# 打印机图标打印说明

## 打印原理

ESC K [打印点阵图形]格式：

ASCII： ESC K n1 n2 …data…

十进制： 27 75 n1 n2 …data…

十六进制： 1B 4B n1 n2 …data…

说明：

该命令打印n1×8点阵图形。该图形的宽度为n1点，高度为8点。每一列的8个点可以由一个8位的字节来表示，最高位在上。n1，n2的数值代表一个16位的二进制数，n1为低8位字节，n2为高8位字节，表示ESCK命令要打印的图形宽度为n2×256+n1。n2应是0或1，点数之和小于等于384，若超出则自动清除。n1应在1到该机型的每行最大点数之间。Data是该点阵图形从左到右每列的字节内容，字节数应等于n1。当图形的高度大于8点时，可按每8点行一个图形单元划分成多个单元。不足8点的用空点补齐。然后按顺序每个图形单元分别用ESCK命令打印出来，最后组成一个完整的图形。

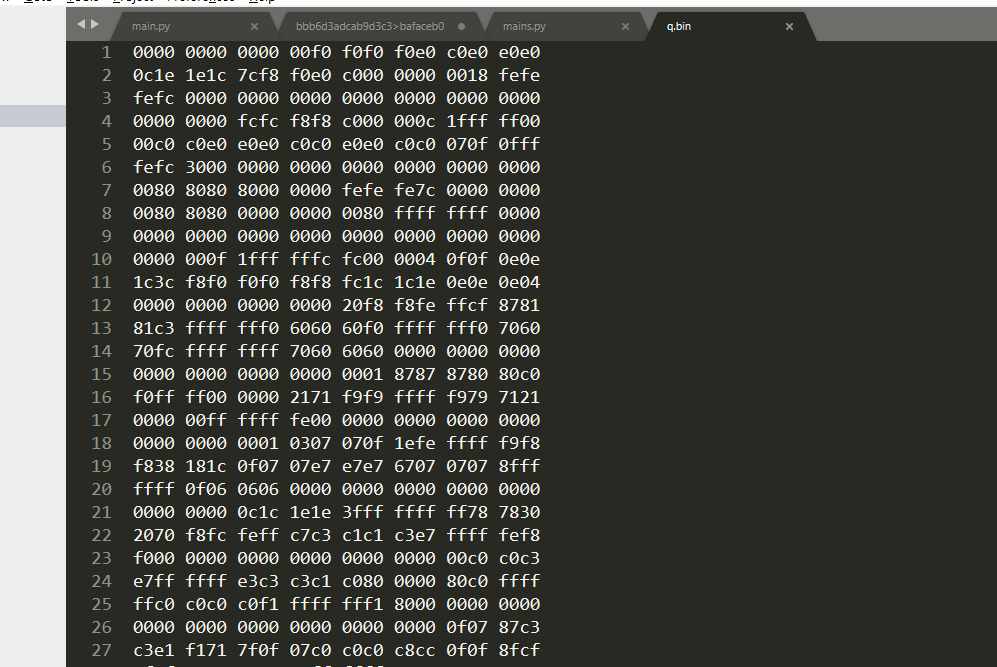
例如：用ESCK命令打印两个汉字“中文”这两个汉字的点阵如下图所示。每个字符由7×8点阵组成有7列。两字符之间有一空格，因此共有15列，于是n1=15，n2=0，15个字节的数据以十六进制表示如下：7CH，44H，44H，FFH，44H，44H，7CH，00H，41H，62H，54H，C8H，54H，62H，41H。



## 图片取模

使用Image2lcd这个软件对图片进行取模，取模的过程就是将图片格式如jpg,bmp等转换为点阵的形式，从而保存为16进制的数据，方便打印机的打印。方式如下：

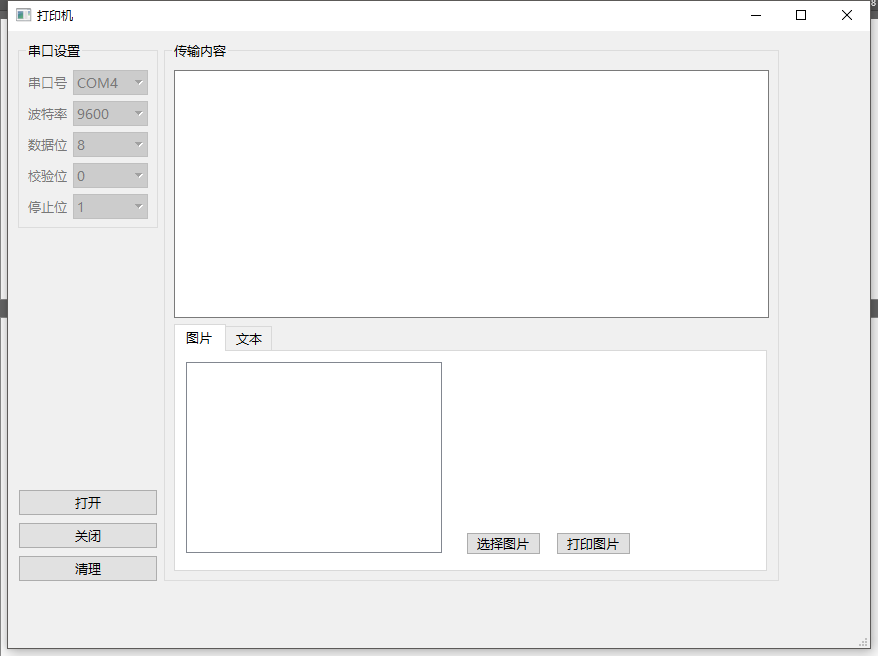


选择需要打印的图标，设置最大宽度和高度为180\*180，并勾选字节内像素数据反序和自底至顶扫描，保存为二进制文件(\*.bin)。打开此二进制文件可以看到：

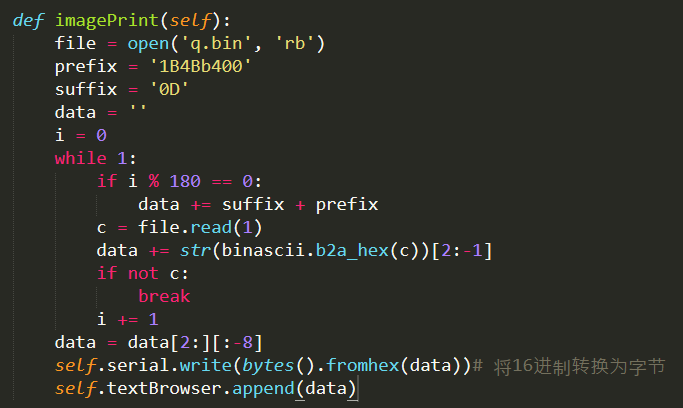
图片已经转换为16进制的形式，但是现在还不能将这些数据直接发送给打印机，还要添加相应的命令。

## 打印图标

为了对二进制文件进行处理和更形象地打印图标，我用PyQt5写了一个串口通信的简单客户端。如下：

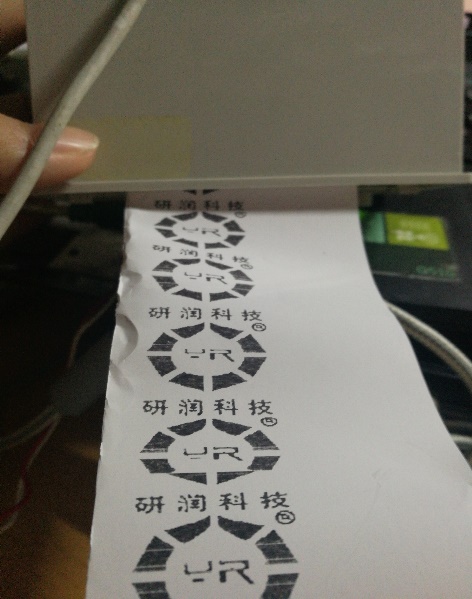
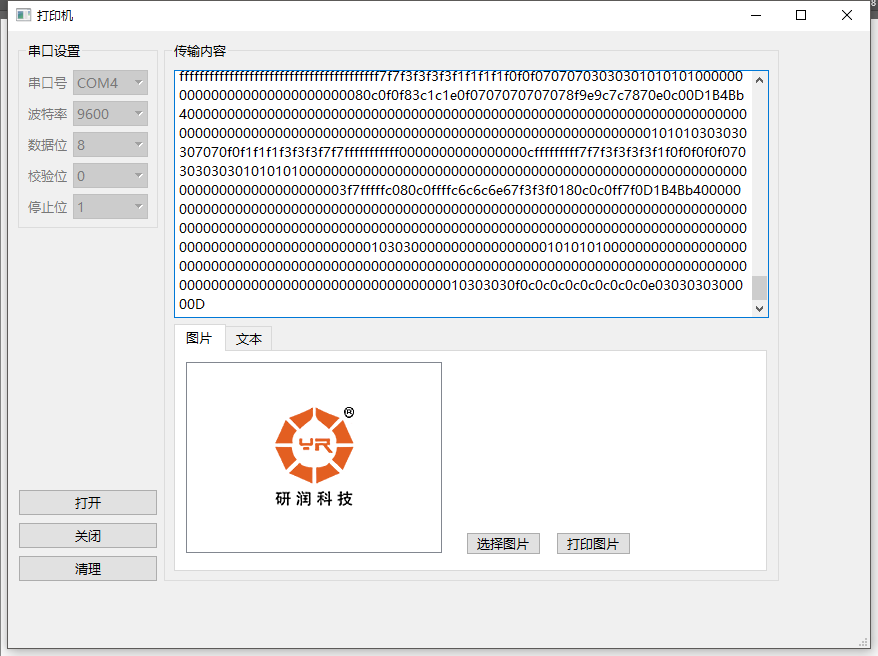


在此客户端中，将对数据的处理集成在一个函数imagePrint()里面：



前缀prefix=’1B4BB400’是执行打印图片命令：ESC K 180 0每行打印的宽度为180，等于图片的宽度。后缀suffix=’0D’是打印机的换行命令，打印一行后需要换行打印下一行。循环读取文件中的每一个字节，每次读过180个字节后，就加上换号命令和打印命令。这样就可以把打印命令加入要发送的数据中。但是这样的方式，在数据的前面会多一个suffix，而后面会多一个prefix，直接用Python切片的方式去除。

在客户端中“打开”正确的端口，点击“打印图片”，打印机开始打印图标。



## 遇到问题

当打印的图片或文本超过打印纸宽度的一半时会出现只打印部分的现象，可能是打印机出现了一点问题，去年使用的时候不是这样的。客户端中的“选择图片”和“打印图片”按钮功能之间没有关系，我想实现一个不同外部软件image2lcd直接对图片取模功能，但是这个有点实现不了。