

智能称重仪表连接针式打印机的开发

□上海耀华称重系统有限公司 傅啸雷

【摘要】传统称重仪表受制于硬件性能，连接针式打印机时只能打印固定模板格式的数据。本文针对智能称重仪表的开发，探讨一种灵活易用的使用方式以及所见即所得的打印模板设计方式。

【关键词】智能；称重；仪表；针式打印机

文献标识码：B 文章编号：1003-1870（2024）04-0047-04

引言

在称重仪表的使用过程中，打印机属于使用频率较高的外部设备之一。随着科技的发展，打印机的类型得到了极大的扩充。除了以前常见的针式打印机、热敏打印机之外，最近喷墨打印机和激光打印机也有了较为广泛的应用。而针式打印机可以使用方便裁切的打孔纸，能够打印多联单，在汽车衡应用场景下依然有着较高的使用率。

然而，目前市场上大部分的称重仪表在连接针式打印机时存在着打印模板设计复杂，仅能打印字符和数字，表格框线限制较多等缺点，不能很好地满足客户的需求。

相对传统仪表，智能称重仪表的性能得到了极大的提升，使解决上述问题成为了可能。

1 打印方案

传统称重仪表的打印方案也经历了多次发展，在模板的设计和功能上都有了一定的改进。

1.1 定制程序模板

最初的方案是：打印模板直接固化在程序中，打印时，仪表把程序中打印模板的占位符用实际数据替换，再向打印机输出制表符和字符。如果要修改成客户自己的模板，则需要找仪表开发人员进行程序定制，开发流程周期长且模板格式呆板，如图1所示。

称重单

序号	0001
日期	1999-07-28
时间	12.02.31
车号	12345
货号	022
毛重	2.000(kg)
皮重	0.300(kg)
净重	1.700(kg)

图1 定制程序打印内容示意

1.2 打印项可选模板

在原有方案的基础上，添加了打印项可选的功能。用户可以选择，哪些项需要打印，哪些不打印，这样就可以打印更丰富、对客户更有用的信息了。可以通过串口、网络或屏幕输入的方式，把打印模板设置到仪表中。这种方式解决了原先定制周期长的缺点，但是模板格式依然呆板，可修改内容少。

1.3 自定义打印模板

这个方案可以说是所见即所得方案的雏形。用户通过上位机软件，把内容元件拖放到设计界面中。可以设计表格线的位置，打印文字的内容、定位、占位长度等。同时在设计的过程中预览框中会显示最终打印的效果，使模板的设计更加直观。但这种方案依然不能打印图片、斜线表格等，设计限制较多，如图2所示。

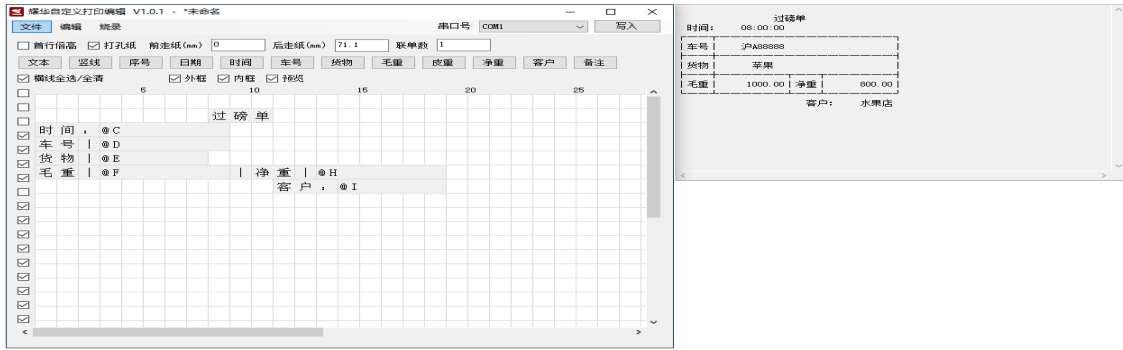


图2 自定义打印编辑上位机

1.4 智能设计模板

此方案就是本文探讨的方案。用户通过电脑表格软件，可以设计任意样式的表格，打印丰富的仪表相关数据、文字及图片，并且整个过程操作灵活简便。另外，传统的方案是将制表符、数据等按一定格式直接发给打印机，存在安装作弊装置、窃听并篡改打印数据的可能性。此方案把整个打印内容作为图形传输给打印机，没有明码数据，窃听及篡改难度大。

2 方案设计

2.1 模板设计方案

称重仪表的用户大部分不会具有设计专业知识，因此，需要在设计时尽量贴近用户操作习惯，最好是平时在操作电脑时经常会使用到的操作。本方案选择使用Excel或WPS表格软件进行设计，这两款软件使用范围广泛，只需要会使用电脑就可以很快上手，并且可以制作多样精美的图表，如图3所示。

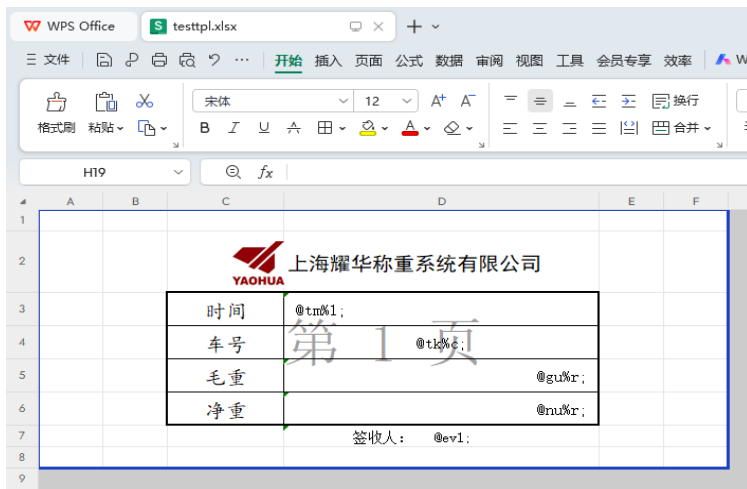


图3 使用WPS表格进行模板编辑

2.2 模板处理方案

打印模板一般分为固定部分和可变部分。通常来说，表格头、表格线、公司标识等都是固定部分。而时间、重量、车号等是可变部分，在模板中用占位符表示。打印模板传入仪表后，仪表在打印前把占位符用实际数据替代。PDF格式的文件在各种设备上显示都能保持一致，并且能在程序中查找占位符精确位置、插入文字，因此上述软件在编辑完成后导出PDF文件再输入仪表中。

2.3 打印指令方案

通常使用针式打印机时，将打印的制表符、文字、数字等，通过指令发送给打印机。一般使用ESC/P系列指令，市面上绝大多数的针式打印机都兼容这套指令。用这种方式打印的优点是，处理速度快，缺点是，不同型号打印机打印出的字体、样式会有一些差别，没法做到和设计完全一致，并且无法打印图片和复杂的表格。本方案通过使用图像打印方式，损失一些处理速度，但是可以把模板原本的样子展现出来，达到所见即所得的效果，如图4所示。

 上海耀华称重系统有限公司 YAOHUA	
时间	2024-02-26 11:25:37
车号	沪A88888
毛重	5000.00kg
净重	3190.00kg

签收人： 耀华

图4 最终打印效果

3 软件开发

3.1 模板格式设计

打印模板的固定部分可以按照用户需求自行设计。为了在可变部分打印尽可能多的仪表数据，并能选择打印对齐的位置，本方案设置了一系列占位符供用户使用。占位符格式如下：

@ 打印项% 对齐方式;

其中，占位符以“@”开头，“;”结尾。“% 对齐方式”为可选项，默认为左对齐。对齐方式如表1所示。

表1 占位符对齐方式

对齐方式	占位符
左对齐	l
居中对齐	c
右对齐	r

由于打印项的项目繁多，在此选取一部分举例，如表2所示。

表2 占位符打印项

打印项	占位符
时间	tm
毛重带单位	gu
净重带单位	nu
扩展字段1	ev1

因此，按规则，如果需要打印净重带单位，右对齐显示，则可以写占位符：

@nu%r;

通过这种方式，可以任意设计各种类型和格式的模板。在实际应用中，也可以把表格编辑功能嵌入到上位机中，通过快捷按钮或菜单，编辑插入各项占位符，使用更加方便。

设计完成后，可以通过U盘、网络等方式把打印模板保存到仪表中，并可以在仪表中进行选择，随时切换。

3.2 占位符处理设计

打印模板在打印之前需要先将模板中的占位符替换成仪表中的实际数据，这需要在仪表中根据实际情况编辑PDF文件。

目前，可以编辑PDF文件的Java库有很多，各种协议的开源、商业库都有。需要注意的是，有些开源协议是具有传染性的，也就是说使用了它提供的开源代码后，写出的程序也必须开源，不能随意使用在闭源系统中。本方案选择Apache基金会的开源库PDFBox，目前最新的版本是3.0.1。该库采用的是Apache2.0协议，对闭源软件较为友好，使用时只需要注明库引用及提供Apache协议即可。同时该库经过长期多个版本演变，可靠性还是可以保证的。当然缺点也是显而易见的，就是使用支持很少，没有完整的说明或教程，能搜索到的有用资料也不多，在使用过程中很多都需要自己摸索。

要替换占位符，需要找到占位符在文中的位置，字的大小、长度和内容。再用白色区域将这一块覆盖，最后把要显示的数值按格式插入到占位符的位置上。

3.3 模板文档加载

首先，需要加载打印模板和打印字体才能进行下一步操作。

// 加载打印模板PDF文档

```
PDDocument doc = Loader.loadPDF(new RandomAccessReadBufferedFile(src));
```

// 加载打印字体

```
PDType0Font font = PDType0Font.load(document, trurTypeFont, true);
```

3.4 模板文字获取

然后，查找占位符，PDFBox并没有直接提供搜索文本的调用方法。所幸的是，通过PDFTextStripper类进行数据流打印时，可以截获到打印的文字信息。因此，可以创建一个类继承自PDFTextStripper，并对载入的PDDocument文档进行一次假的数据流打印操作，这样就可以截获到文字的信息了。

// 触发获取文字信息

```
Writer dummy = new OutputStreamWriter(new  
ByteArrayOutputStream());  
writeText(document, dummy);
```

```
// 截获文字信息
@Override
protected void writeString(String paragraph,
List<TextPosition> textPositions) {
}
```

3.5 占位符查找

截获到文字信息后，需要对每段文字进行检查，判断传入的paragraph是否包含有占位符信息，再通过textPositions获取到文字的信息。由于占位符的内容会变化，并且有可选项，因此不能用常规的文本搜索方式。这里使用正则表达式对传入字符串进行处理。正则表达式是一组由字母、数字和符号组成的特殊文本，可以用来从文本中找出符合要求格式的字符串。通过分析上述的占位符，可以得到匹配的正则表达式为：

```
@(\w+)(%(\w+)){0,1};
```

其中的圆括号可以把查找到的字符串分组，方便后续处理中取出占位符中的打印项和对齐方式。

```
// 设置正则表达式样式
Pattern pattern = Pattern.compile(regex);
Matcher matcher = pattern.matcher(paragraph);
// 重复查找段落中匹配的字符串
while(matcher.find()){
    // 取得匹配字符串信息
    String words = matcher.group(0);
}
```

```
// 取得占位符字符串中，打印项和对齐方式
// 根据正则表达式，打印项和对齐方式分别在组1 和组3
int content = getContentValue(matcher.group(1));
int align = matcher.group(3) == null ? 0 :
getAlignValue(matcher.group(3));
```

3.6 占位符遮盖及数据替换

接下来需要按照刚才找到的占位符的位置，覆盖白色的矩形，把原有的占位符文字遮蔽。考虑到遮盖效果，可以把遮盖范围略微扩大。

完成遮盖后，使用前面载入的字体，把实际的数据替换上去。在实际应用中，可以加载多个字体，根据原有字体的类型信息或用户选择，用不同

的字体插入实际数据。

完成全文的占位符遮盖和数据替换后，将PDDocument另行保存，注意不要覆盖掉模板文件。

```
// 在占位符位置画矩形框并填充
fillRect(contentStream, Color.white, x, y, width, height);
// 在占位符位置插入实际数据
showText(contentStream, font, fontHeight, Color.black,
text, x, y);
```

3.7 图形打印

生成PDF文件后，通过PdfRenderer.Page的render方法，将其转变为Bitmap位图类型。再通过取点采样的方式，就可以得到打印文件每个点的位置、数据等信息。

原有的模板文件可能是彩色文件，而针式打印机只有有点和无点两种状态，所以需要将位图数据采样得到彩色点阵转换为黑白点阵。这里采用了一种简单的采样算法，通过计算彩色的阈值来判断当前点是否需要打印。

```
// 判断当前点是否需要打印
int red = (pixel >> 16) & 0xFF;
int green = (pixel >> 8) & 0xFF;
int blue = pixel & 0xFF;
float gray = (red * 30 + green * 60 + blue * 10) / 100f;
return (1 - gray / 0xFF) > 0.2;
```

通过USB、并口等硬件接口，使用ESC/P指令集，将点阵发送至针式打印机，即可进行称重表格数据的打印。

4 结语

本方案使用占位符替换的方式，通过表格设计、PDF编辑、位图转换、点阵判断等流程，初步实现了智能称重仪表连接针式打印机，进行所见即所得式的复杂称重表格打印。在实际应用中还需要再加入纸张大小选择，打印边距控制，打印参数配置等功能。通过开发，提升仪表的功能和易用性，把技术用到更好地满足用户需求上来。

作者简介

傅啸雷，男（1986—），大学本科，工程师，上海耀华称重系统有限公司工作。研究方向：智能称重仪表与物联网。