[keybd\_event使用方法](http://blog.csdn.net/zhanglidn013/article/details/35988381)

分类： [c++编程](http://blog.csdn.net/zhanglidn013/article/category/915751)2014-06-30 11:25 2218人阅读 [评论](http://blog.csdn.net/zhanglidn013/article/details/35988381#comments)(0) [收藏](javascript:void(0);) [举报](http://blog.csdn.net/zhanglidn013/article/details/35988381#report)

[VC++ DOS命令](http://www.csdn.net/tag/VC%2b%2b%20DOS%e5%91%bd%e4%bb%a4)[键盘事件模拟](http://www.csdn.net/tag/%e9%94%ae%e7%9b%98%e4%ba%8b%e4%bb%b6%e6%a8%a1%e6%8b%9f)

Windows提供了一个模拟键盘API函数Keybd\_event（），使用该函数可以相应的屏蔽键盘的动作。Keybd\_event（）函数能触发一个按键事件，也就是说会产生一个WM\_KEYDOWN或WM\_KEYUP消息。

该函数原型如下：  
  
　　VOID keybd\_event(  
  
　　BYTE bVk, // virtual-key code  
  
　　BYTE bScan, // hardware scan code  
  
　　DWORD dwFlags, // flags specifying various function options  
  
　　DWORD dwExtraInfo // additional data associated with keystroke  
  
　　);  
  
　　从原型可以看出，Keybd\_event（）共有四个参数：

      第一个为按键的虚拟键值，如回车键为vk\_return, tab键为vk\_tab（其他具体的参见附录：常用模拟键的键值对照表）；

      第二个参数为扫描码，一般不用设置，用0代替就行；

      第三个参数为选项标志，如果为keydown则置0即可，如果为keyup则设成"KEYEVENTF\_KEYUP"；

      第四个参数一般也是置0即可。

例子1：模拟按下'A'键  
　　 keybd\_event(65,0,0,0);  
　　 keybd\_event(65,0,KEYEVENTF\_KEYUP,0);

例子2：模拟按下'ALT+F4'键  
　　 keybd\_event(18,0,0,0);  
　　 keybd\_event(115,0,0,0);  
　　 keybd\_event(115,0,KEYEVENTF\_KEYUP,0);  
　　 keybd\_event(18,0,KEYEVENTF\_KEYUP,0);

附：常用模拟键的键值对照表（也可参考<http://msdn.microsoft.com/en-us/library/dd375731(v=vs.85).aspx>）

                                                         键盘键与虚拟键码对照表

　　　 　　字母和数字键 数字小键盘的键功能键

其它键   
　　　　　　键　　 键码　  　键　　 键码　　　 键　　 键码 　　  键　　　　键码   
　　　　　　A　　　65　　   0 　　96 　　　　F1 　　112 　　Backspace 　　　8   
　　　　　　B　　　66　　   1　　 97 　　　　F2 　　113　　 Tab 　　　　　　9   
　　　　　　C　　　67 　　  2 　　98 　  　　F3 　　114　　  Clear 　　　　　12   
　　　　　　D　　　68　　　3　　 99 　　　　F4 　　115　　Enter 　　　　　13   
　　　　　　E　　　69 　　  4 　　100　　　　F5 　　116　　Shift　　　　　 16   
　　　　　　F　　　70 　　  5 　　101　　　　F6 　　117　　Control 　　　　17   
　　　　　　G　　　71 　　  6　　 102　　　　F7 　　118 　　Alt 　　　　　　18   
　　　　　　H　　　72 　　　7 　　103　 　　F8 　　119　　Caps Lock 　　　20   
　　　　　　I　　　73 　　　8 　　104　　　　F9 　　120　　Esc 　　　　　　27   
　　　　　　J　　　74 　　　9　　 105　　　　F10　　121　　Spacebar　　　　32   
　　　　　　K　　　75 　　　\* 　　106　  　　F11　　122　　Page Up　　　　 33   
　　　　　　L　　　76 　　　+ 　　107　　  　F12　　123　　Page Down 　　　34   
　　　　　　M　　　77 　　　Enter 108　　　　-- 　　--　　　End 　　　　　　35   
　　　　　　N　　　78 　　　-　　 109　　　　-- 　　-- 　　　Home　　　　　　36   
　　　　　　O　　　79 　　　. 　　110　　　　--　　 -- 　　 　Left Arrow　　　37   
　　　　　　P　　　80 　　　/ 　　111　　　　--　　 -- 　　 　Up Arrow　　　　38   
　　　　　　Q　　　81 　　　-- 　　--　　　 　--　　 -- 　　 　Right Arrow 　　39   
　　　　　　R　　　82 　　　-- 　　--　　　　--　　 -- 　　 　　Down Arrow 　　 40   
　　　　　　S　　　83 　　　-- 　　--　　　　　-- 　　-- 　　 　Insert 　　　　 45   
　　　　　　T　　　84 　　　-- 　　--　　　　　--　　 -- 　　 　Delete 　　　　 46   
　　　　　　U　　　85 　　　-- 　　--　　　 　-- 　　-- 　　 　Help 　　　　　 47   
　　　　　　V　　　86 　　　--　　 --　　　　-- 　　-- 　　 　Num Lock 　　　 144   
　　　　　　W　　　87 　　　　　　　　　  
　　　　　　X　　　88 　　　　　  
　　　　　　Y　　　89 　　　　　  
　　　　　　Z　　　90 　　　　　  
　　　　　　0　　　48 　　　　　  
　　　　　　1　　　49 　　　　　  
　　　　　　2　　　50 　　　　　　  
　　　　　　3　　　51 　　　　　　  
　　　　　　4　　　52 　　　　　　  
　　　　　　5　　　53 　　　　　　  
　　　　　　6　　　54 　　　　　　  
　　　　　　7　　　55 　　　　　　  
　　　　　　8　　　56 　　　　　　  
　　　　　　9　　　57

模拟键盘平时不是很常用, 但是当调用某些快捷键执行某项功能时, 它真的是那么的方便呀.  你不信?  看看下面的实现, 你就会大呼: 为什么不早点告诉我?  呵呵,  原来没有blog呀,  都靠这些挣分呢.

1) 显示桌面:

         很多软件有显示桌面的功能, 并且大家的方法都是遍历窗口,  然后让它们最小化, 其实 win系统给咱们了一个非常方便的WIN键(就是键盘上在CTRL键和ALT键之间的那个带win标志的按键), 利用它, 可以轻松的完成显示桌面的功能.

    keybd\_event(VK\_LWIN, 0, 0 ,0);  
    keybd\_event('M', 0, 0 ,0);  
    keybd\_event('M', 0, KEYEVENTF\_KEYUP ,0);  
    keybd\_event(VK\_LWIN, 0, KEYEVENTF\_KEYUP,0);

其他的操作也类似, 比如直接显示开始的运行,就把上面的'M'换成'R'即可。

直接    keybd\_event(VK\_LWIN, 0, 0 ,0);  
            keybd\_event(VK\_LWIN, 0, KEYEVENTF\_KEYUP,0);

直接显示“开始”对话框了。

2) 实现快速的全选

        很多的时候，比如listctrl实现全选，你可以用listctrl循环设置每一项的状态为选中，多罗索的事情呀。用快捷键试一试CTRL＋A，其他的快捷键一样的用法，呵呵，你知道怎么办了吧？

    keybd\_event(VK\_CONTROL, (BYTE)0, 0 ,0);  
    keybd\_event('A',(BYTE)0, 0 ,0); //此处可以用  'A', (BYTE)65, 用'a'不起作用.  
    keybd\_event('A', (BYTE)0, KEYEVENTF\_KEYUP,0);  
    keybd\_event(VK\_CONTROL, (BYTE)0, KEYEVENTF\_KEYUP,0);

3) 执行某些特殊的键，比如数字键，大小写，下面是数字键的例子

    bool bState=true;   //true为按下NumLock,false反之  
    BYTE keyState[256];  
      
    GetKeyboardState((LPBYTE)&keyState);  
    if( (bState && !(keyState[VK\_NUMLOCK] & 1)) ||  
        (!bState && (keyState[VK\_NUMLOCK] & 1)) )  
    {  
        // Simulate a key press  
        keybd\_event( VK\_NUMLOCK,  
            0x45,  
            KEYEVENTF\_EXTENDEDKEY | 0,  
            0 );  
          
        // Simulate a key release  
        keybd\_event( VK\_NUMLOCK,  
            0x45,  
            KEYEVENTF\_EXTENDEDKEY | KEYEVENTF\_KEYUP,  
            0);  
    }

4） 你想CTRL＋ALT＋DELETE三键一起按下，

    keybd\_event(VK\_CONTROL, 0, 0 ,0);  
    keybd\_event(VK\_MENU,0, 0 ,0);  
    keybd\_event(VK\_DELETE,0, 0 ,0);

    keybd\_event(VK\_CONTROL, 0, KEYEVENTF\_KEYUP ,0);  
    keybd\_event(VK\_MENU,0, KEYEVENTF\_KEYUP ,0);  
    keybd\_event(VK\_DELETE,0, KEYEVENTF\_KEYUP ,0);  
呵呵，这样不会成功呀，因为这几个键直接是操作系统来截获执行的，而模拟键盘只能发向应用程序，所以这种方法不行的（想显示锁定对话框，用    LockWorkStation();）

5） Window2000/NT/XP已经不提倡用这个函数了，上面的方法只是为了让大家开阔一下思路，怎么替代呢，呵呵，看下面，所以上面的所有代码都可以用这个来完成

   //2000下用这个代替 ,包含 "winable.h"  
    INPUT input[4];  
    memset(input, 0, sizeof(input));

    input[0].type = input[1].type = input[2].type = input[3].type = INPUT\_KEYBOARD;

    input[0].ki.wVk  = input[3].ki.wVk = VK\_LWIN;  
    input[1].ki.wVk  = input[2].ki.wVk = 'R';

    //接下来释放它，这一点很重要。  
    input[2].ki.dwFlags = input[3].ki.dwFlags = KEYEVENTF\_KEYUP;  
    input[0].ki.time = input[1].ki.time = input[2].ki.time = input[3].ki.time = GetTickCount();

    SendInput(4, input, sizeof(INPUT));

感觉比那个有点罗索，呵呵。

＝＝＝＝＝＝＝＝＝＝＝＝＝＝＝＝＝＝＝＝

附WIN键的部分快捷键:

WIN键+D=快速的切到桌面，再次点击返回

WIN键+E=快速打开资源管理器

WIN键+R=“运行”。

WIN键+M=全部视窗最小化。

WIN键+Shift+M=取消全部视窗最小化。

WIN键+F1=Help。

WIN键+F=“寻找”。

WIN键+Ctrl+F=显示“查找电脑”。

WIN键+Tab=切换工作列的程式。

WIN键+Break=显示系统内容。

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

                          关于SendMessage keybd\_event  PostMessage

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

首先你会发现keybd\_event函数中是没有窗口句柄作为参数的，好奇的你一定会觉得很奇怪，那是因为，keybd\_event是全局模拟按键的，只对前台窗口（即当前的活动窗口）才可以，但是如果模拟的按键正好也是某个窗口的全局热键消息，那该窗口也能接收到的

  而SendMessage 、PostMessage是对指定句柄窗口都其作用的，对于做一些一外挂是非常有用的。例如可以做成这样的效果：即用SendMessage/PostMessage在某一个窗口模拟动作，而同时自己可以在其他窗口做其他事情，互不影响的！

  但是有一点要注意，很多人在模拟键盘消息的时候，都会忘记模拟WM\_KEYUP的消息。。。

  还有一点就是,PostMessage中的窗口句柄参数，可以设置为HWND\_BROADCAST，即广播，但不要理所当然地认为是对所有的窗口都起作用！！！它只对系统的顶层窗口起作用，子窗口是收不到这个消息的！！！

  还要注意的是SendMessage是没有HWND\_BROADCAST参数的，那是因为，SendMessage总是等发送的消息在对应的窗口消息队列中处理完毕后才返回的（这是一种负责的行为）， 细想一下就知道为什么SendMessage没有HWND\_BROADCAST参数了！！

个人认为

　　1. 当需要模拟键盘输入命令时，比如Ctrl + V，选择keybd\_event；

　　2. 当需要模拟键盘输入一串字符时，选择PostKeybdMessage；

　　3. 当需要模拟键盘输入单个字符时，选择keybd\_event。

SendMessage(Button.Handle,WM\_LBUTTONDOWN,0,0);　　鼠标左键按下   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifSendMessage(Button.Handle,WM\_LBUTTONUP,0,0);　　　鼠标左键抬起  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifSendMessage(Edit.Handle,WM\_SETTEXT,255,Integer(PChar('abc')));　传递文本  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifSendMessage(Edit.Handle,WM\_Char,Wparam('Q'),2);　　传递字符  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifSendMessage(Button.Handle,BM\_SETSTYLE,BS\_RADIOBUTTON,1);　　改变Button风格  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifSendMessage(ComboBox.Handle,CB\_SETDROPPEDWIDTH,300,0);　　改变CBDownWidth  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifWM\_CUT、WM\_COPY和WM\_PASTE　　剪切，复制，粘帖   
  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif实现任意组合键  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifkeybd\_event(VK\_Control, MapVirtualKey(VK\_Control, 0), 0, 0);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifkeybd\_event(ord('V'), MapVirtualKey(ord('V'), 0), 0, 0);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifkeybd\_event(ord('V'), MapVirtualKey(ord('V'), 0), KEYEVENTF\_KEYUP, 0);   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifkeybd\_event(VK\_Control, MapVirtualKey(VK\_Control, 0), KEYEVENTF\_KEYUP, 0);

模拟鼠标按下事件  
::PostMessage(hwd0,WM\_MOUSEMOVE, MK\_LBUTTON, MAKELONG(x,y));  
::PostMessage(hwd0,WM\_LBUTTONDOWN,MK\_LBUTTON,MAKELPARAM(x,y));  
::PostMessage(hwd0,WM\_LBUTTONUP,MK\_LBUTTON,MAKELPARAM(x,y));  
  
  
模拟键盘按下事件  
keybd\_event(VK\_TAB,0,0,0);  
keybd\_event(VK\_TAB,0,KEYEVENTF\_KEYUP,0);