

基于Linux操作系统核心的汉字显示（2）PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/144/2021_2022__E5_9F_BA_E4_BA_8E_Linu_c103_144365.htm “hello,world.”中的每一个字符都要经过conv_uni_to_pc()这个函数转换成8位的显示字符。这样做的主要目的是使不同语言的国家能把16位的Unicode码映射到8位的显示字符集里，目前主要还是针对欧洲国家的语言，映射结果为8位，不包含双字节(double byte)的范围。这种从Unicode到显示字符的映射表上，会把中文的字符映射到其他的字符上，这是我们不希望看到也是不需要的，所以我们有两种选择：1) 不进行conv_uni_to_pc()的转换 2) 加载符合双字节处理的映射关系，即对蜚控制字符进行一对一的不变映射，我们自己定制的符合这种映射关系的Unicode码表是direct.uni。要想看/装载当前系统的Unicode映射表，可使用外部命令loadunimap。经过conv_uni_to_pc()转换之后，“hello,world.”中的字符被一个一个地填写到tty的缓冲区中，然后do_con_write()调用底层的驱动程序，把缓冲区中的内容输出到显示器上（也就相当于把缓冲区的内容拷贝到VGA显存中去）：

```
sw->con putcs(vc_cons[currcons].d, (u16 *)draw_from, (u16 *)draw_to, (u16 *)draw_rwom, Y, draw_x).
```

之所以要调用底层驱动程序，是因为存在不同的显示设备，其对应VGA显存的存取方式也不一样。上面的Sw->con_putcs()就会调用fbcon.c中的fbcon_putcs()函数(con_putcs是一个函数的指针，在Framebuffer模式下指向fbcon_putcs()函数，也就是说，在do_con_write()函数中是直接调用了fbcon_putcs()函数来进行字符的绘制，比如说在256色模式下，真正负责输出的

函数是：void fbcon_cfb8_putcs(struct vc_data *conp, struct display *p, const unsigned short *s, int count, int YY, int xx)

100Test 下载频道开通，各类考试题目直接下载。详细请访问
www.100test.com