
Subject: How to set a global (system) keyboard hook?

Posted by [midra](#) on Thu, 17 Sep 2009 22:37:05 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi!

I'm new to U++

My problem is: how can I set a global (system) keyboard hook?

I want my dialog to appear on top of other windows when user presses some combination of keys (e.g. Control+Alt+C)

My program is supposed to run in Windows.

Thanks in advance for any help!

Best regards,
midra

Subject: Re: How to set a global (system) keyboard hook?

Posted by [mirek](#) on Fri, 18 Sep 2009 03:55:37 GMT

[View Forum Message](#) <> [Reply to Message](#)

midra wrote on Thu, 17 September 2009 18:37Hi!

I'm new to U++

My problem is: how can I set a global (system) keyboard hook?

I want my dialog to appear on top of other windows when user presses some combination of keys (e.g. Control+Alt+C)

My program is supposed to run in Windows.

Thanks in advance for any help!

Best regards,
midra

You have to use host platform API (Win32).

I guess this one is what you need:

<http://msdn.microsoft.com/en-us/library/ms646309%28VS.85%29.aspx?ppud=4>

Then put GetHWND of some of your TopWindows in there and override

```
virtual LRESULT Ctrl::WindowProc(UINT message, WPARAM wParam, LPARAM lParam);
```

(Do not forget to call TopWindow::WindowProc at the end).

Mirek

Subject: Re: How to set a global (system) keyboard hook?

Posted by [tojocky](#) on Fri, 18 Sep 2009 06:01:14 GMT

[View Forum Message](#) <> [Reply to Message](#)

luzr wrote on Fri, 18 September 2009 06:55

You have to use host platform API (Win32).

I guess this one is what you need:

<http://msdn.microsoft.com/en-us/library/ms646309%28VS.85%29.aspx?ppud=4>

Then put GetHWND of some of your TopWindows in there and override

```
virtual LRESULT Ctrl::WindowProc(UINT message, WPARAM wParam, LPARAM lParam);
```

(Do not forget to call TopWindow::WindowProc at the end).

Mirek

I'm interested in global(system)keyboard too.

but in linux how it can be realized? Is the standard method?

For X11 I found here:

```
int XGrabKey(Display *display, int keycode, unsigned int  
modifiers, Window grab_window, Bool owner_events, int pointer_mode, int keyboard_mode);
```

```
int XUngrabKey(Display *display, int keycode, unsigned int  
modifiers, Window grab_window);
```

Maybe it is time to add cross method in u++?

Subject: Re: How to set a global (system) keyboard hook?

Posted by [tojocky](#) on Fri, 18 Sep 2009 14:03:51 GMT

[View Forum Message](#) <> [Reply to Message](#)

About global hotkey!

FOR WIN32:

acourding by simple example from

[http://msdn.microsoft.com/en-us/library/ms646309\(VS.85\).aspx?ppud=4](http://msdn.microsoft.com/en-us/library/ms646309(VS.85).aspx?ppud=4)

first we need to register hot key by calling:

```
BOOL RegisterHotKey(  
    HWND hWnd,  
    int id,  
    UINT fsModifiers,  
    UINT vk  
);
```

and in U++ is already created a thread that execute function DWORD WINAPI Ctrl::Win32OverwatchThread(LPVOID). In this function is:

```
DWORD WINAPI Ctrl::Win32OverwatchThread(LPVOID)  
{  
    .....  
    MSG Msg;  
=>>HERE while(GetMessage(&Msg, NULL, 0, 0) > 0) {  
    TranslateMessage(&Msg);  
==>>HERE we can process our keyboard event  
    if(IsWindowUnicode(Msg.hwnd))  
        DispatchMessageW(&Msg); // send to appropriate thread  
    else  
        DispatchMessage(&Msg);  
    }  
    .....  
}
```

The problem is that I do not get keycode property from Msg. From where can I get it?

for linux X11:

according by documentation from:

1. <http://www.xfree.org/current/XGrabKey.3.html#toc2>

The simple code like this:

```
Window root;  
XEvent ev;
```

```
Display * dpy = XOpenDisplay(0);
```

```
if(!dpy) return 1;
```

```
root = DefaultRootWindow(dpy);
```

```

char * key_string = "F3";

KeyCode key = XKeysymToKeycode(dpy, XStringToKeysym(key_string));

XGrabKey(dpy, key , AnyModifier, root,
         True, GrabModeAsync, GrabModeAsync);

for(;;)
{
  XNextEvent(dpy, &ev);
  if(ev.type == KeyPress && ev.xkey.keycode == key){
    ==>> Do our action
  }
}

```

First we need register global hotkey by calling:

```

int XGrabKey(Display *display, int keycode, unsigned int
modifiers, Window grab_window, Bool owner_events, int pointer_mode, int keyboard_mode);

```

In u++ we can get message from void Ctrl::EventLoop0(Ctrl *ctrl)

which call void Ctrl::ProcessEvent(XEvent *event)

The good news is that I have keycode from the returned message. like this "ev.xkey.keycode".

The Question:

2. Is possible to add in u++ cross-os method for register global hotkey event?

For first I propose like this:

```

static bool Ctrl::AddGlobalHotkey(Key p_key, callback
p_func)

```

2. If is possible for register global hotkey event, maybe it is possible to unregister global hotkey event.

Maybe I was not understandable.

UPDATE:

For linux I found interesting example:

```

void
xstuff_grab_key_on_all_screens (int    keycode,
    quint  modifiers,
    gboolean grab)

```

```

{
GdkDisplay *display;
int      n_screens;
int      i;

display = gdk_display_get_default ();
n_screens = gdk_display_get_n_screens (display);

for (i = 0; i < n_screens; i++) {
GdkWindow *root;

root = gdk_screen_get_root_window (
gdk_display_get_screen (display, i));

if (grab)
XGrabKey (gdk_x11_display_get_xdisplay (display),
keycode, modifiers,
gdk_x11_drawable_get_xid (root),
True, GrabModeAsync, GrabModeAsync);
else
XUngrabKey (gdk_x11_display_get_xdisplay (display),
keycode, modifiers,
gdk_x11_drawable_get_xid (root));
}
}

```

UPDATE 2:

Good news is for win32 too. It returns keycode in Mgs->IParam.

Ion Lupascu (tojocky).

Subject: Re: How to set a global (system) keyboard hook?

Posted by [mirek](#) on Sun, 20 Sep 2009 07:18:50 GMT

[View Forum Message](#) <> [Reply to Message](#)

Good work tojocky.

Mirek

Subject: Re: How to set a global (system) keyboard hook?

Posted by [mirek](#) on Thu, 24 Sep 2009 21:05:29 GMT

[View Forum Message](#) <> [Reply to Message](#)

Implemented.

See reference/HotKey

Mirek

Subject: Re: How to set a global (system) keyboard hook?

Posted by [tojocky](#) on Fri, 25 Sep 2009 07:55:55 GMT

[View Forum Message](#) <> [Reply to Message](#)

luzr wrote on Fri, 25 September 2009 00:05Implemented.

See reference/HotKey

Mirek

VERY NICE!

TESTED FOR WIN32 XP SP3 AND UBUNTU 9.04

BUT:

fro UBUNTU when I try to hint key: Ctrl + Alt + F11 then my gnome resets.
May be it is reserved hotkey for gnome?

I have another question: how can I hint CTRL+C+C?

I can do this when i get double CTRL+C CTRL+C and verifying interval from it to not be greater than 1 second.

Thank you Mirek, great job!

Subject: Re: How to set a global (system) keyboard hook?

Posted by [cocob](#) on Fri, 25 Sep 2009 09:21:00 GMT

[View Forum Message](#) <> [Reply to Message](#)

With X11 ctrl+alt+F.. is reserved for terminal switching, usually F1,F2,F3,... are text terminals and F7 is the graphical one.

cocob

Subject: Re: How to set a global (system) keyboard hook?

Posted by [tojocky](#) on Thu, 01 Oct 2009 12:11:36 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hello Mirek!

I found the interesting situation about Global Hotekey:

I want to register global hotkey for Ctrl+C+C and get from clickboard copied data.

I did so:

1. Register system hotkey Ctrl+C

```
TranslateHotKeyId = Ctrl::RegisterSystemHotKey(K_CTRL_C,  
THISBACK(GlobalShortcutHandle));
```

2. in my callback I wrote:

```
void GoogleTranslatorDemo::GlobalShortcutHandle(){  
    if(!AtomicRead(test_test)){  
        AtomicWrite(test_test,1);  
  
        bool need_translate = false;  
  
        if(shortcut_stage==0){  
            shortcut_stage = 1; // first Ctrl+C  
            time_stop_shortcut.Reset();  
        }  
        else if (time_stop_shortcut.Elapsed(<600){  
            need_translate = true;  
            shortcut_stage = 0;  
        }  
        else {  
            shortcut_stage = 1; // first Ctrl+C  
            time_stop_shortcut.Reset();  
        }  
  
        if(need_translate){  
            WString cur_text = ReadClipboardUnicodeText();  
  
            if(cur_text.GetLength()){  
                inputwindow.textedit.Set(cur_text);  
                TranslateTextInBaloon();  
            }  
        }  
        AtomicWrite(test_test,0);  
    }  
}
```

On the second press Ctrl+C start translating.

The problem:

When I press Ctrl+C from the other application the standard command (copy text) do not react. In other words did not coping the selected text.

How can I do this?

Subject: Re: How to set a global (system) keyboard hook?

Posted by [mirek](#) on Thu, 01 Oct 2009 18:55:04 GMT

[View Forum Message](#) <> [Reply to Message](#)

tojocky wrote on Thu, 01 October 2009 08:11 Hello Mirek!

I found the interesting situation about Global Hotekey:

I want to register global hotkey for Ctrl+C+C and get from clickboard copied data.

I did so:

1. Register system hotkey Ctrl+C

```
TranslateHotKeyId = Ctrl::RegisterSystemHotKey(K_CTRL_C,  
THISBACK(GlobalShortcutHandle));
```

2. in my callback I wrote:

```
void GoogleTranslatorDemo::GlobalShortcutHandle(){  
    if(!AtomicRead(test_test)){  
        AtomicWrite(test_test,1);  
  
        bool need_translate = false;  
  
        if(shortcut_stage==0){  
            shortcut_stage = 1; // first Ctrl+C  
            time_stop_shortcut.Reset();  
        }  
        else if (time_stop_shortcut.Elapsed(<600){  
            need_translate = true;  
            shortcut_stage = 0;  
        }  
        else {  
            shortcut_stage = 1; // first Ctrl+C  
            time_stop_shortcut.Reset();  
        }  
  
        if(need_translate){  
            WString cur_text = ReadClipboardUnicodeText();  
  
            if(cur_text.GetLength()){  
                inputwindow.textedit.Set(cur_text);  
            }  
        }  
    }  
}
```

```
    TranslateTextInBalloon();  
  }  
}  
AtomicWrite(test_test,0);  
}  
}
```

On the second press Ctrl+C start translating.

The problem:

When I press Ctrl+C from the other application the standard command (copy text) do not react. In other words did not coping the selected text.

How can I do this?

This cannot work. Once you register system hotkey, it gets "captured" by your application - that is the feature of both win32 and X11.

I suggest to use a little bit more exotic combos, like K_ALT|K_CTRL|K_SHIFT|K_C.

Mirek

Subject: Re: How to set a global (system) keyboard hook?
Posted by [tojocky](#) on Fri, 02 Oct 2009 06:03:14 GMT
[View Forum Message](#) <> [Reply to Message](#)

luzr wrote on Thu, 01 October 2009 21:55

This cannot work. Once you register system hotkey, it gets "captured" by your application - that is the feature of both win32 and X11.

I suggest to use a little bit more exotic combos, like K_ALT|K_CTRL|K_SHIFT|K_C.

Mirek

First Ctrl+C I need for copy in clipboard and the second Ctrl+C for processing copied text from clipboard (in my case is process translating).

Question:

Is not possible the given message to send follow to the other application?

Subject: Re: How to set a global (system) keyboard hook?
Posted by [mirek](#) on Fri, 02 Oct 2009 08:06:12 GMT

tojocky wrote on Fri, 02 October 2009 02:03luzr wrote on Thu, 01 October 2009 21:55

This cannot work. Once you register system hotkey, it gets "captured" by your application - that is the feature of both win32 and X11.

I suggest to use a little bit more exotic combos, like K_ALT|K_CTRL|K_SHIFT|K_C.

Mirek

First Ctrl+C I need for copy in clipboard and the second Ctrl+C for processing copied text from clipboard (in my case is process translating).

Question:

Is not possible the given message to send follow to the other application?

AFAIK, no.

Mirek
