

将Linux代码移植到Windows的简单方法（4）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\\_B0\\_86Linux\\_E4\\_BB\\_c103\\_144613.htm](https://www.100test.com/kao_ti2020/144/2021_2022__E5_B0_86Linux_E4_BB_c103_144613.htm) 在编译过程中,很多错误是有由lib目录下的文件产生的,但是lib目录下的文件不是完全都需要的。lib目录只是一个对Tar的补充库。需要的代码才需要编译。具体判断的方法一个是参考Windows C Library库的内容。如果同样的函数,数据类型已经定义,就不需要Lib目录中的相同数据类型的定义和函数实现了。还有一个方法是尽量去掉lib目录中的C文件,只保留头文件,并使得编译能够通过,根据link的错误信息去检查那些lib中的C文件是需要的。除了修改外围的各种头文件之外,还不要忘了修改工程的编译选项,特别是预定义选项。在Tar的移植过程就需要以下的预定义HAVE\_CONFIG\_H, \_POSIX\_SOURCE, MSDOS。HAVE\_CONFIG\_H 表示程序编译需要config.h文件。为了方便期间,在tar移植过程中就放到工程的预编译选项中了。MSDOS, 移植的是Linux下的控制台程序,而Windows平台最接近Linux控制台就是DOS,特别是一些环境变量设置和全局常量的定义。Tar的有些代码针对MSDOS环境已经做了一部分修正,这点在移植过程中可以利用起来。还有一个可选项是\_\_CYGWIN\_\_。有些Linux程序会针对Cygwin平台做出代码上的特殊设定。当遇到这样的代码的时候,一定要加上\_\_CYGWIN\_\_预定义项,能够大大减少移植需要的工作量。还有就是移植过程引入的各种Cygwin代码中也可能需要\_\_CYGWIN\_\_定义(有时候是其他的定义,比如\_POSIX\_SOURCE, 或者\_\_INSIDE\_CYGWIN\_\_)。经过上

述的几个步骤。第一个目标，代码能够编译通过基本上是不会有问题的。只要把握好二个修改代码的基本原则，第一。引入新的代码，而不修改原有的代码。在没有办法进行调试前修改源代码是不允许的，修改的不好就会引起最后代码运行逻辑的混乱，而且在代码能够运行之前是很难发现问题的。所以除非非常有把握，否则不要修改被移植工程的源代码。第二，引入新的代码之后，不能因为这次引入而需要再次引入新的代码。这样子，就进入死循环了。为了解决某个数据类型的定义，而引入了新的不能解释的数据类型。这样还不如不引入新的代码。所以引入新的代码，特别是很多头文件。引入之前一定要做修改，只保留工程本身需要的部分，去除那些不需要的代码。直到能编译通过为止。三：第二个目标，使得代码能够链接过（Link）完成了第一个目标之后，就会有大量的link错误。原因是前面引入了很多外部函数，外部全局常量只有定义而没有实体，于是就会产生link错误。现在需要的是为代码提供引入的函数实体，外部全局变量实体。一般都是函数link(本文于2003年完成. 如需要转载 请联系jackforce at 163.com)不到的比较多。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](http://www.100test.com)