volatile int i=10;   
int j = i;   
...   
int k = i;   
  
volatile 告诉编译器i是随时可能发生变化的，每次使用它的时候必须从i的地址中读取，因而编译器生成的可执行码会重新从i的地址读取数据放在k中。   
  
volatile 影响编译器编译的结果,指出，volatile 变量是随时可能发生变化的，与volatile变量有关的运算，不要进行编译优化，以免出错，（VC++ 在产生release版可执行码时会进行编译优化，加volatile关键字的变量有关的运算，将不进行编译优化。）。  
  
其中编译器编译优化是：  
由于编译器发现两次从i读数据的代码之间的代码没有对i进行过操作，它会自动把上次读的数据放在k中。而不是重新从i里面读。这样以来，如果i是一个[寄存器变量](http://zhidao.baidu.com/search?word=%E5%AF%84%E5%AD%98%E5%99%A8%E5%8F%98%E9%87%8F&fr=qb_search_exp&ie=utf8)或者表示一个端口数据就容易出错，所以说volatile可以保证对特殊地址的稳定访问，不会出错。