

汉字激光照排小史

陈硕 整理于 2024/08

(适逢“748 工程”设立 50 周年)

最新版位于 <https://github.com/chenshuo/typeset>

本文如有任何错误请不吝指正 chenshuo_AT_chenshuo.com

本文大致按年代顺序回顾一下中文印刷，特别是现代汉字激光照排的历史。我注意到有三个因素推动着排版印刷技术的发展：成本 (cost)、效率 (efficiency)、质量 (quality)。其中成本是最关键的，一项新技术要想推广开来，成本优势往往是最决定性的因素，有时甚至可以暂时牺牲质量。

铅字排版与凸版印刷

排版的英文是 typesetting，其中的 type^① 是字模，一般指的是金属活字^②（右图），set 的意思是安置/放置。Typesetting 的字面意思就是安排字模的位置，也就是“排版”。

图 1 中 a 所示部分叫 face，是字模的“脸面”。印刷时，这部分会刷上油墨，压印 (imprint) 纸张，就把“H”这个字迹印到了纸上。这个“脸”的形状决定了印出来的字母的形状，因此字体叫 typeface。

7 世纪初，隋唐时期，中国出现了雕版印刷^③，现存最早的雕版印刷品是印制于唐朝公元 868 年的《金刚经》。

书报常用的正文字体“宋体”^④ 正是为了适应木板刻字而从毛笔书写的“楷体”演变来的。因雕版多用梓 zǐ 木制作，“付梓”即指稿件交付排印。

11 世纪北宋毕昇发明了汉字的活字印刷术^⑤，现存最早的活字印刷品为公元 1103 年《佛说观无量寿佛经》残页。不过活字印刷术在古代中国并未普及，明清

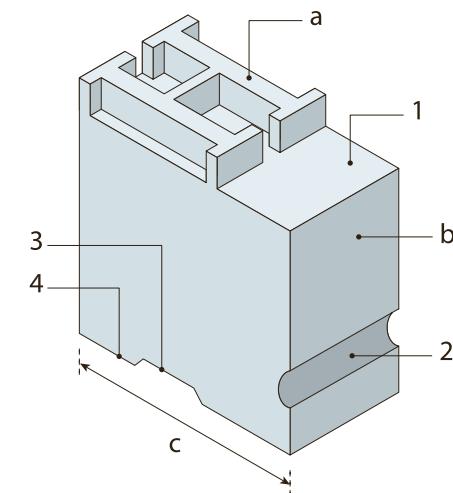


图 1: 字模 (type)，其中 a 叫 face

^①[https://en.wikipedia.org/wiki/Sort_\(typesetting\)](https://en.wikipedia.org/wiki/Sort_(typesetting))

^②https://en.wikipedia.org/wiki/Movable_type

^③<https://zh.wikipedia.org/wiki/雕版印刷>

^④<https://zh.wikipedia.org/wiki/宋体> <https://m.fontke.com/article/24528/> “宋体漫谈”

^⑤<https://zh.wikipedia.org/wiki/活字印刷术>

书籍（小说等）仍以雕版印刷为主。主要原因是汉字数量众多，活字印刷的成本没有明显优势，而且当时的胶泥活字和木活字的印刷质量不及雕版。“汉字字数多”也是汉字激光照排的头号难题。

15世纪中叶，德国人古登堡 (Johannes Gutenberg) 发明了金属活字印刷，印了一百多本《谷登堡圣经》^①，标志着西方图书批量生产的开始^②，也意味着抄书员这个职业的衰退。当时一本《谷登堡圣经》价值 100 克黄金，相当于一个白领文员三年的工资，这比手抄圣经^③便宜得多。



图 2: 拣字排版

第一行文字是 The quick brown fox jumps over the lazy

如上图所示，排版的时候，拣字员一般左手拿字盘^④ 和稿件，右手从面前的字盒 (type case) 里拣取金属活字放到字盘里。通常大写字母放在较远的上层字盒里，因此得名 uppercase；小写字母放在下层字盒里，故名 lowercase，见图 3。

拣字员每次捡出五六行文字，叫作“毛条”，交给拼版师傅进行“组版”操作，才能形成完整的版面。为了阅读方便，书报杂志常常是分栏印刷的，每栏的宽度

^①https://en.wikipedia.org/wiki/Gutenberg_Bible

^②https://en.wikipedia.org/wiki/Printing_Revolution

^③https://en.wikipedia.org/wiki/Biblical_manuscript

^④https://en.wikipedia.org/wiki/Composing_stick



图 3: 上层字盒放大写字母 (uppercase)

一般在 4 至 9 厘米之间，这也便于拣字。其中报纸杂志的栏宽会小一些，因为拿在手里容易卷曲，不似书籍那么平整。

金属活字的主要成分是铅锑合金^①（还有少量的锡），也叫铅活字，这两种金属是有毒的。锑的作用是增加合金的硬度，而且锑有特殊的反常膨胀特性，使得铸造的活字在冷却之后笔画依旧锋利，不会收缩变形。铅活字的长度大约是 23mm (0.918 inch)，假设一栏的宽度是 10cm，每次捡 5 行文字（约 21.2mm），那么字盘的重量有 500g 以上，拿在手里确实不轻。

19 世纪初，清嘉庆年间，西方近代印刷术由英国传教士马礼逊 (Robert Morrison) 传入中国^②，《华英字典》是中国最早用铅活字机械化印刷的出版物，印了 750 套。19 世纪中叶，咸丰年间，美国传教士姜别利 (William Gamble)^③ 在宁波发明了用电镀法制造汉字铅活字铜模，大大降低了制作铅活字的成本。他设计了更为合理的字架，让拣字的效率大为提高。为了与西文混排，他还规定了七种中文字号（初号、二号……），沿用至今。本文的正文是五号字，每个字约 3.7mm (10.5pt) 见方，这个尺寸正是 Gamble 确定的。

19 世纪末，移民到美国的德国钟表匠 Ottmar Mergenthaler 发明了 Linotype 铸排机^④，它能一次完整铸造一整行铅字 (slug^⑤，原意是蛞蝓，俗名鼻涕虫)，让

^①https://en.wikipedia.org/wiki/Type_metal

^②<https://www.icm.gov.mo/rc/viewer/pdfViewerParts/10049/2736>

^③https://www.cgan.net/book/books/print/g-history/gb_9/13_1.htm《中华印刷通史》第 13 章第 1 节

^④https://en.wikipedia.org/wiki/Linotype_machine

^⑤[https://en.wikipedia.org/wiki/Slug_\(typesetting\)](https://en.wikipedia.org/wiki/Slug_(typesetting))

排版效率大大提高。这种机器流行了近一个世纪，到 1970s 年代末才被计算机照排技术淘汰^①。手工操作 Linotype 铸排机，每分钟能排出 3 行新闻行 (newspaper lines)；如果用穿孔纸带驱动，每分钟能排 14 行。



图 4: Linotype slug, 这行文字是 *Good Morning!*

同时期的还有 Monotype 铸排机^②，它是“分体式”的排版设备，由键盘和铸字机 (caster) 组成。键盘会输出穿孔纸带 (paper tape)，再把纸带拿到铸字机上铸造铅字条 (slug)。1968 年出版的《The Art Of Computer Programming》第一版仍然用这种古老的方法排版，Donald E. Knuth 对排版的结果很满意。

激光照排让印刷业“告别铅与火”，“铅”好理解，“火”从何来？我原来想当然地以为铅字排版后可以直接印刷，印完之后把铅字重新放回拣字格子里供下次使用，最近才知道用于排印正文的铅字其实是一次性的。拣字排版后先在纸上印出“大样”，即展示排版结果，供校对用。还要经过压纸型 (flong)^③ 和浇筑拱形铅版^④，才能上滚筒印刷机大量印刷。^⑤ 印完当天的报纸之后，将拱形铅板回炉熔化，排版正文用的铅字也会回炉重铸，印刷厂的铸字车间和浇版车间都有铅炉用于熔铅。由于长期与铅锑等有毒金属接触，印刷厂的工人普遍有职业病。^⑥

^①<https://archive.org/details/FarewellEtaoinShrdlu>

^②https://en.wikipedia.org/wiki/Monotype_system

^③<https://baike.baidu.com/item/纸型/363376>

^④[https://en.wikipedia.org/wiki/Stereotype_\(printing\)](https://en.wikipedia.org/wiki/Stereotype_(printing))

^⑤<https://jnews.xhby.net/waparticles/e5dd22553b49449483d12fc1d9ba882c/lovybSnXYNoT3SKL/1>

^⑥<http://www.93wxjjh.com/cn/impinfo.aspx?mainID=120&cid=1051>

照相排版与平板印刷

铅字印刷属于凸版印刷^①，是有着几百年历史的传统工艺，其工作原理比较直观易懂，无非是大号的印章嘛。到了近代，平板印刷代替了凸版印刷，平板印刷利用油水互斥的基本原理，最初用石灰石 (limestone) 当印板 (printing plate)，因此得名 lithography/石版印刷。经过制版工艺，印板上有文字笔画的部分会沾油墨 (疏水/hydrophobe)，而其余空白部分亲水 (hydrophile)，在印刷的时候，先在石版上涂水，再涂油墨，这样石版上的油墨只沾在文字部分，最后把纸压印在石版上，就得到了成品。^②

胶印 (offset printing)^③是目前主流的平版印刷技术。胶印是把印版上的文字图案油墨先转移 (offset) 到橡胶布 (橡胶滚轮) 上，再印到纸上，因此中文得名“胶印”。这里排版的关键是制作印版，传统做法是先制作胶片 film，然后用胶片晒制 PS 版 (presensitized plate)，现代有计算机直接制版的技术 (computer-to-plate)，本文主要谈论胶片制版。我一开始望文生义，以为胶印是指用胶片印刷，这当然是错误认识。

让感光胶片上呈现文字内容的技术叫“照相排版”，简称“照排”，英文是 phototypesetting^④，输出高质量的胶片是照排机的主要任务，见右图。

目前一提到“照排”，通常指的是第四代激光照排，其实在这之前西文照排系统发展了三代^⑤。第一代是手工照排，然后是采用计算机控制的第二代光学机械式照排机和第三代阴极射线管 (CRT) 式照排机。前两代用遮光板 (stencil) 存储字模，第三代一般是数字存储。不过由于王选教授的卓越贡献，中文电脑照排跳过了第二代和第三代，直接用上了第四代激光照排。第一代到第三代照排是一个字母一个字母地让胶片曝光，而第四代是用一行行激光扫描线让底片感光，因此照排控制器研制难度较大。



图 5: 王选和夫人陈堃鍊检视胶片

^①https://en.wikipedia.org/wiki/Letterpress_printing

^②<https://www.metmuseum.org/about-the-met/collection-areas/drawings-and-prints/materials-and-techniques/printmaking/lithograph>

^③https://en.wikipedia.org/wiki/Offset_printing

^④<https://en.wikipedia.org/wiki/Phototypesetting>

^⑤<https://haagens.com/oldtype.tpl.html>

西文排版的难点

西文正文排版大概有几个步骤，其中每个步骤有自己的难点。

1. 把字母组成单词 word
2. 把单词拼成段落 paragraph
3. 断行 line breaking 与断字 hyphenation
4. 分栏与分页 pagination

第一个步骤看起来简单，“Hello”这个词就是 H, e, l, l, o 这五个字母排在一起，有什么难的？写个 for 循环对各个字母的宽度做累加不就搞定了？

```
for (char ch : word) { output(xpos, ch); xpos += width[ch]; }
```

其实要想获得专业的排版结果，有两个最基本的任务：合字/连字 ligature^① 与缩字 kerning^②。

合字通常发生在 f 和 f, i, j, l, t 等字母相连的时候，经常会用一个字模 glyph 代替两个或三个字母。常见有 find, flow, difficult, shuffle 等词，注意 find 与 find, ffi 与 ffi 的区别。合字通常可以自动处理，但也有例外，例如 shelfful 中的 ff 如果合字反而不那么好看了 shelfful。

缩字 kerning，或者叫字偶距，当两个字母挨在一起的时候，有时会让它们靠得更近一些，常见的有 WAVE, Tom 等等。注意 Tom 和 Tom 的区别，后者的 o 更靠近 T，看起来更协调。

这恐怕是拉丁字母与中文方块字在组词方面的主要区别^③，西文字体通常会包含 ligature 和 kerning 信息。^{④⑤}

把一串字符 (characters) 转换为对应的字模 (glyphs) 并精确定位，这是 text shaping engine 的主要工作，HarfBuzz^⑥ 是广为使用的开源库，支持东西方大多数书面语言。



图 6: 连字的字模

这里其实不是 fi 而是 long S 和 i

^①<https://fonts.google.com/knowledge/glossary/ligature>

^②https://fonts.google.com/knowledge/glossary/kerning_kerning_pairs

^③中文标点挤压有点类似 kerning，《汉语拼音方案》规定“ng 可省作 ŋ”估计可算是 ligature。

^④https://learn.microsoft.com/en-us/typography/opentype/spec/gsub_and_kern

^⑤<https://developer.apple.com/fonts/TrueType-Reference-Manual/RM06/Chap6lcar.html> 和 kern

^⑥<https://github.com/harfbuzz/harfbuzz>

第 2 步，把单词拼成段落，如果不涉及断行的话，这看起来也很简单，无非是把单词后面加个空格就行了。实际上加多宽空格也有讲究，在传统英语排版中，一句话结束后的空格比单词之间的空格要略大^①，就是说句号、问号、感叹号后面会跟大一些的空格。法语的规则比较简单，空格宽度是固定的，称为 French spacing^②，现代排版规则似乎更推荐法式空格^③。

传统英式空格实现起来也没那么简单，因为在英文中，句点(.)除了出现在句末，也可以用于表示缩写词，例如 Mr. Chen 中的 Mr. 是 mister 的缩写，那个句点不是句号，因此 Mr. Chen 之间的空格应该是普通两个单词之间的空格，而不是句末空格。那么一个句点 period 到底是句号 full stop 还是表示单词的缩写呢？排版引擎依据怎样的规则来判断，如果遇到例外情况，如何人工纠正？

第 3 步将一个长的段落 (paragraph) 分割成一些长度相近的行，称为断行^④，然后伸缩单词之间的空白，达到首尾对齐 (justification)。这是决定版面是否易读的关键，基本要求是整齐匀称——文字不能太密也不能太稀疏。因为西文字符不是方块字，每个单词的长度也不一，要达到这一目标并不容易。

断行 (line breaking) 首先有一些硬性的规定，就是哪些地方不能断开。一是某些标点不能出现在行首（逗号、冒号、句号、问号、后引号、后括号等等），某些标点不能出现在行尾（前引号、前括号等等）。中文也有类似的“标点避头尾”^⑤规则，增加顿号、书名号等，而且占两格的省略号和破折号不允许从中间断开。脚注的标号之前也不能断行，上一行出现了反例。这些是最基本的规则，如果违反了，会被视为“不入流”。

对于西文来说，除了标点，有些空格也不宜断开，否则读起来不顺畅。常见有涉及数字和序号的词组，如人名 Charles III、章节名 Appendix A，设想假如在 A 之前断行（就像本行），那么行首的 A 容易被误认为句首冠词，而不是序数词 A/B/C/D。西文这类细节规则很多，计算机不能全都自动识别，往往需要排版人员加以指定。类似地，中文公文排版也有类似“专用词不拆行”的传统。^⑥

在排到接近一行的末尾的时候，如果遇到一个单词，加上其长度会超过行宽，就可以考虑断行了。先看看能不能稍微挤压一下本行前面单词之间的空格，将这个单词排入本行，如果可行，就在挤入这个词之后另起一行。如果挤不下，那就把前面的空格拉伸一些，让行尾对齐，再将这个单词排入下一行的行首。如果发现拉伸之后的空格太宽，就需要对这个单词先断字 (hyphenation)，再从连字号

^①https://en.wikipedia.org/wiki/Sentence_spacing

^②https://en.wikipedia.org/wiki/History_of_sentence_spacing

^③https://en.wikipedia.org/wiki/Sentence_spacing_in_language_and_style_guides

^④https://en.wikipedia.org/wiki/Line_wrap_and_word_wrap

^⑤<https://www.thetype.com/2018/05/14501/> Eric Q. Liu “挤进推出避头尾”

^⑥<https://www.thetype.com/2018/01/14017/> Eric Q. Liu “行行当机不立断”

处断行。(本行是 widow 的例子，应该避免出现。)

以上是人工断行的基本过程，也不难用计算机实现，本质上是贪心算法(greedy algorithm)。但是这样排出来的版面常会有一些稀疏的行(上一行就是)，让版面文字密度不均匀；而且断字较多，影响阅读的流畅性。

对于段落末尾，一般还应防止孤字成行(runt)^①，尤其是防止断字之后的半截儿词单独成行。一般来说，末行的长度至少应该超过首行缩进(indent)，否则就比较难看。这意味着，要么把前面排得更密一些，把孤字挤到上一行去，要么把前面排得稀疏一些，匀出一个或半个词到末行。中文也有类似的孤字不成行规则。

(上一行是“孤字不成行”规则的反例。)

这就说明，一个段落需要整体考虑，才能排出美观的版面。Donald Knuth和他的博士生 Michael Plass 在上世纪 70 年代深入研究了 line breaking 问题，并在 1981 年发表了长达 60 多页的经典论文“Breaking Paragraphs into Lines”^②，提出了 Knuth-Plass 断行算法，采用动态规划(dynamic programming)来找到最优的断行位置，并应用到了 TeX 里。

Knuth-Plass 算法把断行转换为一个组合优化(combinatorial optimization)问题，定义了一个损失函数(文中叫 demerits)，其值由行的疏密程度(文中称为 badness，跟空格的压缩或拉伸程度呈立方关系)和行末罚点(文中称为 penalty，例如断字的默认罚点是 50)确定，这个算法的目标是找到使得各行 demerits 之和最小的断行方法(即最优解)。尽管 Knuth-Plass 是目前我知道效果最好的公开算法，但它仍然有可读性方面的改进余地。^③

- 避免连续两行以相同的单词结尾，不然容易读串行。如果连续几行都以 the 结尾，就不太易读。也会形成 river，感觉版面割裂。
- 避免连续两行发生断字(TeX 对此有罚点)，结合前一条，尤其应避免连续两行末尾相同的词在相同的位置断字。比如说，一本算法书会频繁出现 algorithm 一词，如果连续两行都以上一行这样的方式断字，非常容易读串行。

在找到断点之后，还需要让文字两端对齐(justification)，这样版面才整齐，一眼望去右边缘不会像狗啃了一样。在拉伸空格的时候，如果比较讲究，句末的空格可以比单词之间的空格伸长得更多，同理，在需要压缩空格的时候，句末的空格应该更抗压，这样停顿才合理。TeX 实现了这些细致的操作。不过 TeX 只会拉伸空格单词之间的空格(词距)，不会调整单词内字母的间距(tracking)。另外，允许标点悬挂(句点略微超过右边界)也会让版面更美观。

^①https://en.wikipedia.org/wiki/Widows_and_orphans

^②<http://www.eprg.org/G53DOC/pdfs/knuth-plass-breaking.pdf>

^③<https://tug.org/TUGboat/tb34-1/tb106mitt.pdf>

第 4 步，把一行行的文字排成页面，也叫组版。这其实跟 line breaking 很像，断行是在单词间找到分断点，让每行文字的长度差不多，再左右两端对齐；而分页 (pagination) 是在行间找到分页点，使得每页内容高度差不多，再上下对齐 (vertical justification)。所谓上下对齐，是指一本书翻开后的左右两页的第一行对齐，最后一行也对齐，这样看上去版面比较稳固，不会东倒西歪。如果是分栏的文字，那么左右两栏一般也要上下对齐。这一般通过自动微调行间距来实现。

除了对齐，还有一些影响版面美观的考虑因素，例如孤行控制（防止 widow 和 orphan），这两个术语的含义并不统一，有的地方把 runt 叫 orphan。^①

所谓 widow，指段落的最后一行落在了下一页第一行。例如上一页的页首就出现了 widow 反例。中文也有“孤行不成页”的偏好。排版中的 orphan 原本是指段落的第一行出现在页末最后一行，不过最近的版面规范放宽了对此的要求^②。另外，章节的标题不应该是页面的最后一行，而正文出现在下一页，否则感觉头和身体分离，这也算是避免 orphan 的一个要求。

Michael Plass 的博士论文 “Optimal Pagination Techniques for Automatic Typesetting Systems”^③ 深入探讨了这方面的内容。不过以我的感觉来说，似乎人们对于分页的容忍度要高一些，毕竟记不住前几页到底是从哪儿分隔的，只要当前页面整洁易读就行。

最后，说说断字 (hyphenation)，这是拼音文字独有的，基本规则是在音节 (syllable) 之间断开，因此又叫 syllabification。断字不当会闹笑话，比如 therapists。断字与语言高度相关，本文仅以英语举例，因为我不懂别的外语。英语的拼写和读音之间没有严格的规律，例如 father 和 favorite 中的 fa- 发音就不同，前者接近“发”，后者接近“非”。如果行末出现 fa-，那么读者该默念哪个读音呢？另外，同一个单词 present 作为动词和名词时的发音不同，那么断字的方式也不同 (pre·sent 和 pres·ent)，对于排版软件来说不一定能自动识别。

英语断字的规律不易掌握，一般人需要求助字典才能找到断点。Donald Knuth 指导的博士生 Frank Liang 1983 年的博士论文 “Word Hy-phen-a-tion by Com-put-er”^④ 比较好地解决了英语计算机断字的问题。开源库 libhyphen^⑤ 实现了相同的算法，可以直接调用。

以上这些属于字体排印学 (typography) 的范畴，西文排版发展了近 600 年，有相当多的沉淀。^⑥ 关于字体设计，以后我们有机会再谈。

^①https://fonts.google.com/knowledge/glossary/widows_orphans

^②https://en.wikipedia.org/wiki/Widows_and_orphans

^③<https://tug.org/docs/plass/plass-thesis.pdf>

^④<https://www.tug.org/docs/liang/> 和 <https://tug.org/interviews/liang.pdf>

^⑤<https://github.com/hunspell/hyphen>

^⑥<https://www.pragma-ade.com/pdftex/thesis.pdf>

编年史

1968 年 Stanford 大学 Donald Knuth 教授的《The Art of Computer Programming》^① 第 1 卷出版；次年，第 2 卷出版；1973 年，第 3 卷出版。这套书第 1 版用的是传统的 Monotype 铸字机手工排版，数学公式和计算机代码都很精美。

1973 年 夏天，Bell Labs 的 Ken Thompson 和 Dennis Ritchie 用他们发明的 C 语言重写了 Unix 内核，并于年底发布 Unix V4。

Bell Labs 购买一台 GSI 公司生产的 C/A/T 照排机^②，这是第二代光学机械式照排机。Joe F. Ossanna 用 PDP-11 汇编语言写了 troff 排版软件，用于控制这台照排机。troff 中的 t 代表 typesetter，合起来就是 typesetter roff。roff 是旧的面向行式打印机 (line printer) 的排版软件，只能输出等宽字体。而 troff 可以控制照排机输出专业印刷级别的版面。

CAT 照排机大约有双缸洗衣机大小 (右图)，输出的速度是 50 newspaper lines per minute，比 Linotype 铸排机有数倍的提高。这台机器配有四种字体，每种字体有 102 个字模 (glyphs)，刻在有机玻璃滚筒上 (图中的部件 16)。字体大小 15 级可调 (5pt 到 72pt)，但是缩放是通过改变光学透镜的位置实现的，所以调整字号比较慢。为此早期的 troff 有一个命令行选项 -p 可以以固定的字号输出版面，从而在打印校样 (特别是数学公式较多的文稿) 时获得较高的输出速度^③。

这台照排机可以输出透明胶片 (底片)，也可以输出相纸，曝光之后要经过冲洗 (显影、定影、晾干) 才能看到成品，这也导致了 stderr 的诞生^④。

1974 年 8 月，中国“汉字信息处理系统工程”立项，分三个子项目：汉字通信、汉字情报检索、汉字精密照排，史称“七四八工程”。

介绍 Unix 的经典论文《The UNIX Time-Sharing System》发表在《Communications of the ACM》杂志 7 月刊上^⑤。当时的 Unix 运行在 16-bit 的 PDP-11 小

U.S. Patent Feb. 14, 1978 Sheet 1 of 3 4,074,285

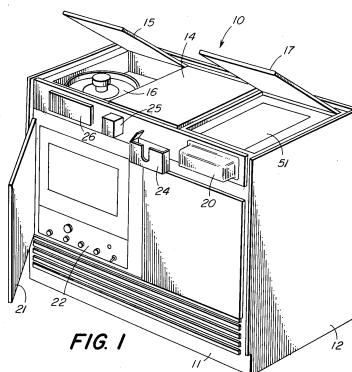


图 7: CAT 照排机外形

^①<https://www-cs-faculty.stanford.edu/~knuth/taocp.html>

^②[https://en.wikipedia.org/wiki/CAT_\(phototypesetter\)](https://en.wikipedia.org/wiki/CAT_(phototypesetter))

^③https://man.cat-v.org/unix_7th/1/troff

^④<https://www2.dms.t.aueb.gr/dds/blog/20131211/index.html>

^⑤<https://dsf.berkeley.edu/cs262/unix.pdf>

型机上，内存地址空间为 64KiB。同年 Unix V5 发布，增加了用于排版数学公式的 eqn(1) 工具。^①

Bell Labs 的 Brian W. Kernighan 和 P. J. Plauger 合写的《Elements of Programming Styles》第 1 版（图 8 左）出版，这是用 troff 排版的第一本书。^②

Donald Knuth 获得图灵奖。

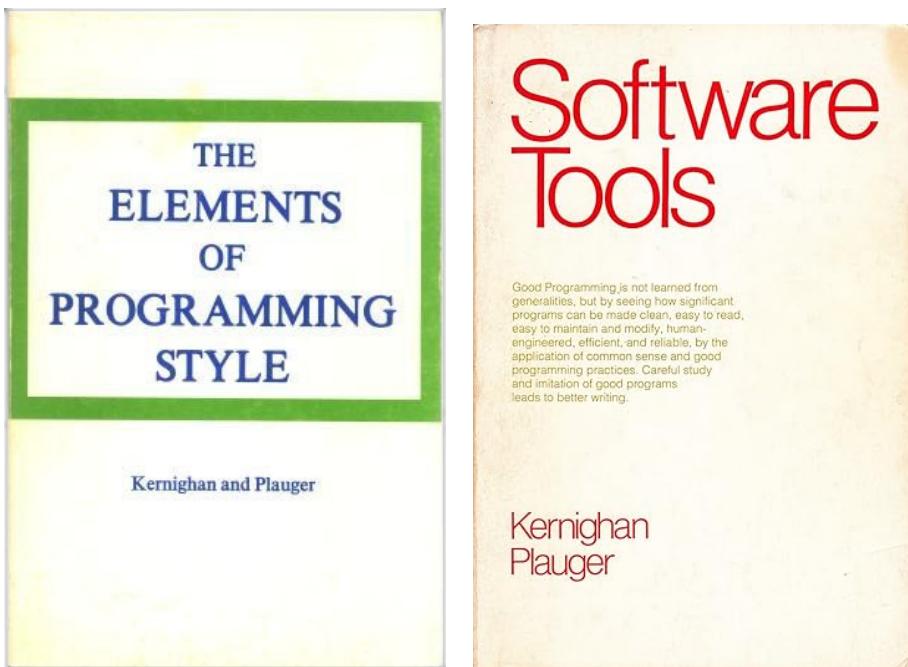


图 8: Unix troff 排版的早期书籍

1975 年 春天，王选得知“七四八工程”，选定其中的“汉字精密照排”子项开始研究，首先确定了数字存储汉字字形的方案。当时全国已有五家从事汉字照排的研究，其中两家搞的是第二代光机式照排机：一家把字刻在有机玻璃平板上，另一家把字刻在有机玻璃滚筒上，跟前述 CAT 照排机类似；另外三家在搞第三代 CRT 式照排机，但都是用的模拟存储，分别是：飞点扫描、字模管、全息模拟存储。王选的数字存储方案首先要解决汉字字形信息量太大的困难，五月，他在一份书面报告中提出用“轮廓加参数”的方法来描述汉字字形，即矢量字库。^③

同年，Unix V6 发布，开始在大学和研究机构流传开来。新增用于排版表格的 tbl(1) 工具。^④

^①Brian W. Kernighan and Lorinda L. Cherry. "A system for typesetting mathematics." CACM 1975/03.

^②Brian W. Kernighan 著 Unix: A History and a Memoir 第 5.3 节，韩磊译《UNIX 传奇：历史与回忆》

^③<https://www.icst.pku.edu.cn/gywm/fzlc/index.htm>

^④M. E. Lesk. "Tbl — A Program to Format Tables." Bell Labs CSTR 49, 1976.

1976 年 王选确定将邮电部杭州通讯设备厂生产的报纸传真机改装成激光照排的输出设备，这台设备的分辨率是 742DPI，一个五号字是 108×108 点（因 $\frac{10.5}{72} \times 742 \approx 108$ ），其字心是 96×96 点。如果不压缩，每个汉字字形需要约 10 000 bits 才能存下，王选的方案能把平均每个字形压缩成 800bits 而不失真。7 月，为了适应当时落后的计算机硬件条件，构思出了“字模信息多级存储”和“分段生成字形点阵”的控制器设计方案。11 月，为了提高激光照排机的输出速度，王选想出四路平行扫描的思路，可以一次（拍）打四个点，同时输出四条扫描线，在滚筒转速不变的情况下，将输出速度提高四倍。

Brian W. Kernighan 和 P. J. Plauger 合写的《Software Tools》第 1 版（图 8 右）出版，扉页说“This book was set in Times Roman and Helvetica Regular by the authors, using a Graphic Systems phototypesetter driven by a PDP-11/45 running under the UNIX operating system.”

Monotype 公司发明了 Lasercomp 激光照排机，这是第一款商品化的第四代激光照排机。

1977 年 3 月，Donald Knuth 收到了出版社用光学照排的《The Art of Computer Programming》第 2 卷第 2 版的校样 (galley proofs)，大失所望。他开始思考自己做排版软件，5 月，他写了第一份备忘录 `TEXDR.AFT①`，交给同事们提意见；6 月，完成了 26 个大小写字母的初步设计；7 月，他新写了一份自认为比较完善规格书 `TEX.ONE②`，交给两个研究生 (Frank Liang 和 Michael Plass) 利用两个月的暑假去实现，然后开始访问中国。9 月，他得知学生们的进展很不顺利，只完成了 15%，于是决定在接下来的一年里 (sabbatical/学术休假) 自己上手做；半年后，次年 2 月，代码录入完毕并编译通过。

12 月，王选组织编写了《七四八工程全电子式汉字精密照排系统方案说明》上中下三册蓝皮本，油印了几百份，在北大和协作单位中流传较多。

1978 年 1 月，在美国数学协会 (*AMS*) 年度 Gibbs 讲座上，Donald Knuth 演讲题目是“Mathematical typography”^③，汇报了他正在做的 `TeX` 排版项目。3 月底，经过紧张调试 `TeX78` 基本可用，实现用的是 SAIL 编程语言，运行在 PDP-10 中型机上。注意，PDP-10 规格比 16-bit 的 PDP-11 更高，PDP-10 的字长是 36-bit，地址空间 18-bit，总内存可达 1MB，足以运行 `TeX` 这种大型程序。5 月，Knuth 开始试排《The Art of Computer Programming》第 2 卷第 2 版，并印出了十来页样张；11 月，排完这本 700 页的著作，期间根据排版需求不断改进 `TeX78④`。9 月，

^①[https://www.saildart.org/TEXDR.AFT\[1,DEK\]1](https://www.saildart.org/TEXDR.AFT[1,DEK]1)

^②[https://www.saildart.org/TEX.ONE\[1,DEK\]1](https://www.saildart.org/TEX.ONE[1,DEK]1)

^③<https://www.ams.org/journals/bull/1979-01-02/S0273-0979-1979-14598-1/S0273-0979-1979-14598-1.pdf>

^④Donald Knuth. The Errors of `TeX`. <https://yurichev.com/mirrors/knuth1989.pdf>

sabbatical year 结束，排版印刷了第一版用户手册 “Tau Epsilon Chi, a system for technical text.” 有近 200 页，供 TeX 早期用户参考。

Bell System Technical Journal (BSTJ) 7/8 月合刊是 Unix 专刊^①，把文档制备 (document preparation) 当作其主要应用之一来介绍，可惜 troff 的原作者 Joe Ossanna 已于 1977 年底因病去世。同年，Brian Kernighan 和 Dennis Ritchie 合著的《The C Programming Language》第 1 版出版 (K&R)，扉页说 “This book was set in Times Roman and Courier 12 by the authors, using a Graphic Systems phototypesetter driven by a PDP-11/70 running under the UNIX operating system.”

1979 年 夏天，Bell Labs 花不到 6 万美元（约合 2024 年的 25 万美元）购买了一台 Mergenthaler 公司（该公司由发明 Linotype 铸排机的那个德国人创立）生产的 Linotron 202 照排机，这是第三代阴极射线管式照排机，输出速度是 350 newspaper lines per minute，是 CAT 照排机的 7 倍。

为了驱动这台照排机，Brian Kernighan 把 troff 改写为与输出设备无关的 ditroff，ditroff 的输出是一种简单的页面描述语言，再经过设备相关的驱动程序去控制 CAT 或 Linotron 202 照排机。

这台照排机一开始运行起来非常不稳定，Bell Labs 的人对其实施逆向工程，替换了原有的控制软件，大大提高了可靠性。还破译了其矢量字体格式（如图 9 所示，是一种折线字体，而不是现在常见的曲线字体），实现自定义字体（国际象棋的棋子）。后来他们把这一过程写成了一份内部备忘录。^②

同年，Donald Knuth 出版了新书《TeX and METAFONT: New Directions in Typesetting》^③，这是 TeX 和 METAFONT 最早的用户手册，恐怕也是最早用 TeX78 排版的正式出版物。我手上正好有这本书，说实话其印刷质量不及后来 TeX82 排印的书，估计原因之一是当时用的输出设备分辨率只有 384DPI^④，还比不上现在的家用激光打印机。同时，Donald Knuth 也开始思考用 Pascal 语言重新实现 TeX，因为 SAIL 是斯坦福大学专有的语言^⑤，只能运行在 PDP-10 机器上，这大大限制了 TeX78 的传播。

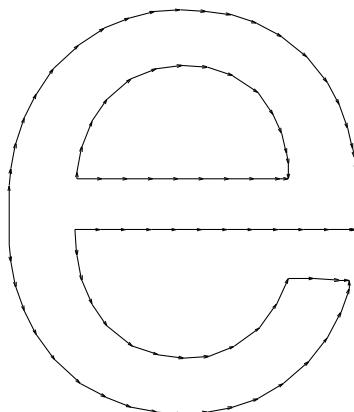


图 9: Linotron 202 的折线字体

^①https://vtda.org/pubs/BSTJ/1978.07_-_Bell_System_Technical_Journal.pdf

^②<https://www.cs.princeton.edu/~bwk/202/>

^③<https://archive.org/details/texmetafontnewdi0000knut>

^④Donald Knuth, “TeX incunabula”, TUGboat, 1984. <https://tug.org/TUGboat/tb05-1/tb09knut.pdf>

^⑤SAIL 是 Stanford Artificial Intelligence Laboratory/Language 的缩写。

Bell Labs 和 Donald Knuth 都投入了相当多的精力来研发字体，可见这是高质量激光照排的关键之一。

7月，华光I型原理性样机排出第一张8开报纸样张（图10），版心宽度65个五号字，约 $240\times350\text{mm}$ 。我复刻了一个清晰版以供查阅^①。《光明日报》8月11日头版用通栏标题作了“汉字信息处理技术的研究和应用获重大突破”的报道，副标题“我国自行设计的计算机—激光汉字编辑排版系统主体工程研制成功”。9月，王选开始把主要精力放在II型机的设计上，II型机的照排控制器取名为TC83, TC是Typesetting Controller 的缩写，预计应在1983年完成。



图 10: 华光 I 型排出的 8 开报纸样张

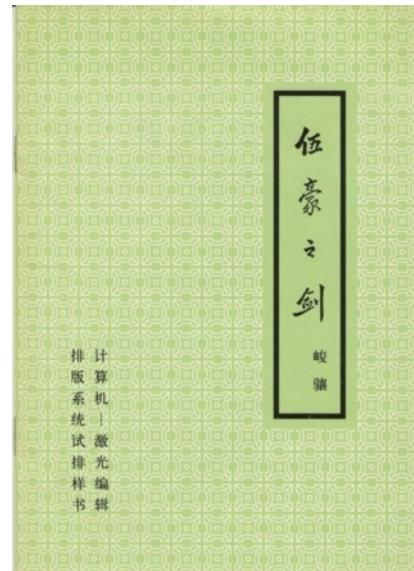


图 11: 样书《伍豪之剑》

1980年9月，华光I型排出32开样书《伍豪之剑》（图11），这本书只有26页，获得了当时最高领导人“应加支持”的批示。^②

2月，TeX 用户组 (TUG) 成立，10月出版了第一份通讯 *TUGboat*^③，其中 Donald Knuth 的两个研究生 Ignacio Zabala 和 Luis Trabb-Pardo 汇报了 Pascal 实现的新 TeX 系统的进展。David Fuchs 介绍了他设计的 DVI 文件格式，这是一种简单的与设备无关的页面描述语言，是 TeX82 的默认输出格式。

^①<https://github.com/chenshuo/typeset/blob/master/wangxuan1979.pdf>

^②<https://www.icst.pku.edu.cn/gywm/fzlc/index.htm>

^③<https://tug.org/TUGboat/Contents/contents1-1.html>

1981 年 《计算机学报》第 2 期几乎是汉字激光照排的专刊，集中刊登了 10 篇相关论文，估计是为 7 月的华光 I 型鉴定会做准备。

- 王选：计算机-激光汉字编辑排版系统简介
- 张合义等人：汉字激光照排机
- 陈堃鍊等人：面向问题的 CL 排版语言及其实现^①
- 俞士汶等人：CL 操作系统的设计与实现
- 王增抗等人：报纸的版面设计
- 陈葆珏等人：高质量汉字字模的存储和调度
- 王选等人：微程序汉字点阵生成器及其模拟和故障诊断
- 裴坤寿等人：激光照排机的扫描控制方式和版面形成
- 甘圣予等人：一种共享字模库的汉字终端系统
- 李新章等人：四路电光调制的激光照排机

Donald Knuth 用 TeX78 重新排版的《The Art Of Computer Programming》第 2 卷第 2 版出版，距离 1977 年项目启动已过去 4 年。

Brian W. Kernighan 发表“PIC — A Language for Typesetting Graphics”^②，介绍用于排版图形的 pic(1) 工具。至此，与 troff 配套的公式 (eqn)、表格 (tbl)、图形 (pic) 排版工具链基本完善，完整命令行是 pic files | tbl | eqn | troff -ms。

1982 年 12 月，John Warnock 和 Charles Geschke 成立了 Adobe 公司，此二人原来在 Xerox PARC^③（施乐 Palo Alto 研究中心）工作，参与了施乐的 Interpress 页面描述语言^④的设计。Adobe 一开始的主打产品是 PostScript 页面描述语言。

这里值得说一说 Xerox 施乐公司，该公司在 1950s 末成功地将静电复印技术 (xerography) 商业化，推出了拳头产品 Xerox 914 复印机，主导了 1960s 年代商用复印机市场。为此公司名字从 Haloid 改为 Xerox，复印机 (photocopier) 也叫 Xerox machine，后来在英语里 xerox 还可以当动词用，意思等同于“复印”，就像 google 如果当动词，等同于“网络搜索”。

Donald Knuth 发布 TeX82，TeX 用户群稳步增长。他在开发 TeX 的早期，主要用 Xerox Graphics Printer (XGP)^⑤ 作为输出设备，这是 Xerox 的试验型点阵 (raster) 打印机，从传真机改装而来。XGP 还不是激光打印机，它的光源是 CRT。XGP 分辨率只有 192 DPI，因此 Knuth 设计的 Computer Modern 字体比较纤细，这样在低分辨率打印输出时笔画粗细正好合适，不至于糊成一团。^⑥

^①CL 是 Computer Laser 的缩写，是后来方正系统的 BD 排版语言的前身。

^②<https://dl.acm.org/doi/10.1145/872730.806459>

^③https://en.wikipedia.org/wiki/Xerox_PARC 其实 Stanford 大学和 PARC 相隔只有几公里

^④<https://en.wikipedia.org/wiki/Interpress>

^⑤https://gunkies.org/wiki/Xerox_Graphics_Printer

^⑥<https://zh.wikipedia.org/wiki/报宋> 中文的“报宋”字体演变也有类似由细到粗的过程

1983 年 中间试验机型华光 II 型问世，照排控制器 TC83 采用 Am2900 位片微处理器 (bit-slicing micro-processor)，集成度大大提高。

Ken Thompson 和 Dennis Ritchie 获得图灵奖。

1984 年 华光 II 型开始在新华社试用，当时这套系统并不稳定，单纯从速度上来讲，激光排版的速度比铅排快不了多少。

Donald Knuth 的《The TeXBook》上市，METAFONT84 完成。

Adobe 发布了 PostScript 页面描述语言 (level 1)，面向印刷市场。同时发布了多款 Type 1 曲线字体，采用三次 Bézier 曲线来描述字形，而且支持字体微调 (hinting)，可以缩放自如。

HP 发布了第一台商业上成功的桌面激光打印机 LaserJet，采用的是佳能 LBP-CX 机芯，分辨率 300DPI，打印控制语言是自家的 PCL，首发价约 3500 美元（折合 2024 年 10500 美元）。佳能的 LBP-CX 机芯是从自家的复印机 PC-20 改装而来，最大的创新是将硒鼓和墨粉做成耗材（墨盒/toner cartridge），提高了可靠性，大大降低了维护成本。不过，由于是从复印机机芯改装，这台打印机的出纸顺序是反着的：印出来有字的一面向上，后面印的页面会盖住前面的。^①

1985 年 新华社用华光 II 型排印《新华社新闻稿》日刊及《前进报》旬报。王选研制成功华光 III 型，是我国第一个实用的科技排版系统，主机由国产 DJS-153 换为美国进口的 DG Desktop 10/SP，这两者指令集是兼容的，大大减轻了代码移植的难度。

Apple 发布了 LaserWriter 激光打印机，也采用的是佳能 LBP-CX 机芯，分辨率 300DPI，零售价高达近 7000 美元。它内置了 PostScript 解释器（也叫 Raster Image Processor，简称 RIP），为此它使用了一枚 12MHz 的 68000 CPU，比与之相连的 Macintosh 主机更快。可以看出，PostScript 的光栅化 (rasterization) 是很耗计算资源的。桌面出版软件 PageMaker 1.0 发布，十年之后被 Adobe 收购。

Leslie Lamport 发布了 L^AT_EX 2.09，大大降低了 TeX 的使用难度和二次开发难度。L^AT_EX 是最流行的 TeX 格式宏包，现在说“用 TeX 排版”，如果没有特别指明，就等于说基于 L^AT_EX 宏包排版。Leslie Lamport 随后将 L^AT_EX 的开发维护工作交给社区，2013 年他获得图灵奖，获奖理由是分布式计算、Paxos 算法等等。

1986 年 经济日报社印刷厂用华光 III 型第一次成功照排 4 开大报《中国机械报》（周报）。该报 1986 年最后一期（12 月 30 日）发表评论员文章“别了，铅排”。

Donald Knuth 出版 *Computers & Typesetting* 丛书 5 卷。^②

^①http://www.hparecive.com/seminar_notes/HP_LaserJet_The%20Early%20History_by_Jim_Hall_110512.pdf

^②<https://www-cs-faculty.stanford.edu/~knuth/abcde.html>

1987年 5月,《经济日报》采用华光III整版编排报纸,一开始软件问题较多,硬件也不稳定,搞得很狼狈。最终华光III系统经受住了每天排印4开大报的考验,是第一代实用的中文激光照排系统。^①

1988年 华光IV型量产。《经济日报》印刷厂卖掉全部铅字,彻底废除了铅排作业,下半年换用IV型系统。《经济日报》热衷于激光照排技术的一个原因是当时其印刷厂位于北京王府井,地盘受限,且环保要求高,被逼采用新技术。^②

1989年 《人民日报》花费430万美元引进美国HTS公司的照排系统失败^③,邀请王选对进口设备进行技术改造。^④

TeX 3.0发布,支持8-bit字符集,Donald Knuth宣布停止TeX开发,将来只做极少量必要的维护。

1990年 《人民日报》全部采用华光汉字激光照排系统,5月28日头版报道“迎接印刷技术革命的挑战——写于人民日报启用激光照排100天”。^⑤

1991年 王选当选中国科学院院士,他研制的第5代照排系统“方正91”上市。

Adobe发布PostScript level 2,很快成为行业标准。

Apple发布TrueType字体,与PostScript Type 1字体竞争。作为回应,Adobe立刻公开了Type 1字体格式的详细规格。^⑥ TrueType采用二次Bézier曲线来描述字形,与Type 1用的三次曲线相比,生成字符点阵的速度快,既可以印刷,也可以用作屏幕显示,而且二次曲线不会出现奇点或自相交等难缠的情况。^⑦

1992年 微软发布Windows 3.1,支持TrueType字体。次年发布Windows 3.2中文版,提供了宋体和黑体两套中文TrueType字体。

1993年 王选指导学生(阳振坤博士)研制出世界上第一个PostScript level 2中文页面解释器,这是方正93电子出版系统的核心。此后王选逐渐淡出科研一线,1994年,当选首批中国工程院院士。

2001年 王选院士获得国家最高科学技术奖(第2届),同年获奖的还有黄昆院士。上一届的获奖人是吴文俊和袁隆平。王选是2023年以前最年轻的获奖者,时年65岁。2006年2月,王选因肺癌病逝,享年70岁。

^①<https://www.pkuef.org/old/info/1175/5334.htm> 汉字激光照排之父王选:科学家的进与退

^②<http://www.rmlt.com.cn/2018/1228/536314.shtml> “当代毕昇”与我国第二次印刷技术革命

^③https://m.thepaper.cn/baijiahao_13029361 《潍坊晚报》2009年国庆特刊《祖国万岁》

^④<https://www.pkuef.org/old/info/1175/6281.htm> 陈望鍊:参与“告别铅与火”的印刷革命

^⑤<https://cn.govopendata.com/renminribao/1990/5/28/1/#870258>

^⑥https://en.wikipedia.org/wiki/PostScript_fonts

^⑦<https://mail.gnu.org/archive/html/freetype-devel/2000-01/msg00052.html>

华光 I~IV 的一些技术细节

本节根据《王选文集》、《计算机学报》1981年第2期、专利申请^①、王选回忆录^②等参考文献，整理总结出一些技术细节，就不一一注明出处了。

华光几代激光照排系统的基本结构类似，都是主机 + 照排控制器 + 激光照排机的结构，以下列出基本配置。

型号	问世	定位	主机	照排控制器
I	1979	原理性样机	DJS-130 小型机	小规模集成电路 + 磁鼓
II	1983	中间试验型	DJS-153 小型机	TC83 (Am2900 微处理器)
III	1985	实用型	DG Desktop 10/SP	TC83 (同上)
IV	1988	大规模生产	长城 286 微机	TC86 (大规模门阵列)

依我个人体会，华光 I 和华光 II 属于科研项目，论文资料相对比较丰富。而华光 III 是实用型照排机，生产了 40 多套（华光 II 一共生产了 8 套），华光 IV 更是一度占据了国内 90% 以上报纸和书刊照排市场，估计卖了上千套。可能是因其涉及商业机密，很多技术细节没有详细披露。

在华光 IV 之后，山东潍坊电子计算机厂和北京大学结束合作。潍坊计算机厂后续推出了华光 V 型，而北大方正则陆续研发了方正 91、方正 93 等照排系统。

以现在的观点，“照排控制器 + 激光照排机”合起来才叫“激光照排机”，或者说现在的激光照排机会内置照排控制器，不过我们仍以当年观点来分开讲各个部件的设计，以下“激光照排机”实际指的是激光影像记录仪 (laser imager)。

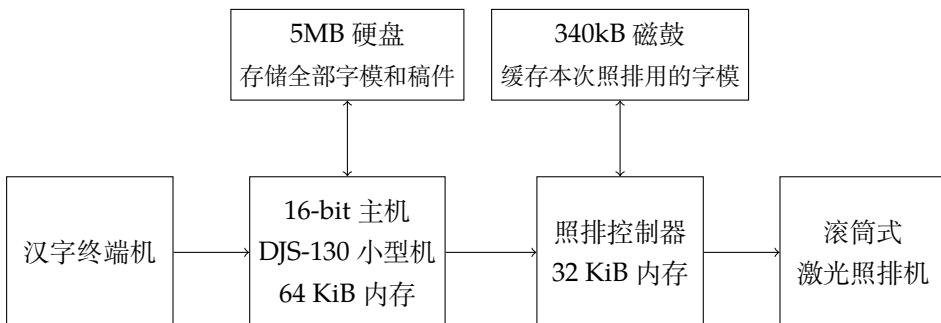


图 12: 华光 I 型原理性样机组成

华光 I 是原理性样机，研制的高峰期间是 1978 年和 1979 年，1979 年 7 月 27 日第一次成功排出 8 开报纸样张（图10），1980 年 9 月 15 日排出一本 32 开样书

^①<https://www.icst.pku.edu.cn/kxyj/kycg/zllb/1222358.htm>

^②<https://wangxuan.pku.edu.cn/wencui?tab=jiguangzhaopai>

(图11)，1981年7月通过鉴定。鉴定结论是“与国外照排机相比，在汉字信息压缩技术方面领先，激光输出精度和软件的某些功能达到国际先进水平”。

输出设备 华光I的激光照排机(激光影像记录仪)系由邮电部杭州通信设备厂(原电信总局五二二厂，现为东方通信股份有限公司)生产的报纸传真机改装而来，是外滚筒结构，底片固定在滚筒的外表面。滚筒外有一激光头小车(图中“扫描头”是也)，安装在与滚筒转轴平行的丝杠上，可上下移动(实际装置中滚筒是水平放置，因此扫描头是横向移动)。滚筒高速旋转，激光在底片上打出光点，完成横向扫描(X轴、主扫描)，同时激光头小车沿轴向匀速移动，完成纵向扫描(Y轴、副扫描)。所以实际上激光扫描线是螺旋线，而不是圆圈。

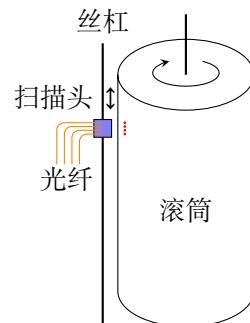


图 13: 滚筒式激光照排机结构示意

分辨率每毫米 29.2 线 (742 DPI)，每个点直径 $25.4\text{mm}/742 \approx 34.2\mu\text{m}$ (1 英寸 = 25.4 毫米)。完整版面是 $16\,756$ 线 (滚筒周长) $\times 11\,388$ 线 (Y 轴)，滚筒转速 1650 RPM，即每秒 27.5 转。由于是四路并行激光输出打点，每转一圈可以扫描 4 条线，一秒钟可以扫完 $4 \times 27.5 = 110$ 线，差不多刚好是一行五号字 (108 线)。华光 I 的版心一行是 65 个五号字 ($108 \times 65 = 7020$ 线)，因此输出速度是 65 字/秒。输出完整的 8 开版面约需 100 秒 ($11\,388/110 \approx 103.5$ 秒)，不到 2 分钟。

版面尺寸 $570\text{mm} \times 380\text{mm}$ ，足以并排两版 8 开报纸，但是需要分两次扫描输出。理论上说可以放一版 4 开的大报，但是直到华光 III 才能出大报。

主时钟为 5.0688MHz 的恒温晶振 (频率稳定度 10^{-8})，经 1024 分频之后得到 495Hz，用于驱动主马达带动滚筒旋转。主马达是高精度同步电机，18 个脉冲旋转 1 周，因 $495/18 = 27.5$ ，一秒钟旋转 27.5 周。每旋转一周，Y 轴的激光扫描头需要移动 4 条线 ($34.2 \times 4 \approx 137\mu\text{m}$)，一秒钟匀速移动 110 线。

5.0688MHz 另外经过 11 分频之后得到 460.8kHz 方波送给照排控制器，用于输出点阵信号，每个周期打出并排的 4 个光点。滚筒每旋转一周，打点 $460.8\text{k}/27.5 \approx 16\,756.4$ 次，我估计这正是版面宽度的由来。由此我推算滚筒的直径约为 $16\,756/742 \times 25.4/\pi \approx 183\text{mm}$ 。注意，为了保证左边界对齐，会用 495Hz 的每 18 个脉冲对 11 分频器清零，这样精确地保证每行的开始在同一位置。

照排机的光学系统经过了几代改良，一开始是四只氦氖激光器，经过四个声光调制器，通过四根光纤接到扫描头；后来改成一只氦氖激光器，经偏振和半反射镜分光成四束光强近似相等的激光，经过四个电光调制器接到四根光纤上，这不仅降低了成本，还使整个照排机的体积缩小了一半；再后来只用一只氦氖激光

器，一个声光调制器，产生四束衍射光，经过透镜在底片上成像 4 个光点，取消了的光纤，降低了光学系统的复杂度和调整难度。

折线字体 用“轮廓加参数”描述汉字字形是王选的重大创举，大大降低了字形存储所需的空间，使得高质量的汉字激光照排成为可能，这里我试图还原一些技术细节。

王选在 1975 年要解决的首要矛盾是当时的硬盘容量很小（以 MB 记）而汉字字数多（每种字体 7000 字起），印刷业对汉字的要求又高，分辨率低了不达标。

首先容易想到的是直接对汉字点阵做压缩，以当时的技术，只有记录黑白段长度的方法，也就是行程码 (run-length encoding)，压缩率不够高，现在通用的无损压缩算法 LZ77/LZ78 在 1975 年还没有问世呢。而且点阵字体 (bitmap fonts) 不容易做缩放，需要每个字体每个字号（从六号到初号有十几档）单独存一份点阵才能保证印刷质量，对汉字来说这么一组合就是天文数字了。

其次是用偏旁部首组字，例如把“新”字拆成“立”“木”“斤”三个字根，再通过拉伸、缩放、平移等操作组合起来。这种方法的压缩率很高，但是组出来的字很难看，笔画粗细不均匀，字的间架结构也不正确，达不到印刷标准。

最终，王选把汉字笔画分为规则笔画（横竖折）和不规则笔画（点撇捺）两大类，所谓“轮廓加参数”，就是用轮廓描述不规则笔画，用参数（笔画的起止位置，笔峰的种类）描述规则笔画。

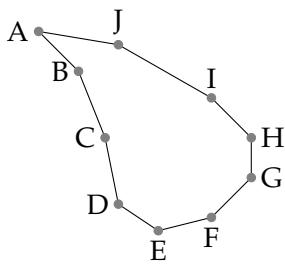


图 14: 不规则笔画“点”的轮廓描述

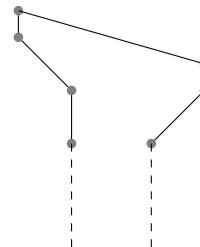


图 15: 规则笔画“竖”的一种起笔笔峰

图 14 用 10 个点描述了某个字中笔画“点”，把 A ~ J 这些顶点依次连起来就是这个“点”，填充之后得到供照排机输出的点阵。图 15 是宋体“竖”笔画的一种起笔笔峰，可以用笔峰编号描述，而不用给出这几个点的具体坐标。这比完全用轮廓描述进一步降低了数据量，对于宋体、仿宋、黑体等“印刷体”来说优势尤为显著。平均下来每个字形需要 100 字节，一套包含 7000 常用字的字体需要 700kB 存储空间，这比今天的 TrueType 字体小一半以上。

用参数来描述规则笔画的另一个好处是保证笔画的粗细一致，王选常用“量”字举例，“量”有9横，无论字体如何缩放，这9横应该一样粗，否则印出来就很难看。“量”在 128×128 点阵下每横是3个点高，缩小到 64×64 点阵时，不能有些横是2点，有些横是1点，用参数描述法就比较容易保证这一条。

华光 I ~ 华光 IV，以及后来的华光 V 和方正 91 系统均采用这套字体技术，这是轮廓字体 (outline fonts)，也可以叫矢量字体 (vector fonts)。但是跟今天的 PostScript 字体和 TrueType 字体有所不同，现代的矢量字体是用二次或三次 Bézier 曲线来描述字体轮廓，而王选当年是用折线（一次“曲线”）来逼近曲线，便于硬件快速实现，跟同时期的 Linotron 202 折线字体是类似的（图 9）。我推测方正 93 用了现在流行的曲线字体，可惜很难找到当年的折线字体来对比了。

华光的六种正文字体（书宋、报宋、小标宋、仿宋、黑体、楷体）以五号字为基本字号，其余字号由五号字变倍而来。五号字的大小是 108×108 点阵，字心 96×96 点，字形设计是在 96×96 的坐标纸上完成的（图 16）。六套字体合计 4.2 万个字头，我推算需要 4.2MB 空间。

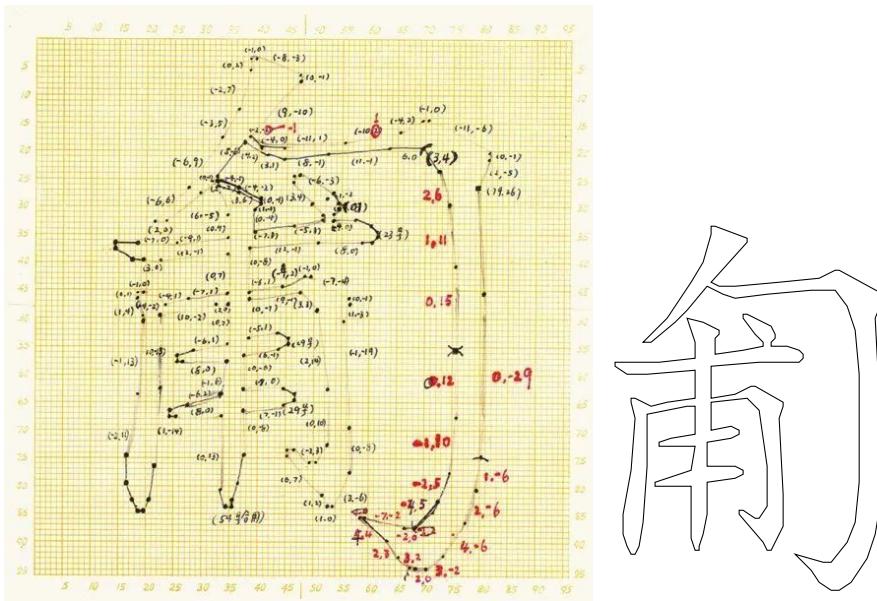


图 16: 王选的“匍”字设计手稿及字形复刻

为了印大字号的标题，华光还提供几种标题字体（大标宋、大黑、隶书、魏碑等），以四号字作为大字体的基本字号，字心 128×128 点，可以放大到特大号（ 512×512 点）仍然保持笔划边缘光滑。这几种字体估计提供 3000 个常用字就够日常排版用了。

照排控制器 是整个系统的核心，这其实是一台专用 (special purpose) 的 16-bit 计算机，负责将压缩的汉字字形复原为供激光照排机输出的点阵，也是王选倾注心血最多的地方。

华光 I 的照排控制器是用国产 74 系列 TTL 小规模集成电路搭的，相当于国外 1960s 初期的硬件水平。照排控制器外接一个容量 340kB 的国产磁鼓^①，用于缓存本次照排用的字形信息；照排控制器的内存是国产磁心 (core)，容量只有 32KiB，这些都是 1960s 末国外已经淘汰的存储设备。

磁鼓的转速是 1500 RPM，读完全部数据需要转 12 圈，耗时 480ms。340kB 的容量大约可以存 3400 个不同的字形，足够排一份报纸的全部版面，或者一本本书。根据统计，通常一本书的正文用宋体，用 3000 多个不同的字，标题用黑体，用几百个不同的字，这样 340kB 勉强够用。

华光 I 照排控制器的数据内存很小，只有 32KiB，不足以缓存整个版面（一版 8 开报纸需要约 10MB 的空间），甚至连一行五号字的点阵信息也不能完整保存。为了克服硬件限制，王选设计了汉字分段复原的办法。版心宽度是一行 65 个五号字，即横向 $65 \times 108 = 7020$ 点，纵向 8 个点为一段，缓存一段点阵需要 7020 字节（实际分配了 7040 字节，估计是取 8 的整数倍），可供激光照排机输出两圈（四路并行打点）。照排控制器的内存中划出 $2 \times 7040B \approx 14.1kB$ 用作点阵缓冲，这是典型的双缓冲技术 (double buffering)，点阵生成器和扫描控制器交替读写这两块点阵缓冲区，一个五号字需要分成 14 段复原，只要将生成一段点阵的时间控制在 72ms 以内 ($\frac{1000}{27.5} \times 2 \approx 72.73$)，就可以跟得上照排机的输出速度。

为了能在 72ms 内复原一行 65 个字，这些字的字模信息（约 7kB）来不及从磁鼓读取，为此内存中划出 $2 \times 7296B \approx 14.6kB$ 用于缓存两行字模。照排机输出一行五号字需要约 1s，因此来得及从磁鼓读下一行所需的字模放入字模缓冲区。

一行文字的照排信息有 512 字节，两行合计 1KiB，这样 32KiB 内存只余下 3.0KiB 用于常数及工作单元。

控制器流水线工作的大致过程是：

1. 根据下一行的照排信息确定哪些字模需要放入下一行的字模缓冲区
2. 根据当前行的照排信息和当前行的字模生成本行汉字的下一段点阵
3. 扫描控制器取出当前行当前段（8 线）点阵送激光照排机（优先级最高）

以上 2、3 两个步骤会反复若干次完成本行汉字照排，而步骤 1 会穿插在其中，以完成下一行汉字的准备工作。用现在的话说，1 和 2 是预取 (prefetching) 或预读 (read-ahead)。

华光 I 型的字模分三级存储，硬盘上存全部字体全部汉字（几 MB），磁鼓上

^①https://en.wikipedia.org/wiki/Drum_memory

缓存当前照排版面所需的汉字（几千个，几百 kB），照排控制器的内存里缓存当前行出现的几十个汉字（几 kB），这是为了应付低速的磁鼓和更慢的硬盘，在硬件升级之后这些巧妙的设计就过时了。

“全部采用国产元器件的照排控制机含 28 块大插件板，每块板比现在 PC 机的母板还大，放在一个大机柜中，磁鼓驱动器也占一个机柜，再加上磁鼓，体积不小。”

王选 1979 年 9 月开始设计华光 II 型机，他决定用国外刚刚面世的 Am2900 位片微处理器^①来实现 II 型机的照排控制器，称为 TC83。与 I 型机相比，大致有以下几方面的改进：

- 磁鼓换成了 512KiB RAM，依旧用于存放本次照排用到的全部几千个字模，在照排过程中无需读硬盘。由于 RAM 可以随机访问，相应地三级字模存储简化为两级，不再需要单独缓存一行汉字的字模。
- 依旧采用逐段生成汉字点阵的方案，一段从 8 线改成 32 线，版面宽度增加到 8192 点 ($\frac{8192}{742} \times 25.4 \approx 280\text{mm}$)，因此点阵缓冲区增大到 $2 \times 32 \times 8192/8 = 64\text{KiB}$ ，是原来的 4 倍多。
- 由于采用了集成度更高的微控制器，整个 TC83 只有 200 个芯片（不含 DRAM），与原理性样机相比，元件减少了一个数量级。
- 支持激光照排机和激光打印机输出，共享字形发生器和控制器。为此，把杭州产的激光打印机分辨率定为 371 DPI，刚好是该厂照排机分辨率的一半，硬件容易实现。

华光 III 的照排控制器还是 TC83，主机换成了 DG Desktop 10/SP 台式机，和原来的国产 DJS-153、进口 Nova 等小型机指令集兼容。排版软件功能大大增强，支持科技排版（数学公式）和大报排版。

华光 IV 的主机换成了 PC 机（长城 286/386），照排控制器叫 TC86，从 1984 年 9 月开始设计，1988 年成为可靠的商品，所以来也改称 TC88。TC86 用 16-bit Am29116 微处理器加两片专用的门阵列芯片（分别叫 WA 和 WI）实现，五号字复原速度到了空前的 710 字/秒。还用 40MB 内存实现了整版点阵缓冲（支持图文混排），最终淘汰了逐段缓冲汉字点阵的方案。^②

方正 91 的照排控制器的集成度进一步提高，把华光 IV 的 WA 和 WI 门阵列和其他一些外围电路集成到一个 CG 芯片里，只占微机的一个插槽。^③

^①https://en.wikipedia.org/wiki/AMD_Am2900

^②40MB RAM 整版缓冲这一说法来自 <https://wangxuan.pku.edu.cn/article?id=1023>，但是专利《高分辨率汉字字形发生器》CN85100285A 中有不同的说法：“主存储器 SS 的容量一般为 512KB—2MB。其中一部分用于存放本次照排所用的几千个汉字的字形压缩信息，称为字形压缩信息区；另一部分用于缓冲一页版面的交替两行或若干行的点阵信息供激光扫描用，称为扫描缓冲区。”

^③https://www.ccf.org.cn/Computing_history/Full_List/2020/First_class/2021-01-20/722005.shtml

心愿

- 收藏一套《七四八工程全电子式汉字精密照排系统方案说明》三册蓝皮本
- 获得华光 I ~ 华光 IV 的折线字体文件和格式说明，复刻当年的输出效果。

感想

- 排版系统非常复杂，保守估计有 10 年以上的工作量，其中有无数的细节和 corner cases 需要应对。^①
- 挑剔的眼光（字面意义上）才能做出好的排版系统。无论是字形、字体渲染、字距、行距、断行等方面，typographer 总能发现现有排版结果的不足，特别是 ebook 有巨大的提升空间。
- 从原型到产品是一个艰巨的过程，“主体工程研制成功”可能只是跑通了一两个基本的 end-to-end tests。
- 往往每一次技术升级，都意味着一代熟练工人的分 xià 流 gǎng。从铅字印刷到激光照排，从内燃机车到电力机车^②，固定电话从人工交换（接线员/operators）到 PSTN^③ 再到现在的 VoIP。
- 待补……

如发现本文有各种错误请不吝赐教

<https://github.com/chenshuo/typeset/issues>

https://twitter.com/bnu_chenshuo/status/1827131054182035592

^①<https://ieeexplore.ieee.org/document/6591969> “There is more to typesetting than setting type”

^②<https://new.qq.com/rain/a/20240313A040TV00>

^③https://en.wikipedia.org/wiki/Public_switched_telephone_network