
Subject: Re: GET KEY STATE for Linux and win32
Posted by [tojocky](#) on Fri, 02 Oct 2009 11:11:19 GMT
[View Forum Message](#) <> [Reply to Message](#)

The second variant of code is:

```
bool get_key_state(dword p_k){
    bool result;
#ifdef PLATFORM_WIN32
    result = true;
    //... Here need to optimize
    if(p_k & 0xffff)
        result &= (bool)(((char)::GetKeyState(p_k & 0xffff)) & 128);

    if((result)&&(p_k & K_ALT))
        result &= GetAlt();

    if((result)&&(p_k & K_SHIFT))
        result &= GetShift();

    if((result)&&(p_k & K_CTRL))
        result &= GetCtrl();
#elif PLATFORM_X11
    result = true;
    char key_map_stat[32];
    XQueryKeymap(Xdisplay, key_map_stat);

    if(p_k & 0xffff){
        KeyCode k_code = XKeysymToKeycode(Xdisplay, p_k & 0xffff);
        result &= ((key_map_stat[k_code >> 3] >> (k_code & 7)) & 1);
    }

    if((result)&&(p_k & K_ALT)){
        const KeyCode alt_l = XKeysymToKeycode(Xdisplay,XK_Alt_L);
        const KeyCode alt_r = XKeysymToKeycode(Xdisplay,XK_Alt_R);

        result &= (((key_map_stat[alt_l >> 3] >> (alt_l & 7)) & 1)
            ||((key_map_stat[alt_r >> 3] >> (alt_r & 7)) & 1));
    }

    if((result)&&(p_k & K_SHIFT)){
        const KeyCode shift_l = XKeysymToKeycode(Xdisplay,XK_Shift_L);
        const KeyCode shift_r = XKeysymToKeycode(Xdisplay,XK_Shift_R);

        result &= (((key_map_stat[shift_l >> 3] >> (shift_l & 7)) & 1)
            ||((key_map_stat[shift_r >> 3] >> (shift_r & 7)) & 1));
    }

    if((result)&&(p_k & K_CTRL)){
```

```

const KeyCode control_l = XKeysymToKeycode(Xdisplay,XK_Control_L);
const KeyCode control_r = XKeysymToKeycode(Xdisplay,XK_Control_R);

result &= (((key_map_stat[control_l >> 3] >> (control_l & 7)) & 1)
||((key_map_stat[control_r >> 3] >> (control_r & 7)) & 1));
}
#else
result = false;
//ASSERT_(false, "Get key state is not implemented for this platform")
#endif
return result;
}

```

This will process correct:

```

if(get_key_state(K_CTRL_C)){
.....// add our code
}

```

Mirek, is it useful to add in U++?
