 GPS 导航、手机、机顶盒等领域中的广泛应用,其处理器运算能力不断增强,越来越多的嵌入式设备采用了较复杂的 GUI 系统。由于用户对中文支持与文字显示速度的要求越来越高,当前低精度的点阵字虽然在显示速度较快,但难以满足高分辨率的显示要求,而高精度的点阵字库与普通的 TrueType 字库存在数据量大、显示速度慢等问题。因此,目前中文显示存在显示速度慢、分辨率低等多种问题。

由于国内更多的是将 SDL 应用于游戏编写、多媒体开发等,还未得到广泛应用于中文显示。同时,开发人员设计嵌入式平台软件时,必须了解不同操作系统的具体差别,从而影响了软件的开发周期。SDL 是一个自由的跨平台的多媒体开发包,使用 SDL 图形库编写的程序几乎可以不用改写就可以在 SDL 支持的操作系统上编译执行。而且 SDL 接口函数速度快、结构简单、移植性强等优点可以大大改善中文显示技术。因此,本文提出了在嵌入式平台上,利用 SDL 的图形库显示中文。通过测试多种显示汉字方法的速度比较,证明 SDL 在显示中文方面具有良好的可行性。

2 SDL

2.1 SDL 简介

SDL 是一个用 C 语言编写的免费跨平台多媒体开发库,使用 LGPL 许可证,可用于游戏、游戏开发工具、模拟器、样本演示、多媒体应用等。目前 SDL 的版本支持 Linux、Windows、BeOS、MacOS、MacOS X、FreeB-SD、BSD/ OS、Solaris 和 IRIX 等操作系统^[1]。对大多数情况来说,无论 DirectX、OpenGL 还是 SDL,都是低层的 API。SDL 与 DirectX 主要对应关系如表 1 所示。

表 1 SDL 与 DirectX 主要功能的对应关系

主要功能	SDL	DirectX
图像、视频	SDL_Video、 SDL_Image、OpenGL	DirectDraw、 Direct3D
声音	SDL_Audio、SDL_Mixer	DirectSound
输入支持	SDL_Joystick、SDL_Base	DirectInput
网络	SDL_Net	DirectPlay
多媒体	SDL_Video、 SDL_Audio、 SDL_Sound、SDL_Filter	DirectShow

相比较,选择 SDL 就意味着跨平台,减少了对操作系统的依赖。SDL 的 C 语言风格浓郁,程序简单易懂,封装性好,启动速度快,编程灵活性好,修改程序方便,可移植性强。

2.2 SDL 初始化

在 EVc(Embedded Visual C++)下,通过 AppW-izard 向导创建一个对话框之后,在 Project—>Set

① 基金项目:浙江省重大科技攻关项目(2005C11019)
收稿时间:2008-09-18

-ting 里将 SDL 的动态链接库和头文件加入^[2,3], 如: SDL.dll、SDL_ttf.dll、SDL.h、SDL_ttf.h 等。

在 SDL 初始化中, 先调用 SDL_ CreateRGBSurface-face(…)创建一个表面层。该函数可以设置所要创建表面层的高、宽、位色数等。初始化主要代码:

```
m_pMain=SDL_CreateRGBSurface(SDL_SWSURFACE, m_WINX, m_WINY, 32, rmask, gmask, bmask, amask);
```

该文使用的 flag 是 SDL_SWSURFACE, 它的作用是说明所建立的 surface 是储存在系统内存中的, 创建的是一个 32 位色的表面层, m_WINX、m_WINY 表示宽和高, rmask、gmask、bmask、amask 分别对应 RGB 三原色以及 α 值(透明色)。如果成功就返回一个 SDL_Surface 指针, 即表面层, 否则可以通过 SDL_GetError()获得错误信息; 再利用这个表面层的数据区 pixels, 用 CreateDIBSection(…)创建一个内存位图句柄。主要代码:

```
m_hMainBmp=CreateDIBSection(pDC->GetSafeHdc(), bmi, DIB_RGB_COLORS,(void **>(&m_pMain->pixels), NULL, 0)其中, pDC 是通过 GetCDC()获得窗口的 CDC 指针, bmi 是一个 BITMAPINFO 结构体; m_pMain->pixels 表示表面层的数据区。
```

3 利用SDL显示GBK点阵汉字

GBK 是一个汉字编码标准, 全称《汉字内码扩展规范》, 英文名称 Chinese Internal Code Specification, 总体编码范围为 8140-FEFE, 首字节在 81-FE 之间, 尾字节在 40-FE 之间, 剔除 xx7F 一条线^[4]。基本上采用了 GB2312-80 所有的汉字及码位, 并涵盖了原 Unicode 中所有的汉字 20902, 总共收录了 883 个符号, 21003 个汉字及提供了 1894 个造字码位, 简、繁体字融于一库。

显示 GBK 点阵汉字需要 GBK 字库文件。该文为了操作方便, 将 GBK 字库文件 HZK16.GBK 和 ASC16 文件分别转换成数组, 存放在 mygbk.h 文件中。在实际中可以将该字库文件直接放在内存中, 因为转换后的文件容量会增加, 对移动设备不适用。在程序中直接调用这个文件, 查出所需要的数据。该文已经在 2.2 节中提到 SDL 利用 SDL_ CreateRGBSurface(…)所创建的表面层内包含一个唯一数据区 pixel。SDL

显示文字就是利用该数据区, 在由 CreateDIBSection(…)创建的内存位图中“画”点阵汉字。

点阵汉字显示过程中的主要细节:

- ①如果是汉字逐个读取偏移量之后的 32 字节, 是 ASCII 码则只需读取 16 字节。
- ②对于所读入的每个字节, 要确定该字节每个位的值。如果为“1”, 在内存相应的位置显示, 否则不显示。
- ③如果是汉字且其内码大于 0xA0A0, 即为 GB 点阵, 可以按照 GB 点阵的计算公式:

qh=s-0x81;
wh=(s+1)-0x81;
offset=(94*qh+wh)*32 (1)

其中, s 为一个汉字首地址, qh 为区码, wh 为位码, 94 为一个区的汉字数。

- ④如果是汉字但不属于 GB 点阵汉字, 则根据该汉字内码所在 GBK 扩展的内码范围计算出其偏移量 offset。

该文采用更为简易的方法是自行制作一个根据内码顺序存储的点阵文件, 其对应关系见图 1。此方法虽然在程序开始要创建一个点阵文件, 需要消耗点时间, 但为以后的汉字显示争取了时间, 将原来计算偏移量的三个公式缩减为一个公式, 大大减少了显示每个汉字的时间。其点阵偏移量计算公式为:

offset=(内码值-0x8140) * 32 (2)

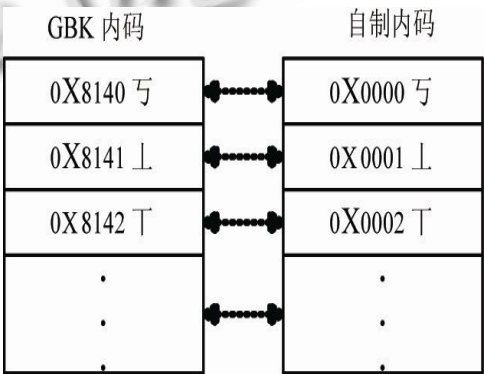


图 1 GBK 内码与自制内码的对应关系

在内存中将文字“画”完之后, 通过 SelectObject(…)将内存位图选入当前设备环境, 最后通过 CDC 的一个函数 BitBlt(…)将内存位图复制到屏幕。GBK 点阵显示主要流程如图 2 所示。

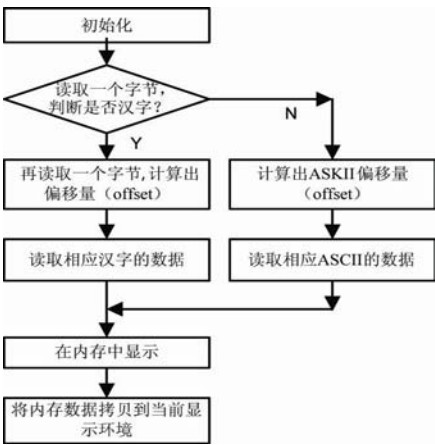


图 2 GBK 点阵单个文字显示流程图

4 利用SDL显示TrueType字体

TrueType 字体，中文名称全真字体，是由 Apple Computer 公司和 Microsoft 公司联合提出的一种新型数学字形描述技术，它用数学函数描述字体轮廓外形^[5]。TrueType 采用几何学中二次 B 样条曲线及直线来描述字体的外形轮廓，具有如下优势：①真正的所见即所得字体。②支持字体嵌入技术。③操作系统的兼容性。

由于利用 SDL 显示 TrueType 字体，必须使用 *.ttf 文件，从而需要扩展库 SDL_ttf 的支持。本文已在 2.2 节叙述添加动态链接库的方法。利用 SDL 显示 TrueType 字体主要步骤及函数：

- ①在 TTF 初始化中，用 TTF_Init() 装载扩展库。
- ②通过 TTF_OpenFont(const char *file, int ptsize) 打开字库，如果打开成功就会返回一个 TTF_Font 类型的指针，将该指针传递给 TTF_SetFontStyle(TTF_Font *font, int style) 函数，并通过该函数设置显示文字的字体。其中，file 表示所要打开的 ttf 文件，ttf_size 用来设置显示文字的大小，font 表示 TTF_OpenFont(…) 返回的指针。
- ③在内存中显示文字的时候要使用 TTF_RenderUNICODE_Solid(TTF_Font *font, const Uint16 *text, SDL_Color fg)。其中，text 表示要显示的内容，fg 设置显示文字颜色，因为 WinCE 是 Unicode 编码。如果该函数调用成功，则返回一个 SDL_Surface 类型的指针。
- ④通过 SDL_BlitSurface(SDL_Surface *src, SDL_Rect *srcrect, SDL_Surface *dst, SDL_Rect

*dstrect) 将③中返回的 SDL_Surface 块移到 2.2 节所创建的 m_pMain 主层面上，其中，src 表示源层面，dst 表示目的层面，srcrect 表示源层面的块移区域，dstrect 表示将块移到目的层面的区域。

TrueType 字体显示流程图如图 3 所示。

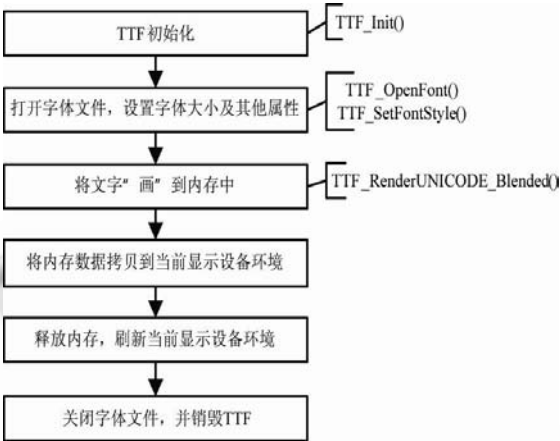


图 3 TrueType 字体显示流程图

5 利用Windows API函数显示中文

5.1 利用 DrawText 函数显示

在 Windows CE 下面很多 Windows 的 API 函数都被裁剪，如：文字显示 TextOut(…) 函数，取而代之的是 DrawText (…) 函数。该函数是将文字显示在特定的矩形框内。

在 EVC 平台建立一个 MFC 应用程序，将显示函数 DrawText (…) 放在 OnPaint () 子函数下，使其每次屏幕刷新时都调用该函数，同时修改 OnEraseBkgnd (…) 中的返回值，使其返回 TRUE，这样可避免出现文字抖动现象。

由于 WinCE 应用程序的一个重要特征是 Unicode 类型，因此利用 DrawText(…) 函数必须注意汉字的编码形式。对于所要显示的文字必须通过 MultiByteToWideChar(…) 将文字转换成宽字符类型。

5.2 利用 SetPixel 函数显示 GBK 点阵汉字

SetPixel 函数的主要功能是在特定的位置“画”点，利用该函数显示，首先要通过 CreateCompatibleBitmap(…) 和 CreateCompatibleDC(…) 分别创建一个内存位图和一个与当前设备环境相关的 dc，然后在该内存位图中“画”点。最后的内存位图显示及流程与 SDL 类似。

6 测试结果

为了力求测试数据的准确性，每一种方法都是显示相同数量的文字，而且对显示时间取多次测试数据的平均值，包括两种测试环境：①在 PC 机的 SDK 模拟器上，主频 1.61GHz，内存 512MB；②在 PXA270 开发板上，安装 Windows CE 4.20 的版本，主频最高达 500MHz，内存 32MB。测试数据如表 2 所示。

表 2 显示 150 个汉字平均时间

测试方法	平均显示时间(ms)	
	模拟器	PXA270
1 SetPixel 函数显示 GBK 点阵	2961	1289.8
2 DrawText 函数	18	117.8
3 利用 SDL 显示 GBK 点阵	84.8	312
4 利用 SDL 显示 TrueType 字体	1153	1637.2

从表 2 可以得知：①对于方法 1，模拟器上的显示速度远高于 PXA270，这是因为 SetPixel 函数在模拟器上没有优化过，该方法不适合显示文字数量较多的场合；另外由于有些手持式设备不支持 SetPixel 函数，则无法利用该 API 函数显示文字。②对于方法 2、3、4，模拟器上显示速度都比 PXA270 上快，这是因为在主频和内存等方面，PC 机的性能都比实际开发板强大；③方法 2 虽然显示速度最快，但只能将文字显示在特定的矩形框内，而且非中文平台上不能单独显示中文，适用范围有限；④方法 3 的显示速度适中，基本可以满足用户要求，但其显示的文字精度不高，在放大一定倍数后就会出现锯齿，即字体失真；⑤方

法 4 虽然不会出现任何失真情况，但其显示速度慢，而且每种 TrueType 字体文件较大，对于内存有限的嵌入式设备不适用。

7 结束语

该文介绍了基于 SDL 图形库开发支持中文显示应用软件的方法，不仅分析了各种 API 函数显示中文的功能，而且将这些 API 函数与 SDL 的显示方法做了详细比较和分析，其中，利用 SDL 显示 GBK 点阵的方法已经应用在本实验室的车载 GPS 导航仪电子阅读器和车胎压力检测显示中，经实际测试，证明利用 SDL 显示中文可行，性能良好。但在实际使用中，究竟采用哪种方法应根据具体情况确定。

参考文献

1 蒋鹏,刘玓.基于 SDL 库的跨平台多媒体应用.福建:福建电脑,2003,5:39-40.

2 Bob Pendleton.Game Programming with the Simple DirectMedia Layer. Specialized Systems Consultants, 2003,110:1.

3 Pazera E. Focus On SDL.USA: Course Technology PTR, 2003.

4 张晓培,李祥.从 Unicode 到 GBK 的内码转换.微计算机应用,2006,27(6):757-759.

5 孙晓辉,陈晓,王春.在嵌入式浏览器中使用 truetype 矢量字库.电视技术,2007,31(8):120-121.