Bome's Mouse Keyboard

USER MANUAL

1 Overview

This program simulates a music keyboard which you can play with the mouse or the computer keyboard. You can control your MIDI devices without having a real keyboard - for example you can play the internal synth of your sound card. With a virtual MIDI cable (for example Hubis Loopback Device or MIDI Yoke) you can play any sequences into your sequencer or route your external synthesizer through the chord-maker of Bome's Mouse Keyboard. Additionally, you can define knobs and control with them any MIDI controller. Furthermore, you can use your mouse wheel and the joystick to send controller messages.

2 System Requirements

- Windows 98/ME/NT4/2000/XP (Vista should work, too)
- Some kind of MIDI connection/synthesizer/soundcard

Optionally supported:

- Mouse with wheel
- Joystick

3 Register/Purchase

This program is not Freeware. It's PostcardWare. If you find that Bome's Mouse Keyboard is useful for you and you want to continue using it, you have 2 options:

1. Pay the registration fee (more details).
   Or you can use the secure server to directly purchase the program.
2. Send me a real postcard (for private use only):
   Find out more at the postcardware site.

In both cases, you'll receive an email with a personalized password to remove the reminder screen at startup. Allow 4-12 weeks for postcard shipment and processing. With an online order, you'll get the license key immediately.

4 Quick Start

This is a quick guide to get started with Mouse Keyboard:

1. Select a MIDI out device in the menu. Everything that you play will be sent to that MIDI device.
2. Now you can already play: Use the left mouse button or the keyboard keys. A right click will play in Pitching Mode: While the right mouse button is pressed, you can alter the pitch/modulation by moving the mouse horizontally/vertically.
3. Adjust the size of the keyboard by
   • resizing the window
   • changing the base note (Alt+Mouse drag)
   • adjust the key width (in the Settings Dialog)
4. Change the chord and play: Ahhh !
5. To find a good patch/program, enter the Hold Mode by playing a note with CTRL+right mouse button. The note will keep on being played when you release the mouse button. Now, change the program number or patch - it will replay your note for each new program so it's made easy browsing through the programs. The same applies to the bank numbers or chords. Which bank numbers to use is detailed in the documentation of your MIDI device.
6. Change the master volume with the volume knob. You can add more knobs in the menu Knobs|Set Up...

5 Keyboard Shortcuts

There are a number of key shortcuts that you can use:

Function	Shortcut
chord +/-	Pos1/End
patch +/-	PageUp/PageDown
program +/-	Shift+PageUp/Shift+PageDown
Panic	Ctrl+Space
Transpose	Ctrl+T
Settings	Ctrl+S
Knob Editor	Ctrl+K
Chord Editor	Ctrl+E
Choose Patch set	Ctrl+P
Octave +/-	F3/F4
Apply Snapshot	F5...F12
Store Snapshot	Ctrl+F5....Ctrl+F12

6 Function Reference

When you have a real keyboard connected to your computers MIDI port, select this as MIDI in and check the menu Options|Midi thru. Like this, the keys you play on the real keyboard are visualized on the mouse keyboard and the note and any chords you selected are routed to the MIDI out device.

To use several mouse keyboards at once (with their own settings and knobs) create a copy of MouseKeyboard.exe under a new name, e.g. SynthKeyb.exe. SynthKeyb will then use its own set of settings.

6.1 Mouse Operations

There are several modes to play with the mouse:

- Left mouse button: *Normal Mode*: the note/chord that you press down will be played. Moving the mouse while having the mouse button pressed will always play the new note/chord and release the old one.
- Right mouse button: *Pitching* mode: only the initially pressed note/chord will be played. With moving the mouse while the right mouse button is still pressed you can alter 2 controllers - one is changed by moving up/down, the other one by moving left/right. When you release the right mouse button, the note/chord will stop playing. You can set these controllers in the Settings dialog. By default, you change pitch and modulation wheel.
- CTRL+mouse button: *Hold Mode*: When you release the mouse button while holding down CTRL, the note/chord will keep on playing. The note will not be released before you play another note with the mouse. While a note is on hold, you can change patch/program, chords, etc. to browse through them: each time you change something, the note will be replayed with the new patch/program/chord.
- SHIFT: Pressing SHIFT while playing will lower the velocity of the played notes/chords. The velocity is the dynamic volume of the key. It differs from the general volume in that it does not change the volume of anything else played on that synthesizer - just the notes played with the mouse are affected. You can set the amount of lowering in the Settings dialog.

6.2 Other mouse operations

- ALT+left mouse button on the keyboard: Drag the mouse keyboard left/right to change the view of keys.
- Wheels (at left of keyboard): You change the value by dragging the

mouse up and down. Holding SHIFT while dragging is the *Fine Mode* which will give fine control. Double clicking the wheel will revert it back to its default position. For the pitch wheel, holding CTRL while releasing the mouse button will keep the wheel at its position without rolling back into the middle position. You can change the controller of the right wheel in the Settings dialog.

- Mouse Wheel: (the little wheel on the mouse itself - where present): You can change a controller with the mouse wheel. Holding down shift while rolling is the Fine Mode as with operating the wheels. The controller which is changed with the mouse wheel can be chosen in the Settings Dialog.
- Knobs: They work exactly the same way as the wheels (including Fine Mode with SHIFT and double click). However, the knobs will be selected (blue on selection) and grab the mouse wheel while selected. So rolling the mouse wheel while a knob is selected, will always change the knob controller. Once a knob is selected, you can also change its value with the cursor up/down and page up/down keys.

6.3 Computer Keyboard Usage

The computer keyboard has 2 manuals mapped to the notes: The first manual occupies the first 2 rows: the 1st row ("23567..." on English keyboard layout) are black keys, while the second row ("qwerty..." on English layout) are the white keys. The second manual is used analogously with the third and fourth rows of the computer keyboard. It is one octave lower than the first manual. By default, the first white key is mapped to note *C*, i.e. keys Q and Z on the English keyboard layout. You can change this in the Settings Dialog.

Playing with SHIFT pressed down will lower the velocity just as it does when playing with the mouse.

6.4 Knobs

You can define knobs that appear on the main keyboard window. A knob can send MIDI controller messages, so that you can control your favorite MIDI software or MIDI device by way of these controllers. If you have selected a MIDI input port/device, the knobs will also react to incoming controllers.

When you first start Mouse Keyboard, one default knob is already set up: the one for the volume controller which affects the main channel volume of what you play.

In order to add a knob, use the menu function Knobs|Set up (or the keyboard shortcut Ctrl+K). The Knob Preset Editor will open. With the *knob count* buttons you can increase the number of knobs to e.g. 2. A new tab

titled *Knob 2* appears, where you can define this new knob:
1. Choose the controller number in the drop-down box
2. You can change the name to anything you want, this name appears over the knob in the main window.
3. The default value will be set when you double click the knob in the main window
4. The display range is either signed (-64..0..+63) or unsigned (0..127). E.g. *Panorama* is a signed controller, but volume is unsigned.

After editing the knob(s), you should save them as a preset. A **knob preset** is the definition of a collection of knobs. Press the New button and enter the name of this knob preset. Now when you add another knob, just use the Save button to update the knob preset.

In order to activate this preset, click the *Apply* or *OK* button, and you should see your knob(s) in the main keyboard window.

In the main window, use the knobs by dragging them up and down. A **double click** will instantly reset them to their default value.

When you have created several knob presets, you can quickly select them in the menu Knobs|Presets.

When you want to quickly change all knobs to predefined positions, take a **Snapshot**: In the menu, choose Knobs|Snapshots|Define|Snapshot 1 to define the first snapshot. You can also press Ctrl+F5. Enter an arbitrary name for this snapshot. This will take a snapshot of all positions of all knobs. Now whenever you choose this first snapshot with Knobs|Snapshots| Snapshot 1, or by pressing F5, all knobs will instantly jump to the pre-recorded positions. You can define up to 8 snapshots.

6.5 Patches

The **instrument** that you play is selected with the *Patch* drop down list. A patch is comprised of the name of the instrument, and its setting for MIDI program and MIDI bank, which are displayed in the second row of the Mouse Leyboard's main screen.

When you select a patch in the drop down list, you will see that the program and bank numbers will change accordingly. Conversely, changing the program and/or bank will try to find a matching patch in the patch list, and if successful, select it.

Now different soundcards/synthesizers/tone generators can have different instruments assigned to the MIDI programs and MIDI banks. The most common standard is the General MIDI standard which is used as default patch list in Mouse Keyboard. However, if your synthesizer is not General MIDI, then the names in the patch list are not meaningful.

In order to adapt the patch names to your synthesizer, you can load a different *Patch Set*. For that, use the menu Patches|Choose Patch Set (or the keyboard shortcut Ctrl+P). Bome's Mouse Keyboard ships with a wide selection of patch sets. They are provided by Steinberg in the form of the **Cubase Patch Name Script** format. So if your synthesizer is not supported by the included patch sets, try to find a patch set online, e.g. by searching for *Cubase patch name script* and the name of your synthesizer. Alternatively, you can create your own patch set definitions. For that, see the Appendix A: patch script specification.

7 Joystick Support

You can use up to 2 joysticks to generate MIDI controllers. For that, open the settings window (Ctrl+S) and change to the Joystick tab.

In the left drop-down list, choose the joystick action to define. In the right drop-down list, choose the controller that is sent for the selected joystick action. Repeat that for every action that you want to use.
To enlarge the number of different actions, and controllers that can be sent with the joystick, the first 4 joystick buttons can be used as SHIFT keys. For that, select the *button X pressed* actions in the joystick action list. Those controllers that you define will be sent pressing down the specified button, and then moving the joystick in the specified direction.

8 Transpose

Mouse Keyboard allows transposing your played notes by a specified number of semitones. E.g. a transpose of 2 means that all notes are sounding 2 semitones higher; hitting a C key will cause a D note to be played. You can also transpose down with negative transposition values.

To activate transposition, invoke Options|Transpose in the menu, or press Ctrl+T. Usually, transpose is only active as long as this window is open. Closing it will restore non-transposed mode. To keep the transpose after closing the transpose window, and also across Mouse Keyboard sessions, check the *Keep transpose when this window is closed* checkbox.

In order to **transpose incoming MIDI notes**, check *Transpose MIDI Thru notes*. Provided that you have enabled Options|Midi Thru, you can use Mouse Keyboard to transpose the notes played on an external keyboard, or any MIDI device selected as MIDI input in Mouse Keyboard.

9 Distribution

Bome's Mouse Keyboard may be distributed freely in the form of the original distribution file. Please contact Bome Software if you write a review of Bome's Mouse Keyboard. Magazines are encouraged to purchase a special license entitling them to publish the full, registered, version of Mouse Keyboard with the magazine. Please contact Bome Software to get a quote.

10 Contact

Post your comments, proposals, questions in the Discussion Forums. Bome staff as well as other Mouse Keyboard users will provide quick and qualified responses. For any other inquiries, go to the Bome Software contact form.

11 Updates

Latest updates can be downloaded at the web site. There you can also subscribe to the announcement mailing list, which will keep you up to date of further developments.

12 What's New

See the Version History.

13 Thanks

Many people have made proposals and bug reports. Among these are Sören Åslund, Falk, Jason Hensley, Sascha Kamps, Moohman, Felix Petrescu, Christian Segui. Thank you very much ! When you feel your name is missing here, send me an email.

Appendix A: Patch script specification

A.1 Cubase (SX) Patch Name Script Documentation

The scripts in this folder and sub folders are simple text files that define the names associated with MIDI programs and MIDI banks. You can create your own scripts with a text editor and the specification below. Or you can find tools on the Internet for ease of use.

Also, scripts for many devices can be downloaded by using search terms like *Cubase patch name script download*.

The patch scripts included with Bome's Mouse Keyboard, and the following specification, is copied from the installation of Cubase SX, with permission from Steinberg.

A.2 Concept

The program names are associated with MIDI messages Program Change and Bank select MSB and LSB.
The program names appear as a popup menu with (optional, if group level >1) submenues.
The program names can be put into groups of different levels.
Root level is 1.
Programs defined at level one can be selected direct from the main popup menu.
Higher levels can only be accessed via submenues.
It is allowed to define program names more than once, for flexible menu architecture.
The defined key names appear, when the drum edit is opened on a midi part, which is set to a program that has defined key names.
One script could contain different modes, which can be associated to midi channels (for cases like: drumchannel, FX channel, performance channel, ...).
The first defined mode without channel ("[mode]%s") is the default mode for all channels, which does not have a specific channel mode ("[mode%n]%s").
The script can be changed during runtime and will be updated, when the user wants to select a sound of that script.

A.3 Commands

only 1 command per line!
`%n`: means number (unsigned integer beginning with 0)
`%L`: means group level (unsigned integer beginning with 1)
`%s`: means string (ends at line end)

>>comments
```
[comment]%s                        //can be inserted anywhere
```

>>header
```
[cubase parse file]                //first command
[parser version %n]                //second command. current %n is
                                   //      0001
[creators first name]%s            //used to identify the script
[creators last name]%s             //used to identify the script
[device manufacturer]%s            //used to identify the script
[device name]%s                    //used to identify the script
[script name]%s                    //used to identify the script
                                   //      and is displayed
[script version]%s                 //used to identify the script
```

>>optional settings
```
[bank: do swap value bytes]        //swap MSB and LSB ("Roland"
                                   //      style)
[bank: don't swap value bytes]
[bank: do send msb first]
[bank: don't send msb first]
[bank: do send only one byte]      //send only first byte (interacts
                                   //with [bank: do send msb first])
[bank: don't send only one byte]
[bank: do send as program change]  //send bank select as several
                                   //      Program changes
[bank: don't send as program change]
```

>>name definition bulk
```
[define patchnames]                //start of name definition bulk
[mode]%s                           //define topmost grouplevel like
                                   //  "XG Voices" and "XG Drumkits"
                                   //  to distinguish different modes
                                   //  of a single device
[mode%n]%s                         //as above, but mode is associated
                                   //  with MIDI channel %n [1 - 16]
                                   //  eg: [mode 10] XG Drumkits
[g%L]%s                            //define group:
                                   //  %L level [1 - 9],
                                   //  %s name
[sorted]                           //following program name entries
                                   //  in that group will be shown
                                   //  sorted alphabetically
[p%L,%n,%n,%n]%s                   //define program name:
                                   //  %L level [1 - 9],
                                   //  %n program change [0 - 127],
                                   //  %n bank MSB [0 - 127],
                                   //  %n bank LSB [0 - 127],
                                   //  %s program name
```

```
[p%L,%n,%n]%s                    //define program name:
                                 //  %L level [1 - 9],
                                 //  %n [0 - 127] program change,
                                 //  %n bank (14 bit) [0 - 16383],
                                 //  %s program name
[k%n]%s                          //define key name (for the
                                 //  previously defined program):
                                 //      %n key number [0 - 127],
                                 //      %s key name
```

>>end of script
```
[end]                            //last command
```