



USER MANUAL



Applied Acoustics Systems

Information in this manual is subject to change without notice and does not represent a commitment on the part of Applied Acoustics Systems DVM Inc. The software described in this manual is furnished under a license agreement. The software may be used only in accordance of the terms of this license agreement. It is against the law to copy this software on any medium except as specifically allowed in the license agreement. No part of this manual may be copied, photocopied, reproduced, translated, distributed or converted to any electronic or machine-readable form in whole or in part without prior written approval of Applied Acoustics Systems DVM Inc.

Copyright ©2015 Applied Acoustics Systems DVM Inc. All rights reserved. Printed in Canada.

Program Copyright ©2008-2015 Applied Acoustics Systems, Inc. All right reserved.

Strum GS is a Trademark of Applied Acoustics Systems DVM Inc. Windows and Windows Vista are registered trademarks of Microsoft Corporation in the United States and other countries. Mac OS and Audio Units are registered trademarks of Apple Corporation. VST Instruments and ASIO are trademarks of Steinberg Soft Und Hardware GmbH. RTAS and AAX are registered trademarks of Avid Technology Inc. All other product and company names are either trademarks or registered trademarks of their respective owner. Unauthorized copying, renting or lending of the software is strictly prohibited.

Visit Applied Acoustics Systems DVM Inc. on the World Wide Web at

www.applied-acoustics.com

Contents

1	Introduction	7
1.1	System Requirements	7
1.2	Installation and Authorization	8
1.3	Getting Started	9
1.3.1	Using <i>Strum Session</i> in Standalone Mode	9
1.3.2	Exploring the Factory Sounds	9
1.3.3	Using <i>Strum Session</i> as a Plug-in	10
1.4	Getting Help	10
1.5	About this Manual	10
2	An Overview of <i>Strum Session</i>	11
2.1	Playing Mode	11
2.2	Chord Detection and Voicing	11
2.3	Strumming	12
2.4	Signal Flow	12
2.5	Interface	13
3	Playing Strum GS-2	16
3.1	Keyboard Mode	16
3.1.1	Hammer-On and Pull-Off	16
3.2	Guitar Mode	16
3.2.1	Chord Keys	17
3.2.2	Strumming Keys	17
3.2.3	Guitar Mode with Auto-Strum	20
3.2.4	Chord Change	21
3.3	Loop Mode	22
3.3.1	Adding Expressivity to Loops	23
3.3.2	Creating MIDI Loops	23
3.3.3	Organisation of MIDI Loops on disk	24
3.4	Chord Voicing	24
3.5	Specific Techniques	26

3.5.1	Arpeggios	26
3.5.2	Trills	27
3.5.3	Muted Strum	27
3.5.4	Partial Strumming	27
3.5.5	Bass & Chords	28
3.5.6	Stationary Bend	28
4	Bank and Program Management	30
4.1	Banks and Programs	30
4.2	Saving Programs	30
4.3	The Bank Manager	30
4.4	Using MIDI Bank and Program Changes	32
4.5	Backups of Banks and Programs	32
4.6	Exchanging Banks and Programs	32
4.7	Restoring the Factory Library	33
5	Parameters	34
5.1	General Functioning of the Interface	34
5.1.1	Knobs	34
5.1.2	Switches	34
5.1.3	Drop-down Menus	34
5.1.4	Modulation Signals	34
5.1.5	Synchronisation	35
5.2	The Parameters	35
5.2.1	Compressor	36
5.2.2	EQ	36
5.2.3	Distortion	37
5.2.4	Amplifier	38
5.2.5	Chorus	39
5.2.6	Tremolo	39
5.2.7	Delay	40
5.2.8	Reverb	40
5.2.9	The Pickups Module	40

5.2.10	The Play Mode	40
5.2.11	Pitch and Modulation Wheel	41
5.2.12	The Clock Module	41
5.2.13	Strumming	42
5.2.14	Chord Display	42
5.2.15	Chord Voicing	43
5.2.16	Loops Module	43
5.2.17	Keyboard	44
6	Utility Section	45
6.1	The MIDI LED and MIDI configuration	45
6.2	Tuning	45
6.3	History and Compare	45
6.4	Volume	46
6.5	Level Meter	46
6.6	The About Box	46
7	Audio and MIDI Settings	47
7.1	Audio Configuration	47
7.1.1	Selecting an Audio Device	47
7.1.2	Latency	47
7.2	MIDI Configuration	48
7.2.1	Selecting a MIDI Device	48
7.2.2	Creating MIDI Links	48
7.2.3	Creating a default MIDI Map	48
7.2.4	MIDI Program Changes	49
7.2.5	MIDI Bank Changes	49
7.2.6	Pitch bend	49
7.2.7	Aftertouch	49
7.2.8	Live MIDI Loop Control	50
7.2.9	Modulation wheel	50

8	Using <i>Strum Session</i> as a Plug-In	51
8.1	Audio and MIDI Configuration	51
8.2	Automation	51
8.3	Multiple Instances	51
8.4	Saving Projects	51
8.5	Performance	51
9	Appendix - Lists of Chords Detected by <i>Strum Session</i>	52
10	License Agreement	59

1 Introduction

Strum Session is a synthesizer dedicated to the emulation of acoustic and electric guitars. The synthesizer is entirely based on the AAS physical modeling technology and uses no sampling nor wave tables. Instead it produces sound by solving, on the fly, mathematical equations modeling the different components of a guitar and how they interact. This elaborate synthesis engine responds dynamically to the control signals it receives while you play thereby reproducing the richness and responsiveness of real instruments.

Strum Session features elaborate modeling of the vibrations of the strings, pick or finger action, the body of the guitar, pickups, a two-channel amplifier, speaker cabinet, spring reverb and various effects. *Strum Session* also includes many functionalities which make it easy to reproduce the playing techniques of a guitarist on a keyboard. *Strum Session* includes a chord detection module and performs automatic voicing of chords. In other words, you play the chords you know on the keyboard and *Strum Session* will voice them automatically, for different musical styles, as a guitarist would have played them. The picking-hand technique of a guitarist is reproduced by an auto-strum function, special strumming keys or using standard MIDI loops.

Before discussing the synthesizer in more detail, we would like to take this opportunity to thank you for choosing an AAS product. We sincerely hope that this product will bring you inspiration, pleasure and fulfill your creative needs.

1.1 System Requirements

The following minimum computer configuration is necessary to run *Strum Session*:

Mac OS

- Mac OS X 10.11 (El Capitan) or later
- Intel Core i5 (circa 2012), Apple M1 processor or later
- 64-bit DAW

Windows

- Windows 10 64-bit or later
- Intel Core i5 (circa 2015) or later
- 64-bit DAW

Keep in mind that the computational power required by *Strum Session* depends on the number of voices of polyphony and the sampling rate used. These computer configurations will enable you to play the factory sounds with a reasonable number of voices but performances will vary depending on your specific computer configuration.

1.2 Installation and Authorization

Installation and authorization of *Strum Session* is quick and easy. For the installation of our different products we use so-called *custom installers* which include both the program itself and your licence information. Installation and authorization can therefore be carried out automatically in a single step and from a single file when your computer is online. AAS products use a copy protection system based on a proprietary challenge/response key exchange and therefore their authorization does not rely on other third party software and/or hardware.

In order to start the installation process, simply double-click on the installer file that you have downloaded. This will first install the program and then use the licence information included in the custom installer file to carry out automatically the challenge/response procedure.

Once the installation is completed, you can check your licence information by starting the program and clicking on the chevron icon at the top of the interface. This will open a dialog box in which you should see your serial number and the email address which you used in order to get the installer file. Note that your serial number is also sent to you by email when your custom installer is created.

If your computer is offline when running the installer, or if the authorization procedure could not be completed for another reason, the dialog box will not show your serial number and you will be prompted to authorize the program. In that case, click on the *Authorize* button and follow the on-screen instructions. Note that it is possible to use the program during 15 days before completing the authorization process. After that period, the program will not function unless it is authorized.

1.3 Getting Started

1.3.1 Using *Strum Session* in Standalone Mode

Strum Session comes with a standalone versions allowing you to play it without having to open your sequencer. This can be convenient to explore *Strum Session* and its library, play it live or do some sound design work. To start *Strum Session* in standalone mode, simply follow the instructions below:

- **Windows** - Double-click on the *Strum Session* icon located on your desktop or select *Strum Session* from the **Start > All Programs >** menu.
- **Mac OS** - Double-click on the *Strum Session* icon located in the Applications folder.

Before you start exploring the program, take a moment to set up your audio and MIDI configuration as explained below.

Audio and MIDI Configuration

Audio and MIDI configuration tools are available by clicking on the **Audio Setup** button located in the lower left corner of the *Strum Session* interface. The **Audio Setup** dialog first allows you to select an audio output device from those available on your computer. Multi-channel interfaces will have their outputs listed as stereo pairs.

On Windows, the audio output list is organized by driver type. The device type is first selected from the *Audio Device Type* drop-down list. If you have ASIO drivers available, these should be selected for optimum performance. The **Configure Audio Device button** allows you to open the manufacturer's setup program for your audio interface when available.

Once the audio input has been selected, you can then select a sampling rate and a buffer size from those offered by your audio interface.

The list of available MIDI inputs appears at the bottom of the dialog. Click on the checkbox corresponding to any of the inputs you wish to use.

1.3.2 Exploring the Factory Sounds

Strum Session comes with a wide range of factory programs right out of the box which amounts to a huge range of sounds before you have even turned a single knob. As you would expect, the best way of coming to grips with the possibilities *Strum Session* offers is simply to go through the programs one at a time.

Strum Session uses the notions of *Banks* and *Programs* to organize and classify sounds. A program or preset is a stored set of parameters corresponding to a given sound. The programs are grouped and organized in banks.

The name of the currently loaded bank and program are displayed at the top of the interface. One navigates among the different banks and programs by using the arrows in each of the corresponding boxes or by opening the associated drop-down menu by clicking inside these boxes. Banks and programs are managed using the *Bank Manager* which is revealed by clicking on the *Manage* button appearing above the right-top corner of the *Bank* box. Playing programs and organizing them is pretty straightforward, please refer to Chapter 4 for a complete description of the bank and program management operations.

1.3.3 Using *Strum Session* as a Plug-in

Strum Session integrates seamlessly into the industry's most popular multi-track recording and sequencing environments as a virtual instrument plug-in. *Strum Session* works as any other plug-in

in these environments so we recommend that you refer to your sequencer documentation in case you have problems running *Strum Session* as a plug-in. Note that in plug-in mode the audio and MIDI inputs, sampling rate, and buffer size are determined by the host sequencer.

1.4 Getting Help

AAS technical support representatives are on hand from Monday to Friday, 9am to 6pm EST. Whether you have a question on *Strum Session*, or need a hand getting it up and running as a plug-in in your favorite sequencer, we are here to help. Contact us by phone or email at:

- North America Toll Free: 1-888-441-8277
- Worldwide: 1-514-871-8100
- Email: support@applied-acoustics.com

Our online support pages contain downloads of the most recent product updates, and answers to frequently asked questions on all AAS products. The support pages are located at:

1.5 About this Manual

Throughout this manual, the following conventions are used:

- Bold characters are used to name modules, commands and menu names.
- Italic characters are used to name controls on the interface.
- Windows and Mac OS keyboard shortcuts are written as Windows shortcut/Mac OS shortcut.

2 An Overview of *Strum Session*

Strum Session is a synthesizer which was designed with the goal of enabling keyboard players to easily create realistic guitar tracks. This is a task which is usually difficult to achieve due to the very different nature of these two types of instruments. *Strum Session* was therefore designed to reproduce not only the sounding mechanisms of a guitar but also the main elements of the playing technique of a guitar player.

Very generally playing on a guitar can be described in terms of fretting hand (usually left hand) and picking hand (usually right hand) techniques. The fretting hand is used to fix the chords or notes played on the different strings of the guitar while the picking hand is used to set the different strings into vibration and therefore play melodies or strumming patterns. In *Strum Session*, these functions have been integrated into a chord detection, voicing, and strumming module.

2.1 Playing Mode

In practice, *Strum Session* can be played using one of three available modes: *Keyboard*, *Guitar*, or *Loop* mode.

In *Keyboard* mode, *Strum Session* reproduces what is played on the keyboard as for any other keyboard instrument without trying to voice what is being played. The range of the instrument is then limited to the natural range of a guitar as the notes still need to be played on the modeled strings on which *Strum Session* is based.

In *Guitar* mode, *Strum Session* voices chords played with the fretting hand and reproduces picking hand technique using special strumming keys as described below.

Loop mode is similar to *Guitar* mode except that strumming and picking patterns are played from recorded MIDI files.

2.2 Chord Detection and Voicing

Chords played on a keyboard and a guitar share the same name but are played or voiced differently. Because of the tuning of the strings on a guitar, the notes of the chord are not usually played in the same order as on a keyboard and the range of the notes is usually different, notes may also be repeated. Furthermore, the same chords can be played in different positions on the guitar depending on the playing or musical style. This specific way of playing chords on a guitar is very characteristic of the tone and sound of the instrument. It is therefore very important, in order to obtain a realistic guitar sound, to play chords as a guitar player would rather than how they are played on the keyboard.

In *Guitar* or *Loop* mode, the voicing of chords is performed in two steps by *Strum Session*. A chord played on the keyboard is first interpreted by the chord detection module in order to determine which chord was played on the keyboard. *Strum Session* then finds a guitar voicing

corresponding to this chord. In other words, *Strum Session* finds how a guitar player would have played this particular chord.

When chords are played on the keyboard, the order of the notes above the root is not taken into account by the chord detection engine. This implies that you can play the chords as you know them on the keyboard without having to know or learn special voicings used by guitar players. For example, you could play chords in close position with the left or right hand, you could also play the root with the left hand and the rest of the chord with the right hand or play the chords with the notes spread across both hands. *Strum Session* will take care of finding the right voicing and playing position on the guitar depending on the settings of the chord type and playing position controls as described in section 3.4

The name of the chord detected as well as the specific voicing chosen by *Strum Session* is displayed in the fretboard located in the lower portion of the graphical interface of the *Play* view. Note that the voicing chosen by *Strum Session* varies depending on the chord type chosen by the user as is described in Section 3.4. *Strum Session* will try to detect a chord as soon as three or more keys are depressed on the keyboard. Please refer to section 9 for a list of the chords detected by *Strum Session*.

2.3 Strumming

On a guitar, notes and chords are triggered by the action of the picking hand (usually right hand). The resulting sound is very typical of a guitar as the guitarist triggers the strings sequentially, more or less rapidly, with an up and down motion of the hand. Strings can also be played individually, in different order, to create arpeggio patterns. Different sonorities can be obtained by damping or muting the strings either by applying the picking hand on the strings near the bridge (palm muting) or releasing the pressure on the notes played with the fretting hand (scratching) while strumming. Using these different types of hand motions and techniques, the guitarist can create complex melodic and rhythmic patterns. In *Strum Session*, these different effects can be achieved through a strumming module which is controlled by special strumming keys or MIDI loops as will be explained in more details in Chapter 3 and 5.

2.4 Signal Flow

The general signal flow of *Strum GS* is presented in Figure 1 and illustrates schematically how the different modules in *Strum GS* interact. From left to right, the synthesizer first includes a chord detection module which parses the MIDI signal it receives and determines the chords played on the keyboard. This information is then sent to the voicing engine which, taking into account how a guitarist would actually play the different chords, determines which notes are played on the different strings of the guitar. The corresponding information, for each of the six strings of the instrument, is then sent to a triggering or strumming module which generates an excitation signal for each of the individual string. This is the signal received by the synthesis or string module which then simulates the vibration signal of the individual strings of the guitar. The output of this

module corresponds to the signal that would actually be measured at the bridge of the instrument and which is transmitted to the body of the instrument. The body module completes the instruments and simulates how sound is radiated both from the top-plate of the body (or soundboard) and the air cavity of the body. Finally, in addition to the modules reproducing the guitar itself, a multi-effect processor has been included to allow further shaping of the sound radiated by the instrument. The multi-effect processor comprises an equalizer, a compressor, two effects, and a reverb.

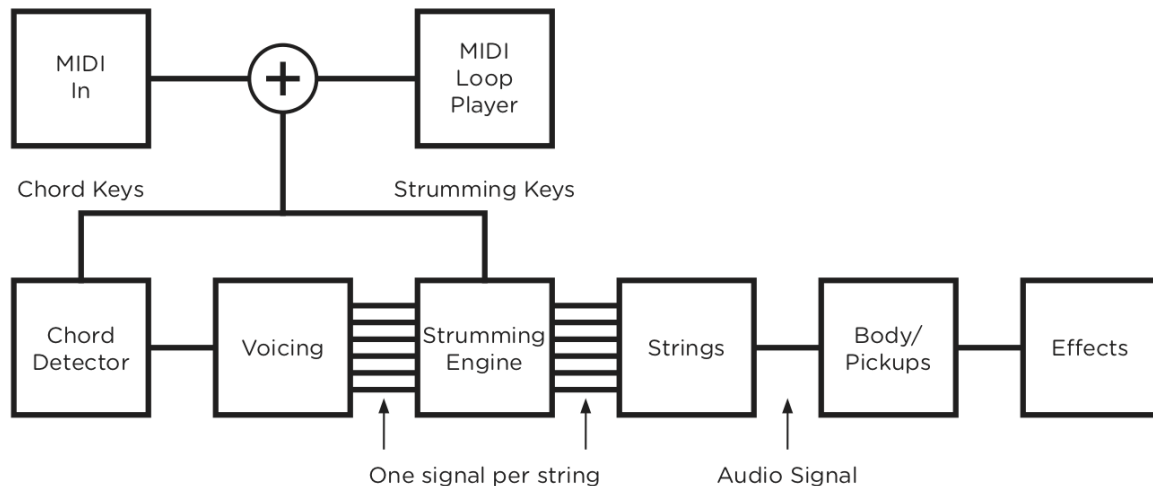


Figure 1: General signal flow of *Strum GS*.

It is important to note how the MIDI signal received by the synthesizer is parsed. In *Guitar* and *Loop* mode, signal received from the MIDI input or MIDI player module, is interpreted in terms of the fretting hand (usually left hand) and picking hand (usually right hand) of a guitar player. MIDI notes with number ranging between E two octaves below middle C (MIDI note number 40) and C one octave above middle C (MIDI note number 72) are associated with the fretting hand and their corresponding signal is sent to the chord detector. Notes in the octave just above (MIDI note number ranging between 72 and 84) are special strumming keys, used to trigger different picking hand techniques, and are therefore used to control the strumming engine. The use of these strumming keys is explained in more details in section 3.2.2.

In the case of the electric guitar, in addition to the instrument itself, the amplifier and cabinet are fundamental to the production of sound. A two-channel amp, with cabinet and spring reverb has therefore been added to the signal flow between the two effects of the multi-effect processor as show in Figure 2.

2.5 Interface

The interface for acoustic and electric guitars is different and changed accordingly when selecting an acoustic or electric guitar preset. While the interface changes when switching from one type of



Figure 2: Position of the amplifier in the signal chain in the case of the electric guitar.

guitar to the other, the general functioning is similar regardless of the type of guitar used. In the *Play* view the EQ module appearing with acoustic guitars is replaced by an amplifier module in the case of the electric guitar.

The lower section of interface includes a playing mode selector, a master clock, a chord display, a loop player as well as modules controlling strumming and voicing and vibrato which will be described in more details in Chapter 5.

On the left of these parameters, one finds a pitch bend wheel and a modulation wheel. The modulation wheel is normally used to control the amount of vibrato in the sound but it can also be used to adjust any other parameter through MIDI links which will be described in Chapter 7. Just below is a clickable five octave keyboard giving information on the keyboard layout specific to each playing mode and allowing one to play different notes which can be useful when no MIDI keyboard is connected to the computer. The electric guitar presets also include a pick-up switch that allows one to listen to signal from the neck or bridge pickup.

The middle section of this view allows one to turn the effects from the multi-effects module, compression and equalizer *on* and *off* and to rapidly adjust their main parameters.



Figure 3: The Strum Session interface for the acoustic guitar.



Figure 4: The Strum Session interface for the electric guitar.

The size of the graphical interface can be adjusted by click-dragging the handle in its lower right corner. While resizing the interface the zoom factor is displayed in the upper left corner of the interface. This zoom factor can always be displayed by positioning the mouse on this corner. When clicking on the zoom factor, a drop-down menu with specific size ratios is displayed. Selecting a value resizes the interface automatically to that ratio. Note that this resize feature is only available for 64-bit versions of the program.

3 Playing Strum GS-2

Strum Session can be played in three different modes: *Keyboard*, *Guitar*, or *Loop* mode. The playing mode is selected by clicking on the corresponding button in the *Play Mode* box in the lower left part of the *Play* view of the interface. In this chapter, we review how these different modes work and describe the special keyboard layout each of them uses. We also review the different options that affect chord voicing in *Guitar* and *Loop* mode and discuss how to obtain specific guitar techniques.

3.1 Keyboard Mode

The *Keyboard* mode is the simplest way to play *Strum Session*. In this configuration, *Strum Session* basically plays notes on the guitar as they are played on the keyboard. In other words, *Strum Session* does not attempt to voice notes or chords played on the keyboard. The main difference with a traditional keyboard instrument is the range of the instrument. *Strum Session* indeed needs to play the notes on the different strings of the modeled guitar so will only react to notes in the normal range of the guitar that is from the second E below middle C (MIDI note number 40) to the third A above middle C (MIDI note number 93).

The *Keyboard* mode is the only one that allows one to play melodies and intervals, it is therefore used for lead guitar playing. It also enables one to play chords without voicing from *Strum Session* and therefore to control exactly how they are played.

3.1.1 Hammer-On and Pull-Off

In *Keyboard* mode, hammer-ons and pull-offs are automatically triggered by playing legato notes spaced by one or two semi-tones. Hammer-on and pull-off are playing techniques used by guitar players to play legato or grace notes. The hammer-on effect is obtained by first picking a note and then hammering down another finger onto the same string at a higher fret. The pull-off effect is almost the opposite of the hammer on. It is obtained by first picking a note and then sharply pulling-off the finger from the fretboard in order to hear a second fretted note on the same string. The sounds produced using these techniques are softer and less percussive than the ones produced by picking the notes.

3.2 Guitar Mode

In this mode, the MIDI signal received by *Strum Session* is interpreted in terms of the fretting hand (usually left hand) and picking hand (usually right hand) of a guitar player. The range of the MIDI keyboard has therefore been divided into two sections, the *Chord keys* section and the *Strumming Keys* section as illustrated in Figure 5.



Figure 5: Sections of the MIDI keyboard as used by *Strum Session* in *Guitar* mode. The C with the dot is middle C

3.2.1 Chord Keys

This section is associated with the fretting hand of the guitarist and includes notes ranging from the second E below middle C (MIDI note number 40) to B one octaves above middle C (MIDI note number 71). This section of the keyboard is where chords are played and it is the MIDI signal from this section which is sent to the chord detection and voicing modules as explained in section 2.4. *Strum Session* will try to detect a chord and find a corresponding voicing as soon as three or more key are depressed on the keyboard. Note that the exact guitar voicing chosen by *Strum Session* depends on the setting of the Voicing module as will be explained in section 3.4.

Automatic Chords

In *Guitar* mode, chords are always played by *Strum Session* when a single note or special configuration of two or three notes are played in the *Chord Keys* section of the keyboard making it very easy to create strumming patterns. When only a single note is played, a major chord is automatically constructed using this note as the chord root. A minor chord is created by playing a root note and the first black key to its left. A seventh chord is played by depressing a root note and the first white key to its left. A minor seventh chord is obtained by playing a root note and the first black and white key to its left. These special rules are illustrated in Figure 6 with the key of C as an example. Note that other two note intervals are ignored and a major chord is played using the first note of the interval which was played as the root of the chord.

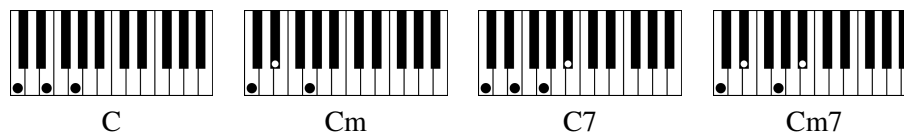


Figure 6: Automatic chords. The key of C is used as an example but all keys follow the same rules.

3.2.2 Strumming Keys

The twelve notes from the C note located one octave above middle C (MIDI note numbers 72 to 83) are interpreted as special strumming keys and are used to trigger different picking hand techniques. Note that these special strumming keys appear in a mirror configuration starting on C located three octaves below middle C allowing one to trigger them with the left hand instead of the right hand.

Depending on the strumming keys used one can trigger downstrokes, upstrokes as well as muffled downstrokes and upstrokes and palm muted downstrokes. Strings can also be triggered individually to play arpeggio patterns. The effect of the different strumming keys, illustrated in Figure 7, is now reviewed.

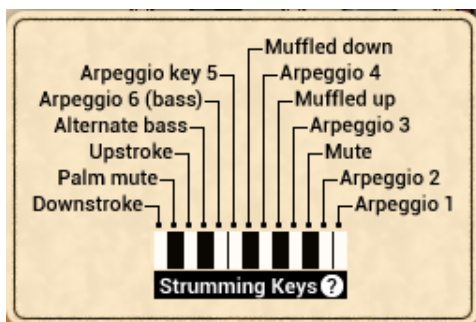


Figure 7: Strumming keys as used by *Strum Session* in *Guitar* mode.

Downstroke and Upstroke

A downstroke is achieved by dragging a pick (or the fingers) across the strings of the guitar in a downward motion. Notes are therefore played from lowest to highest. An upstroke is obtained in the same manner but dragging the pick or fingers upwards from the highest to the lowest string. A downstroke is triggered by depressing C while an upstroke is triggered by depressing D one octave above middle C (MIDI note 72 and 74). These strokes represent the simplest way to play chords on a guitar and can be played alternately. Note that when strings are not used in a specific voicing (strings with no dot on the chord display), they are not played when using a downstroke or an upstroke. If no note is played on the keyboard, the downstroke and upstroke keys trigger a strum but the strings are muted. The effect is the same as using the muffled down- and upstroke keys which are described below



Figure 8: Downstroke and upstroke on a C chord with the strumming keys.

Palm Muting

Palm muting is a technique which consists in partially muting the strings with the help of the picking hand by letting it lightly touch the strings near the bridge. The pick (or fingers) is then dragged across the strings in a downward or upward motion. In *Strum Session*, the palm muted downstroke is obtained by depressing C# one octaves above middle C (MIDI note number 73).



Figure 9: Alternating downstroke and palm muting.

Muffled Downstroke and Upstroke

On a guitar, fretted notes can be muted by decreasing the pressure applied by the fretting fingers on the strings. The guitarist can control the effect by applying more or less pressure on the strings and obtain a percussive effect by striking the strings with the picking hand. This effect is also known as scratching. In *Strum Session*, the muffled downstroke and upstroke are obtained by depressing F# and G# one octave above middle C (MIDI note number 78 and 80) respectively. Using these keys has the same effect as using the down- and upstroke keys without playing a chord on the keyboard, but these strumming keys allow one to obtain the effect without releasing the chord.



Figure 10: Scratching with the strumming keys.

Mute All

A muted strum is a stroke which is immediately muted by applying the full hand on all the strings in order to completely stop the sound. In *Strum Session*, this effect is reproduced by depressing

B \flat one octave above middle C (MIDI note number 82).

Playing Individual Strings

The strings of the guitar can be played individually enabling one to arpeggiate or create finger picking patterns. Individual strings are triggered by depressing notes corresponding to white keys between E one octave middle C and C two octaves above middle C (MIDI notes 76, 77, 79, 81, 83 and 84, arpeggio keys 6 to 1).

In chords using the six guitar strings, these arpeggio keys (number 6 to 1) correspond to the individual strings from string 6 (lowest E) to string 1 (highest E). When some voicings use less than six strings, the arpeggio keys corresponding to the unused strings repeat the note of the following or preceding string depending if it is the lowest or highest note of the chord. For example for a chord using strings 5 to 1, the lowest note on string 5 will be repeated on the arpeggio key 6 (E). For a chord using strings 6 to 2, the highest note of the chord from string 2 will be repeated on the arpeggio key 1 (second C in the strumming key section). Exceptionally for chords played on four strings, string 4 will be repeated on the arpeggio keys 6 and 5 (E and F), string 1 will be repeated on the arpeggio keys 2 and 1 (B and C) while strings 2 and 3 will be moved to the arpeggio keys 3 and 4 (G and A). Indeed, this configuration ensures that the lowest note of the chord will appear on arpeggio key 6 and that arpeggio keys 4, 3, and 2 will always play a different note which is convenient to play an arpeggio patten in any chord voicing type.

Alternate Bass

In certain musical styles, such as country and bluegrass, one often finds rhythm-guitar patterns obtained by switching the root or fifth played on the bass string and a strum on the remaining strings. To easily recreate these effects, *Strum Session* maps the fifth or alternate bass on the E \flat key (MIDI note number 75) located one octave above middle C. When there is no fifth in the chord, the root of the chord is instead mapped to this key.

3.2.3 Guitar Mode with Auto-Strum

When the *Auto-Strum* mode is selected, strumming is triggered as notes or chords are played on the keyboard without having to use the strumming keys. This mode is switched *on* or *off* by clicking on the *Auto* button in the *Strumming* section of the interface. In *Auto-Strum* mode, chord are strummed using a downstroke or in other words, notes are played from lowest to highest.

When the *Auto-Strum* mode is switched *off*, notes and chords are processed by the chord detection and voicing modules but the strings are not triggered. In order to play them, one must use the different strumming keys as described in section 3.2.2. As soon as a strumming key is depressed, the strings are triggered. Note that strumming keys are always active whether the *Auto-Strum* mode is switched *on* or *off*.

Strum Down

Play a chord in the *Chord Keys* section of the keyboard as shown in Figure 11. The chord is detected by *Strum Session* and voiced on the guitar fretboard. This triggers a downstroke from the strumming module and consequently the strings are played from the lowest to the highest. In order to trigger a new downstroke without muting the strings, just release one note from the chord and play it again as shown in Figure 12.



Figure 11: A simple strumming sequence.

3.2.4 Chord Change



Figure 12: Re-triggering a downstroke.

Once a chord has been played, the chord detection, voicing and strumming modules are only triggered when a new note or a strumming key is played; releasing notes from a chord has no effect. Consequently, *Strum Session* holds the current chord as long as all its notes are not released. This can be used to make rapid chord changes. For example, it is possible to switch from a C chord to a G chord without muting the strings by playing C-E-G and then by replacing C and E by B and D without releasing the G as shown in Figure 13. The same effect can be obtained with chords which do not share common notes by using the sustain pedal.



Figure 13: Smooth chord transition.

On the other hand, if one wishes to remove a note from a chord, for example to switch from a C7 chord (C-E-G-B \flat) to a C chord (C-E-G), it is not sufficient to release the B \flat key. It is necessary, in order to trigger again the chord detection module, not only to release the B \flat from the chord but

also to depress a strumming key or to release another note from the chord, for example the G, and replay it when one wants to strum the new chord.

Chords and Change of Pitch

Playing chords on higher or lower notes on the keyboard does not affect where *Strum Session* plays them on the fretboard as it always tries to play chords on a position as low as possible. *Strum Session*, however, will take into account the fret position as set by the *Position* parameter of the **Voicing** module and play the chords on this fret or above whenever possible. The only exception to this is when drop chords are used in which case *Strum Session* will try to follow chord position as closely as possible.

3.3 Loop Mode

Reproducing complex strumming patterns can rapidly become complicated and requires to be a good keyboard player. The good news is, however, that *Strum Session* is supplied with a library of MIDI loops which you can use to easily start creating a new piece. A MIDI loop is a recording of strumming patterns created using the special strumming keys available in *Guitar* mode as described in section 3.2. These patterns are automatically applied to chords played on the keyboard.

In *Loop* mode, the MIDI keyboard is divided into a *Chord Keys* section where chords played by the fretting hands are played and a *Loop Keys* section, as illustrated in Figure 14, from which MIDI loops reproducing strumming key patterns are triggered. Note that the automatic voicing of chords in *Loop* mode is done in the same way as when playing in *Guitar* mode.

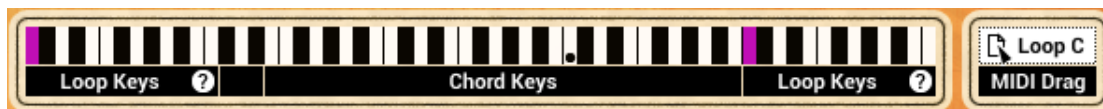


Figure 14: Sections of the MIDI keyboard as used by *Strum Session* in *Loop* mode. The C with the dot is middle C

Each *Strum Session* preset is saved with a set of seven loops called a *loop pack*. The seven white notes in the *Loops Keys* section of the keyboard, from C to B located one octave above middle C, are used to trigger one of the seven loops of a loop packs as shown in Figure 15. The same *Loop Keys* also appear in mirror configuration starting on the C key located 3 octaves below middle C. Loop packs can be loaded from the **Loops** module from the *Play* view as will be explained in section 5.2.16. The currently selected loop is indicated by the purple key in the *Loop Keys* section of the keyboard.

Once triggered, loops play until the notes held in the *Chord Keys* sections are released or if the B \flat or second C of the *Loop Keys* section is depressed. The difference between these two keys is that the B \flat , *Mute* key mutes all strings and the sound stops as soon as the key is depressed. When the C or *Stop* key is used, the loop stops but the sound is allowed to decay naturally.

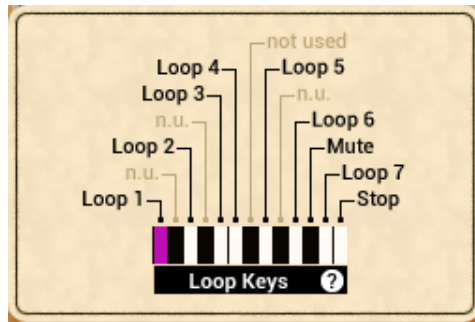


Figure 15: Strumming keys as used by *Strum Session* in *Loop* mode.

3.3.1 Adding Expressivity to Loops

In order to make MIDI loops more expressive, *Strum Session* allows one to link two different MIDI controllers to some performance parameters such as playing softer or louder, slower or faster, increase or decrease the strumming span and putting emphasis on lower or higher strings when strumming. To access this modulation matrix, open the MIDI configuration window by clicking on the **MIDI** button located just below the MIDI led in the top part of the interface and assign effects to controllers in the Live MIDI Loop Control section of the window.

3.3.2 Creating MIDI Loops

Loops for *Strum Session* are easily created in your favorite sequencer by respecting the following rules:

- All the events must be on MIDI channel 1.
- The loop must begin at the start of the file and finish at the end of the track. In other words, if the track lasts for four bars but there are notes only in the first bar, *Strum Session* will still loop over the four bars.
- The tempo and the time signature must be defined at the beginning of the loop and must not change during the rest of the loop. *Strum Session* indeed ignores tempo changes occurring in the middle of a loop.
- The loop must be recorded on disk on a file using MIDI format 0 or 1 and having a .mid extension.

Here are other advices which help to create more realistic loops:

- Always slightly vary the velocity of strumming keys in order to get a more lively result.
- If the loop is short, it may be preferable to repeat the same patterns many time with different velocities for each repetition.

- If the loop includes chords, always make sure that they are well quantized and that there is always a strumming key played at the same time. This will avoid unwanted strokes when the loop is played while the *Auto-Strum* mode is switched *on*.
- Avoid using MIDI controllers whose effect can be changed by the user. For example, the pitch wheel can either be used to create a bend or a slide; the hold pedal can either hold the current chord or be used to trigger an alternate strum.

3.3.3 Organisation of MIDI Loops on disk

Loops are stored at a specific location on disk:

On **Mac OS**:

/Library/Application Support/Applied Acoustics Systems/Strum GS-2/MIDI Library

On **Windows**:

%PROGRAMFILES(X86)%\Applied Acoustics Systems\Strum GS-2\MIDI Library

Clicking on the *Show Files* button on the *Loops* module of the *Play* view opens a Finder or Explorer window (on Mac OS X or Windows respectively) at this specific location. Loops are organised into banks whose name correspond to folder names within the loops folder. These names are those appearing in the *Bank* drop-down menu in the *Loops* module and Loop packs under these categories are stored within these folders. The seven loops in a pack all share the same name except for the letter associated with the note (letters A to B) used to trigger a specific loop which is appended to the loop name.

3.4 Chord Voicing

On the guitar, there are usually many different ways to play the same chord. Each of these positions or voicing sounds differently and suits different musical styles. The specific voicing chosen by *Strum Session* for a chord depends on the *Type* parameter from the *Chord* section of the interface. One can choose between four types of chords:

- **Open Chords.** These are chords played with a combination of fretted notes and open strings. These chords are usually played only within the first three frets of the fretboard. Open chords are easy to play and extensively used when playing folk music. Note that all chords do not necessarily have an open position on the guitar.
- **Movable Chords.** These chords do not use open strings. As a result they can be moved along the fretboard of the guitar allowing one to easily play the same voicing in different tonalities. This type of chords includes barre chords which are obtained by using one finger to press down multiple strings across the fretboard. Movable chords always use the maximum possible number of strings and are therefore useful to play arpeggios. Furthermore they allow one to play the same arpeggio patterns in different keys.

- **Drop Chords.** These are four note chords which allow for fast and subtle movement between chords. The positions used by *Strum Session* are based on drop 2 and drop 3 chords. These chords are obtained by dropping the second or third voice of a chord down one octave. This type of chord sounds lighter and is extensively used in jazz.
- **Powerchords** A power chord, also known as fifth chord, consists in a note and another one a fifth above. In other words it is like a triad without the third. Powerchords are extensively used in rock music especially with highly distorted sounds as including the third usually results in unpleasant frequency components in the distorted sound. They are also easy to play. In powerchord mode, any chord played on the keyboard which contains a natural fifth will be played with 3 notes: the root, the fifth and the octave. If the chord is inverted on its fifth, the power chord played by *Strum* is also inverted and the notes played in the following order: fifth, root and octave. Note that it is also always possible to play powerchords with the other chord types by playing a fifth, root and fifth's octave.

In addition to these voicing categories, the voicing can be made more precise by specifying what should be the lowest note of the chord played by *Strum Session*:

- **Lowest.** The lowest note in the guitar voicing chosen by *Strum Session* is the lowest note played on the keyboard. This type of voicing is useful if it is necessary that the bass of the chord sequence follows a specific movement. Note that for open and movable chords many inversions do not have an interesting voicing on the guitar and in this case, the four note voicing generated by *Strum Session* may sound a bit out of place in a chord progression.
- **Root.** The lowest note in the guitar voicing chosen by *Strum Session* is the root of the detected chord. It is very common for guitar chords to have the root in bass position. This choice of chord type allows one to play chords on the keyboard using any voicing and still obtain a guitar chord in root position.

The *Playing Position* parameter from the *Chord* section allows one to specify the lowest fret on which the lowest note of a chord should be played. This control gives *Strum Session* an indication of the position on the neck where chords should be played and chords are voiced accordingly when possible. This parameter is of course only valid for movable chords and it is therefore inactive when open chords are chosen.

Note that in all chord voicing one finds a fifth even if it was not played on the keyboard which in practice corresponds to what guitar player do. The only exception is for drop chords where only notes played on the keyboard are used in the voicing and therefore may not contain a fifth.

Although there is no general rule and there are many ways to play in different musical styles or obtain different effects, we give some guidelines on chord types and performance settings which should work well in specific situations:

- **Folk.** Open-Root and medium strumming speed.

- **Country and Bluegrass.** Open-Root with a high strumming speed.
- **Pop-Rock.** Movable-Root.
- **Arpeggios with bass motion.** Movable-Lowest or Open-Lowest.
- **Jazz.** Drop chords.
- **Samba Bossa.** Drop chords. Use the arp 6 strumming key to play the bass and arp keys 4, 3 and 2 simultaneously to play the rest of the chord.
- **Funk.** Movable-Root with *Playing Position* on a high fret.
- **Flamenco** Open-Root with a rapid strumming speed. Use the alternate strum on the four highest strings.

3.5 Specific Techniques

We now look at ways to create more elaborate effects. For these examples, it is assumed that we are using the *Guitar* mode and that the *Auto-Strum* mode is switched *off* as we will now be using the strumming keys. This way of playing *Strum Session* involves to think more in terms of a guitar player and it is preferable to play the chords slightly before triggering the strumming action in order to avoid glitches when a strumming key is used before a chord is fully formed on the keyboard . This is just like a guitar player who needs to position the fretting hand on the fretboard before strumming with the picking hand.

3.5.1 Arpeggios

A chord can be arpeggiated using the arpeggio strumming keys (E, F, G, A, B and C). The effect of these keys depends on the chord played by *Strum Session* as explained in section 3.2.2. As many chords include only four strings (a bass and three high notes) it is common for the arpeggio 2 and arpeggio 1 keys to play the same note. Furthermore, depending on the position of the chord on the fretboard, the alternate bass key (arpeggio 5 key), can play a lower, higher or the same note as the bass key (arpeggio 6 key). The most useful arpeggio keys are therefore usually the arpeggio 6, 4, 3 and 2 keys (E, G, A and B). A good position to play arpeggios is to use the thumb to play the bass, the index to play the arp 4 key, the middle finger for the arp 3 key, the ring finger for the arp 2 key and finally the little finger for the arp 1 key. This position is similar to that of a guitar player.

Here is a simple example of an arpeggio. Play a C chord (C-E-G) with the left hand, and then use the right hand to trigger the Bass (E), arp 4 (G), arp 3 (A) and arp 2 (B) keys.

When playing arpeggios, it might be interesting to use *Movable-Lowest* chord types in order to obtain a motion of the bass. To try this, set the *Type* from the *Chord* section to *Movable-Lowest* and play the arpeggio from the preceding example using the following chord progression: C (C-E-G), CMaj7/B (B-C-E-G), C7/B \flat (B \flat -C-E-G) and Am7 (A-C-E-G).



Figure 16: A simple arpeggio.



Figure 17: Bass motion and arpeggios.

3.5.2 Trills

This technique consists in switching rapidly between two notes by using hammer-ons and pull-offs. Trills are available in *Keyboard* mode and are obtained by holding the first note and depressing and releasing the second note. The interval between the notes must be one or two semi-tones.

3.5.3 Muted Strum

A muted strum is a stroke which is muted immediately after having been played by applying the full hand on the strings. It is less dry than a scratch (muffled stroke) and brighter than a palm muted stroke. One way to recreate this effect is to strum a chord using the down- and upstroke keys (C or D) and playing the mute all key (B \flat) very rapidly.

3.5.4 Partial Strumming

A guitarist does not always strum all the strings in a chord and they also often vary the number of strings played between strumming strokes. To some extent, in *Guitar* mode, this is taken into account automatically by *Strum Session* using the MIDI velocity signal received from the keyboard. More control on this parameter, however, may be desired. One way to control exactly the number of strings that are strummed is to use the *Arpeggio* keys instead of the *Downstroke* and *Upstroke* keys. In other words, one strums a chord by depressing a certain number the *Arpeggio* keys together thereby controlling which strings are played in the chord. This can also be used to put emphasis

certain strings and also to introduce some variety and expression in the strumming. Another option is to alternate between this technique and the use of the downstroke and upstroke strumming keys.

Another technique which can be used to obtain partial strums consists in releasing strumming keys before the strum is completed. In this way, the strings still not strummed, are not played. In other words, partial strumming is achieved by playing staccato on the strumming keys while full strums are obtained by playing them normally. This technique is easier to perform when the strumming speed is relatively slow.

When using loops it is possible to vary in real-time the character of the strumming by mapping specific MIDI controllers to the strumming span and to the weight of low and high strings as explained in section 3.3.

3.5.5 Bass & Chords

In certain musical styles, such as country and bluegrass music, one often finds rhythm-guitar patterns obtained by switching between the root or fifth played on the bass string and a strum on the remaining strings. To obtain this effect, it is therefore important that the lowest string is note included in the stroked part of the pattern. In order to reproduce this effect with *Strum Session*, one must first use the bass or alternate bass key (E or Eb) and hold it while playing one of the stroke triggers. Indeed, when *Strum Session* detects that a stroke key is played while the bass or alternate bass key is depressed, it removes this string from the strum.



Figure 18: Bass and Chord.

3.5.6 Stationary Bend

In this technique, the guitarists plays two notes simultaneously and bends one of the two notes. It is possible to obtain the effect with *Strum Session* by taking advantage of the fact that only the last note played is bent.

In *Keyboard* mode, one first plays the note which should not be bent and then the second one. The bending effect is then obtained by moving the pitch wheel or using the aftertouch. This will

only work, however, for intervals larger than one tone because otherwise *Strum Session* always automatically triggers a hammer-on or pull-off when intervals of one or two semi-tones are played legato.

In *Guitar* mode, the last string triggered with an arpeggio key can be bent either by using aftertouch or pitch wheel.

4 Bank and Program Management

Strum Session comes with several factory presets, called *programs*, covering a wide range of sounds. This collection of programs lets you play and familiarize yourself with this synthesizer without having to tweak a single knob. Soon, however, you will be experimenting and creating your own sounds and projects that you will need to archive or exchange with other users. In this section, we review the management of programs.

4.1 Banks and Programs

Sounds are stored in banks containing so-called *programs*. The name of the currently selected bank is shown in the *Bank* drop-down display located at the top of the *Strum Session* interface. The list of available banks is viewed by clicking on the *Bank* display. A bank can be selected by navigating in the list of banks using the left and right-pointing arrows in the display or by clicking on its name when the list of banks is open.

The list of programs included in the currently selected bank can be viewed by clicking on the *Program* display located below the *Bank* display. A program is selected by using the left and right-pointing arrows or by clicking directly on its name. Once a program is selected, the value of the different parameters of the synthesizer are updated and it can then be played.

4.2 Saving Programs

Programs are saved by clicking on the *Save* button located on the top of the *Program* display. When a program has just been loaded, this command is greyed and therefore inactive. It is activated as soon as a parameter of the interface is modified. Clicking on this command replaces the stored version of the program with the new one.

The **Save As** command is activated by clicking on the corresponding button which opens the **Save Program** pop-up window. It is then possible to save the program under a new name or its current one in any of the available program banks. Note that if the original name of the program is used, a new program with the same name will be created at the end of the program list meaning that the original program is not erased. This also implies that it is possible to have many programs with the same name in the same bank.

4.3 The Bank Manager

Banks and Programs can be edited using the **Bank Manager**. The manager window is displayed by clicking on the *Manager* button located above the *Bank* display. It is closed by clicking again on the same button. On the left of the window, one finds the list of banks. Clicking on a bank name fills the list of programs located in the center of the window with the name of these included in the selected bank.

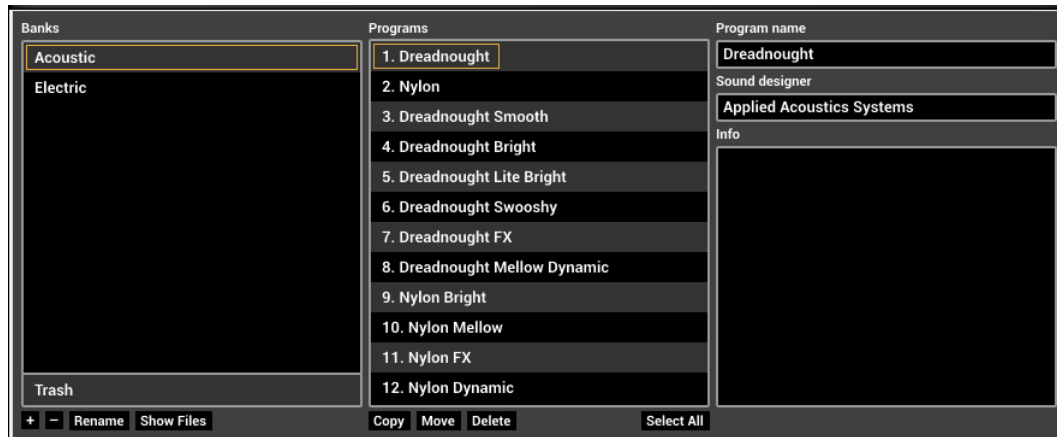


Figure 19: Bank and program manager window.

A new bank can be created by clicking on the + button below the bank list. This opens the **Create New Bank** window in which the name of the new bank can be entered. A bank can be deleted by first selecting it in the bank list and then clicking on the - button. Be careful, this command erases a bank and all the programs it contains; this operation is permanent and can not be undone. In order to rename a bank, simply click on the *Rename* button and enter a new name.

Banks and the information corresponding to each of its programs is stored in a simple text file on your computer hard disk. In order to view these bank files, click on the *Show Files* button under the bank list. On Windows, this command will open an Explorer window at the location where the files are stored. On Mac OSX, the command has a similar effect and opens a Finder window. All the bank file names follow the same format and begin with the bank name. These files can be used for backups or to exchange presets with other users.

The list of programs included in the selected bank is displayed in the program list in the center of the manager window. Presets are selected by clicking on their name which updates the program information appearing on the right of the preset list. Program information includes the name of the preset, its author and comments. This information can be updated by clicking on the corresponding box which opens an edition window. Note that multiple presets can be updated simultaneously by selecting more than one preset at once and clicking on a preset information box.

A multiple selection consisting of adjacent programs is obtained by holding down the *Shift* key on the computer keyboard and then clicking on the name of the first program to be copied and then the last one. A non-adjacent multiple selection is obtained by holding down the *Ctrl/command* computer key and clicking on the name of the different programs to be copied. It is also possible to select all programs at once by clicking on the *Select All* button at the bottom of the program list.

Programs can be copied to another bank by clicking on the *Copy* button. A program must first be selected by clicking on its name on the program list; it is then copied by moving the mouse to a given bank in the *Bank* list on the right and clicking on the bank name. The **Move** command is activated by clicking on the *Move* button; it copies a preset to a new bank but also erases it in the

original bank. A multiple selection of programs can be used with the **Copy** and **Move** commands

Programs can be deleted from a bank by first selecting them and then clicking on the *Delete* button. This will move the programs to a special bank called *Trash* which is located below the regular list of banks. This means that deleted programs can always be recuperated as long as they are not deleted from the *Trash* bank. The content of the *Trash* bank is viewed by clicking on its name; the different programs can then be moved to the other banks as explained above. The *Trash* bank can be emptied by clicking on the *Empty Trash* button which appears below the program list when the *Trash* bank is selected. Be careful as this command can not be undone.

4.4 Using MIDI Bank and Program Changes

Banks and programs can be changed using MIDI bank and program change commands. For more information on how to use these commands, please refer to sections 7.2.4 and 7.2.5.

4.5 Backups of Banks and Programs

User banks are stored on disk as simple text files located in the following folders:

On **Mac OS**:

/Users/[user name]/Library/Application Support/Applied Acoustics Systems/Strum Session 2/Banks

On **Windows**:

%AppData%\Applied Acoustics Systems\Strum Session 2\Banks

The bank files saved by *Strum Session* are named using the following convention:

[name of bank].SS2 Bank

These files contain all the information corresponding to the programs they include. These files can be displayed directly from *Strum Session* by opening the *Bank* manager and clicking on the *Show Files* button. This will open an Explorer or Finder window on Windows or Mac OS respectively at the right location.

The simplest way to create a backup of banks and programs is to make a copy on an external media of the above mentioned folders. Individual banks can be backed-up by making copies of individual bank files.

4.6 Exchanging Banks and Programs

Banks and programs can easily be shared with other *Strum Session* users. This operation simply involves the exchange of the above mentioned user bank files. When a new bank file is copied to the bank folder, it is automatically available to *Strum Session*.

Note that individual programs can not be exported. They always appear inside a bank file. If you only wish to share a few programs, create a new bank, copy the programs you wish to exchange to this bank and share the corresponding bank file.

4.7 Restoring the Factory Library

If necessary, it is possible to restore the original factory library of banks and programs. The original factory bank files are located in the following folders:

On **Mac OS** startup disk:

/Library/Application Support/Applied Acoustics Systems/Strum Session 2/Factory Library

On **Windows 64-bit**:

C:\Program Files (x86)\Applied Acoustics Systems\Strum Session 2\Factory Library

On **Windows 32-bit**:

C:\Program Files\Applied Acoustics Systems\Strum Session 2\Factory Library

Restoring the factory library simply involves copying the files contained in these folders and pasting them in the user bank folders listed in Section 4.5. The user bank folders can be opened directly in an Explorer or Finder window, on Windows and Mac OS respectively, or by using the *Show Files* command directly from the *Strum Session* bank manager.

Note that if you have bank files with the original factory bank names in your user bank folder, they will be replaced by the original factory files. This means that you will lose programs that you would have modified or created in these banks. This operation must therefore be done with caution and it is recommended that you make copies or rename your user banks before proceeding with the restore.

5 Parameters

This section can be used as a reference for the different controls appearing on the *Strum Session* graphical interface. We begin by describing the behavior of the different types of controls appearing on the interface and then describe the parameters of each module of the synthesizer.

5.1 General Functioning of the Interface

5.1.1 Knobs

The synthesizer parameters are adjusted using controls such as knobs, switches and numerical displays. A specific control is selected by clicking on it. A coarse adjustment is obtained by click-holding the parameter and moving the mouse, or the finger on a track pad, either upwards and downwards or leftwards and rightwards. The value of the parameter replaces its label while it is being adjusted. Double clicking on a knob brings it back to its default value when available.

Fine adjustment of a control is obtained by holding down a modifier key of the computer keyboard (Shift, Ctrl, Command or Alt key) while adjusting the parameter. Note that there is one exception to this in the **Strings** module where pressing the Ctrl key on Windows or Command key on Mac OS and turning a knob sets the value for all strings at once.

5.1.2 Switches

Switches are turned *on* or *off* by clicking on them. They are used to activate or deactivate modules and the *sync* feature of some parameters.

5.1.3 Drop-down Menus

Some displays reveal a drop-down menu when clicking on them. Adjustment of the control is obtained by clicking on a selection.

5.1.4 Modulation Signals

The *Velocity* modulation knobs are used to modulate the value of a parameter depending on the velocity signal received from the keyboard so that the value of a parameter increases as notes are played harder on the keyboard. The position of the knob is used to adjust the amount of modulation applied to the parameter. In its leftmost position, the modulation source is turned *off* and the value of the parameter does not vary with the velocity signal from the keyboard. Turning the knob clockwise increases the effect of the modulation signal on the value of the parameter.

5.1.5 Synchronisation

The rate of certain effect modules and the tempo of loops can be synchronized to the clock of a host sequencer or controlled using the built-in **Clock** module. In order to synchronize effect modules, their *Sync* must be turned *on*. Synchronization values for effects are adjusted with the *Sync Rate* parameter and range from 4 whole notes (16 quarter notes) to a thirty-second note (1/8 of a quarter note) where the duration of the whole note is determined by the host sequencer clock. The effect can also be synced to a triplet or dotted note. To adjust this parameter, click on the *Sync Rate* button and choose a rate value from the drop-down menu.

In standalone mode, the master clock is that from the **Clock** module on the *Play* view and the duration of a whole note is adjusted using the *Rate* control of the **Clock** module or the *Tap Tempo* pad. In plug-in mode, the master clock is that of the sequencer when the *Sync to Host* button is switched *On*. When it is in its *off* position, the tempo is controlled by the internal **Clock** module using the *Rate* control.

5.2 The Parameters

We focus in this chapter on the middle and bottom part of the *Strum Session* interface. The top part of the interface includes the bank manager described in Chapter 4 as well as audio and MIDI monitoring functions that will be described in Chapter 7. The middle part of the interface includes an **EQ**, **Compressor**, **Reverb**, **Chorus**, and **Delay** module in the case of the acoustic guitar and a **Compressor**, **Distortion**, **Amplifier**, **Tremolo** and **Reverb** module when an electric guitar program is selected. The lower part of this view is dedicated to different guitar playing parameters.

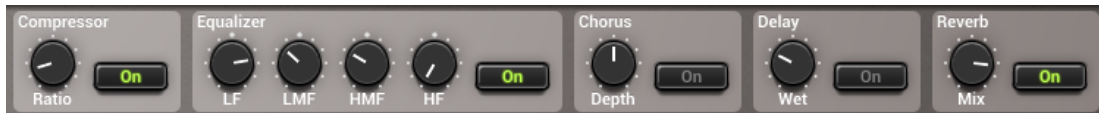


Figure 20: Middle part of the interface for the acoustic guitar.



Figure 21: Middle part of the interface for the electric guitar.

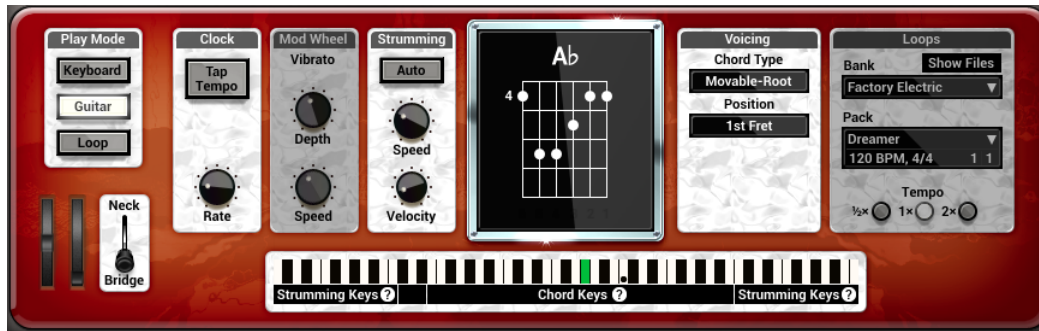


Figure 22: Lower part of the interface with guitar playing parameters.

5.2.1 Compressor

The **Compressor** module is used to automatically compress, in other words reduce, the dynamics of a signal. This module receives two input signals. The first one is the signal to be compressed while the second one is a control signal which triggers the compression process when it rises above a given level.



The **Compressor** starts to enter into action once the input signal reaches a threshold value. The amount of compression applied to the part of the signal exceeding the threshold value depends on the *Ratio* parameter which varies between value of 1:1 and 1:16. This parameter represents the ratio, in dB, between the portion of the output signal from the compressor above the threshold value and the portion of its input signal also exceeding the threshold value. As one might expect, increasing the ratio also increases the amount of compression applied to the signal. For example, a ratio of 1:5 means that if the input signal exceeds the threshold by 5 dB, the output signal will exceed the threshold by only 1 dB.

5.2.2 EQ

The **EQ** module provides equalization over the low, mid, and high frequency bands. It is composed of a low shelf filter, two peak filters, and a high shelf filter in series, labelled **LF**, **LMF**, **HMF**, and **HF** respectively.

The functioning of the low shelf filter is depicted in Figure 23. The filter applies a gain factor to low frequency components located below a cutoff frequency (110 Hz) while leaving those above unchanged. The *LF* knob is used to adjust the gain factor applied to the signal in a ± 15 dB range. In its center position there is no attenuation (0 dB). Turning it clockwise boosts the amplitude of low frequencies while turning it anti-clockwise reduces it.



The high frequency content of the signal is controlled with a high shelf filter that works in the opposite manner as the low shelf filter as illustrated in Figure 23. The filter applies a gain factor to

components located above a cutoff frequency (4800 Hz) while leaving those below unchanged. The gain factor applied to the signal, in a ± 15 dB range, is adjusted using *HF* knob. In its center position there is no attenuation (0 dB). Turning it clockwise boosts the amplitude of high frequencies while turning it anti-clockwise reduces it.

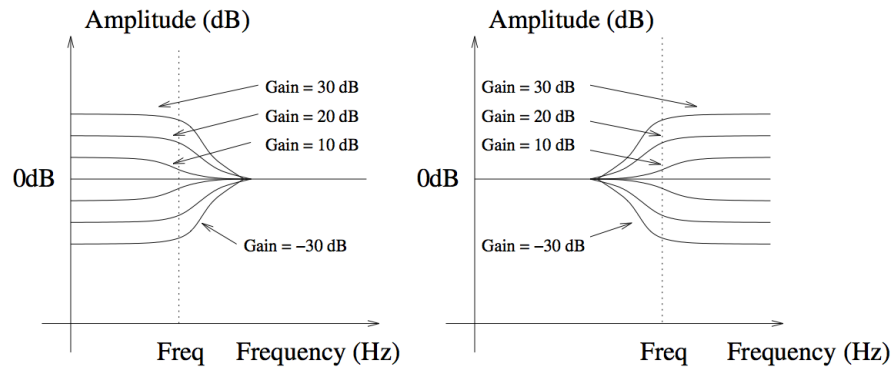


Figure 23: Low and high shelf filters.

The **EQ** module features two peak filters allowing to shape the signal in two frequency bands as illustrated in Figure 24. The filters apply a gain factor to frequency components in a band located around the cutoff frequency of the filters. This cutoff frequency of these two filters are located around 530 and 1600 Hz respectively. The gain factor applied at the cutoff frequency is controlled by the *LMF* and *HMF* knobs and can vary in a ± 15 dB range. When in its center position there is no attenuation (0 dB). Turning it clockwise boosts the amplitude of frequencies located around the cutoff frequency while turning it anti-clockwise reduces it.

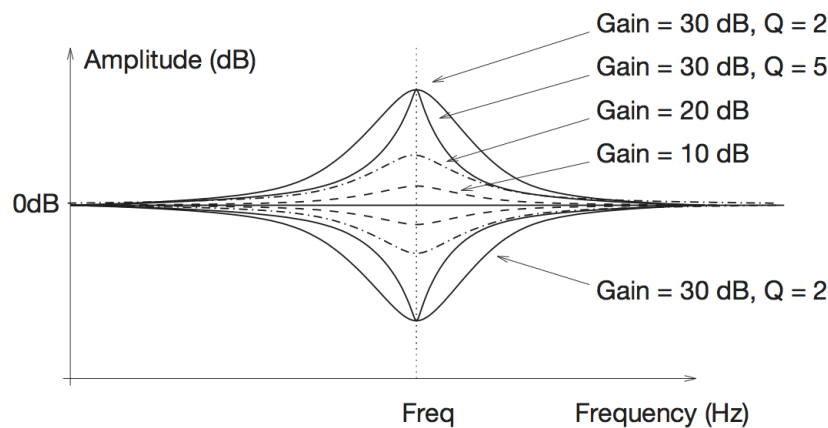


Figure 24: Peak filter.

5.2.3 Distortion

This module emulates the effect of a distortion pedal. The *Drive* control is a gain knob acting on the input signal. This parameter allows one to adjust the amount of distortion introduced in the signal by controlling how rapidly the signal reaches the non-linear portion of the distortion curve applied on the signal. In its leftmost position, the amplitude of the input signal is reduced by -6 dB; turning this knob clockwise allows one to increase its amplitude.



5.2.4 Amplifier

Strum Session is equipped with a versatile amplifier with spring reverb. With relatively few parameters, this amplifier module allows one to obtain a rich variety of sounds for different music styles. This module is available when an electric guitar preset is loaded.



The amplifier section of this module is switched *on* or *off* by clicking on the LED located in the top right corner of the module. The *Drive* knob is used to adjust the amount of distortion in the sound. The sound becomes more and more distorted as the knob is turned clockwise. The *Mid* knob is used to set the amount of mid-range frequencies in the sound. In its middle position, the sound is not modified, mids are cut or boosted by up to ± 12 dB by turning this knob to the left or right. The *Level* knob is a gain knob which is used to adjust the overall volume of the amplifier.

The *Low* and *High* parameters are used to boost or cut low and high frequencies respectively by up to ± 18 dB by turning the knob from its center position. These controls have a similar behavior for both channels. Additional control on the frequency response is obtained by using the *Bite* control which is switched *on* or *off* by clicking on the *Bite* LED just before to the channel selector. This parameter boosts high frequencies while cutting some low frequencies for a brighter sound.

The low-cut (or high-pass) filter is used to remove from the output sound of the instrument frequency components below the cut-off frequency. The cut-off frequency of the filter is increased by turning the knob clockwise. when this knob is in its leftmost position, the filter has no effect on the sound.

The spring reverb is turned *on* or *off* by using the *spring* LED in the right section of the amplifier. The *Mix* knob is used to set the amount of wet signal in the mix, turning the knob clockwise increasing the amount of reverberation in the signal.

5.2.5 Chorus

The chorus effect is used to make a source sound like many similar sources played in unison. It simulates the slight variations in timing and pitch of different performers executing the same part. The effect is obtained by mixing the original signal with delayed version obtained from the output of delay lines as shown in Figure 25. In the case of a chorus effect, the length of the delay lines must be short in order for the delayed signals to blend with the original signal rather than be perceived as a distinct echo. The length of the delay line can be modulated introducing a slight perceived pitch shift between the voices.

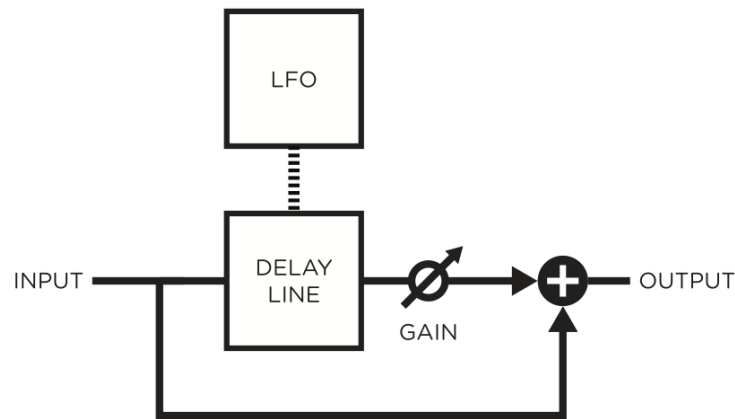


Figure 25: **Chorus** module.

The amount of modulation of the length of the delay lines is adjusted using the *Depth* knob. In the left position, there is no modulation and the length of the delay lines remains constant. As the knob is turned to the right, the length of the delay line starts to oscillate by an amount which increases as the knob is turned clockwise thereby increasing the amount by which the different voices are detuned.

5.2.6 Tremolo

The **Tremolo** module, introduces low frequency amplitude modulation, or tremolo in the sound. The *Depth* is used to set the amount of modulation in the amplitude of the signal in other words the amount of tremolo effect. In its leftmost position, the amplitude is not modulated and turning the knob clockwise gradually increases the amplitude of the modulation.



5.2.7 Delay

The **Delay** module consists in a stereo feedback loop with a short delay in the feedback. The output signal from the **Delay** module includes a mix of input signal (dry) and delayed signal (wet). The *Wet* knob is used to adjust the amplitude of the wet component in the final output. The amplitude is increased by turning the corresponding knob clockwise from no signal to an amplitude of +6dB.



5.2.8 Reverb

The *Reverb* effect is used to recreate the effect of reflections of sound on the walls of a room or hall. These reflections add space to the sound and make it warmer, deeper, as well as more realistic since we always listen to instruments in a room and thus with a room effect.

The ratio between the direct sound and the room response is adjusted with the *Mix* knob. This parameter is used to adjust the perceived distance between the source and the listener. In its leftmost position, only the direct sound is heard while when fully turned to the right, one only hears the room response.



5.2.9 The Pickups Module

The **Pickups** module simulates the action of the pickups on the guitar. There are two pickups in *Strum*, labeled *Neck* and *Bridge*. It is common, in different guitar types to listen to a single pickup at each of these position or to mix the signal from the bridge and neck pickups or to combine one of these with the signal from a third one positioned in the middle. On a real guitar, one can choose between these different configurations using switch on the body of the instrument.



5.2.10 The Play Mode

The **Play Mode** module allows one to switch between three playing modes: *Keyboard*, *Guitar* and *Loop* mode. In *Keyboard* mode, *Strum Session* plays notes as they are played on the keyboard without trying to voice chords. It is mainly used to play leads and to exactly control the voicing of chords. In *Guitar* mode, *Strum Session* plays in terms of a fretting and picking hand as a guitar player would. Chords played on the keyboard are automatically voiced by *Strum Session* and special strumming keys are used to strum chords and play individual strings. The *Loop* mode is similar to the *Guitar* mode except that the strumming key patterns are played from recorded MIDI files. For more information on the specific keyboard layout and playing techniques for each of these modes, please refer to sections 3.1, 3.2 and 3.3.



5.2.11 Pitch and Modulation Wheel

The MIDI pitch wheel allows one to vary the pitch of the note played. The pitch wheel can be moved with the mouse but it is also automatically connected to the pitch wheel signal received from your MIDI keyboard. The range of the pitch bend is 2 semi-tones up or down by default but can be changed. To adjust the range of the pitch bend, open the MIDI configuration window by clicking on the **MIDI** button located just below the MIDI led in the top part of the interface and use the **Pitch Bend Range** drop-down menu to select the range in semi-tones. Note that pitch bend also includes a *Slide* option. When this option is selected, pitch changes will occur in a discrete manner by semi-tone steps in order to reproduce the effect of a fretting hand finger going over the frets of the fretboard.



The modulation wheel is used to control vibrato. It can be activated on screen or from the modulation wheel of your MIDI controller (MIDI controller number 1). Other parameters can be linked to the modulation wheel using MIDI links as explained in Section 7.2.2. The amount of vibrato in the sound is adjusted using the *Depth* parameter of the **Mod Wheel** module while its speed is controlled using the *Speed* knob.



5.2.12 The Clock Module

This module is used to control the tempo of certain effects of the *Effects* section and loops. In order to synchronize effects, their *sync* button must be switched *on*.

When using *Strum Session* in standalone mode, the clock tempo, in bpm, is set by using the *Rate* knob. The tempo can also be adjusted by clicking at the desired tempo on the *Tap Tempo* pad of the module. Once the new tempo is detected, the value of the *Rate* knob is automatically adjusted. Note that double-clicking on the *Rate* sets the tempo to that of the currently loop pack.



In plugin mode, the *Tap Tempo* pad is replaced by a *Sync To Host* button. When this button is turned *on*, the rate of effects and loops is synchronized with the clock the host sequencer. When switched *off*, the tempo is determined by the value of the *Rate* knob.

5.2.13 Strumming

The *Strumming* module includes parameters which determine how the strumming is performed. The *Speed* parameter controls how rapidly the different strings are played when a strum is triggered. The speed of the strum is increased by turning the knob clockwise. This parameter can be modulated by the keyboard velocity using the *Velocity* knob. In its leftmost position, the speed is always that corresponding to the value of the *Speed* knob. Turning the velocity knob clockwise increases the strumming speed for high keyboard velocities while it reduces the speed for low keyboard velocities.



The *Auto* button is used to switch *on* or *off* the *Auto-Strum* option. It is only active when *Strum Session* is played in *Guitar* mode. When auto-strumming is *on*, chords are triggered with a downstroke as soon as one or more keys is depressed in the *Chord Keys* section. In other words, chords are played without having to use the special strumming keys. When the *Auto-Strum* option is turned *off*, chords are still recognized and voiced by *Strum Session* but the strings are not triggered until strumming keys are used. For more details on how to use the strumming keys, please refer to section 3.2.2.

5.2.14 Chord Display

The *Chord Display* is where *Strum Session* displays the name of the detected chord or note and the corresponding voicing selected. Strings are labeled from lowest to highest as follows:

- String 6: E (second E below middle C, MIDI note number 40)
- String 5: A (second A below middle C, MIDI note number 45)
- String 4: D (first D below middle C, MIDI note number 50)
- String 3: G (first G below middle C, MIDI note number 55)
- String 2: B (first B below middle C, MIDI note number 59)
- String 1: E (first E above middle C, MIDI note number 64)



Notes played are identified by a circle on the corresponding string and the position on the fretboard is determined by the fret number appearing in the upper left corner of the display. When a string is triggered, its number is highlighted at the bottom of the display while it vibrates. For a list of all the chords detected by *Strum Session*, please refer to section 9.

The name of the detected chord is displayed at the top of the displayed. When the chord contains no third or no fifth the corresponding *no 3* or *no 5* message is highlighted. While *Strum Session* can recognize a great variety of chords and find most voicings used by guitar players, it is possible that it will not find a voicing in its chord database for a particular chord played on the keyboard. In these cases, the *no match* message is lit. In this case, *Strum Session* will not display

a chord name but still propose a guitar voicing constructed according to a certain set of rules. The chord should still sound right but the message is displayed in order to indicate that the voicing chosen by *Strum Session* is probably very difficult to play on the guitar and might therefore not be commonly used by guitar players.

5.2.15 Chord Voicing

On the guitar, chords can be voiced in many different ways. The specific voicing chosen by *Strum Session* for a chord depends on the *Type* parameter. One can choose between open, movable, powerchords, and drop chords. For more information on the effect of these different chord types please refer to section 3.4. The voicing can be made more precise by using the *Position* parameter which instructs *Strum Session* on which neck position the chords should be played. The position is specified in fret number and indicates the lowest fret on which the lowest note of the chord should be played. It is not always possible to satisfy this constraint and *Strum Session* will respect this position whenever it is possible. This parameter is only valid for movable and drop chords as well as powerchords and it is therefore inactive when open chords are chosen.



5.2.16 Loops Module

The **Loops** module is used to play *Strum Session* with MIDI loops and it is only active when the *Loop* mode is selected. Loops contain strumming key patterns that are applied over the chords that are played in the *Chord Keys* section of the keyboard.

Strum Session presets are saved with packs of seven loops which are triggered using the white keys from the *Loop Keys* section of the keyboard as described in section 3.3. Loop packs are grouped into banks which can be browsed using the *Bank* drop-down menu. Individual packs in a bank are selected using the *Pack* drop-down menu. Selecting a pack automatically loads the seven loops included in a pack and maps them to the white keys of the *Loop Keys* section of the keyboard. A specific loop is selected and triggered by depressing one of these keys. The intended tempo of the loops in BPM, its time signature, and information on location within the loop appear just below the loop pack name.

Clicking on the *Show Files* button opens a Finder or Explorer window (on Mac OS X or Windows respectively) at the location where the MIDI loop files are stored. Banks correspond to folder names and loop packs under these categories are stored in these folders. Loops in a pack all share the same name except for the note (letters A to B) used to trigger a specific loop which is appended the loop name.

Once a loop is triggered, it is played at the tempo set by the *Clock* module or that of the host sequencer as explained in section 5.2.12. The tempo of the loop can be halved or doubled by using one of the *Tempo* buttons located below the pack name. Once triggered, loops play until the notes held in the *Chord Keys* sections are released or if the B \flat or second C of the *Loop Keys* section is depressed. The difference between these two keys is that the B \flat , *Mute* key mutes all strings and the sound stops as soon as the key is depressed. When the C or *Stop* key is used, the loop stops but the sound is allowed to decay naturally.



Loops can easily be dragged and dropped unto a sequencer track, the computer desktop or a specific folder by using the *MIDI Drag* control located below the **Loops** module. Click holding anywhere on this control selects the loop currently being played (the one corresponding to the purple key in the *Loop Keys* section) allowing to drag it where desired. Note that once a loop is copied onto a track, *Strum Session* should be used in *Guitar* mode for playback so that the keyswitch pattern that was copied is played correctly. For more information and tips on how to use and edit loops in a project, please refer to section 3.3.

5.2.17 Keyboard

The lower part of this view includes a keyboard covering five octaves. The specific layout of this keyboard depends on the playing mode selected allowing one to easily see how the keyboard is mapped in each mode as described in chapter 3. In *Guitar* and *Loop* mode, more information on the effect of individual notes can be obtained by clicking on the question mark appearing in the end sections of the keyboard. Clicking on the individual notes of this keyboard has the same effect as depressing the same note on an external MIDI keyboard. This can be useful to test *Strum Session* when no MIDI keyboard is connected to your computer.



6 Utility Section

The utility section is located at the top of the *Strum Session* interface and it includes important parameters and monitoring tools. For information on *Banks* and *Programs* please refer to Chapter 4.



6.1 The MIDI LED and MIDI configuration

The MIDI LED is located on the left of the level-meter. The LED blinks when the synthesizer receives MIDI signal. If the application is not receiving MIDI signal, make sure that the host sequencer is sending MIDI to *Strum Session*. If you are running in standalone mode, make sure that the MIDI controller you wish to use is well connected to your computer and that it is selected as explained in Section 7.

Clicking on the *MIDI* button just below the MIDI led opens the MIDI configuration pop-up window from which one can adjust parameters such as the pitch bend range, enable bank and program changes and manage the MIDI map used by *Strum Session* as explained in Chapter 7

6.2 Tuning

The *Tune* control, located to the right of the MIDI LED, is used to transpose the frequency of the keyboard. This control is composed of two numbers separated by a dot. The first number indicates a value in semi-tones while the second one indicates a value in cents (one hundredth of a semi-tone). The amount of transposition can be adjusted by click-dragging upward or downward on the semi-tone and cent controls. Double clicking on these controls brings back their value to zero. When the value of the *Tune* parameters is set to 0.00, the frequency of notes are calculated relative to A4 440Hz.

6.3 History and Compare

The *History* control allows one to go back through all the modifications that were made to programs since the application was started. In order to travel back and forth in time, use the left and right-pointing arrows respectively. The application will switch between different program states and indicate the time at which they were modified.

The *Compare* button, located above the *Program* display, is used to switch between **Edit** and **Compare** mode. This button is visible only once a modification is applied to a given program. It

allows one to revert to the original version of a program in order to compare it with the current version. When in **Compare** mode, edition is blocked and it is therefore not possible to modify any parameter. The **Compare** mode must then be switched *off* by clicking on the *Compare* button in order to resume edition.

6.4 Volume

The *Volume* knob is the master volume of the application. It is used to adjust the overall level of the output signal from the synthesizer. General level is increased by turning the knob clockwise.

6.5 Level Meter

The level meter allows one to monitor peak and RMS (root means square) level of the left (L) and right (R) output channels from the synthesizer. As a limiter is located at the output of *Strum Session*, it is important to make sure that the amplitude of the signal remains within values that ensure that no distortion is introduced in the signal at the output.

The 0 dB mark on the level meter has been adjusted to correspond to -20 dBFS (full scale). This means that at that level, the signal is -20 dB below the maximum allowed value. This 0 dB level mark should typically correspond to playing at mezzo forte (moderately loud) level. This ensures a headroom of 20 dB which should be more than enough to cover the dynamics of most playing situations and therefore guarantee that no additional distortion is added in the output signal.

A peak value mark allows one to follow the maximum level values reached by the output signal. The limiter is triggered when this mark enters the red zone of the level meter (17 dB) and it remains active while the side vertical bars at the top of the level meter are switched *on*.

6.6 The About Box

The **About** box is open by clicking on the chevrons located at the very top of the interface or on the product or company logo. The box is closed by clicking again on the chevrons or outside the box. Useful information is displayed in this box such as the program's version number, the serial number that was used for the authorization and the email address that was used for registration. The box also includes a link to the pdf version of this manual.

7 Audio and MIDI Settings

This chapter explains how to select and configure Audio and MIDI devices used by *Strum Session*. Audio and MIDI configuration tools are accessed by clicking on the *Audio Setup* button located in the lower left corner of the *Strum Session* interface and the *MIDI* button located just below the MIDI led in upper part of the interface.

Note that in plug-in mode the audio and MIDI inputs, sampling rate, and buffer size are set by the host sequencer.

7.1 Audio Configuration

7.1.1 Selecting an Audio Device

Audio configuration tools are available by clicking on the *Audio Setup* button located in the lower left corner of the *Strum Session* interface. The **Audio Setup** dialog first allows you to select an audio output device from those available on your computer. Multi-channel interfaces will have their outputs listed as stereo pairs.

On Windows, the audio output list is organized by driver type. The device type is first selected from the **Audio Device Type** drop-down list. If you have ASIO drivers available, these should be selected for optimum performance. The *Configure Audio Device* button allows you to open the manufacturer's setup program for your audio interface when available.

Once the audio input has been selected, you can then select a sampling rate and a buffer size from those offered by your audio interface.

7.1.2 Latency

The latency is the time delay between the moment you send a control signal to your computer (for example when you hit a key on your MIDI keyboard) and the moment when you hear the effect. Roughly, the latency is equal to the duration of the buffers used by the application and the sound card to play audio and MIDI. To calculate the total time required to play a buffer, just divide the number of samples per buffer by the sampling frequency. For example, 256 samples played at 48 kHz represent a time of 5.3 ms. Doubling the number of samples and keeping the sampling frequency constant doubles this time while changing the sampling frequency to 96 kHz and keeping the buffer size constant reduces the latency to 2.7 ms.

It is of course desirable to have as little latency as possible. *Strum Session* however requires a certain amount of time to be able to calculate sound samples in a continuous manner. This time depends on the power of your computer, the preset played, the sampling rate, and the number of voices of polyphony used. Note that it literally takes twice as much CPU power to process audio at a sampling rate of 96 kHz as it would to process the same data at 48 kHz, simply because you need to calculate twice as many samples in the same amount of time.

Depending on your machine you should choose, for a given sampling frequency, the smallest buffer size that allows you to keep real-time for a reasonable number of voices of polyphony.

7.2 MIDI Configuration

7.2.1 Selecting a MIDI Device

The list of available MIDI inputs appears at the bottom of the **Audio Setup** dialog. Click on the *Audio Setup* button located in the lower left corner of the *Strum Session* interface and then click on the checkbox corresponding to any of the inputs you wish to use.

7.2.2 Creating MIDI Links

Every control on the *Strum Session* interface can be manipulated by an external MIDI controller through MIDI control change assignments. In most cases this is much more convenient than using the mouse, especially if you want to move many controllers at once. For example, you can map the motion of a knob on the interface to a real knob on a knob box or to the modulation wheel from your keyboard. As you use the specified MIDI controllers, the controls move on the *Strum Session* interface just as if you had used the mouse.

In order to assign a MIDI link to a controller:

- On the *Strum Session* interface, right-click/Control-click on a control (knob, button) and select the **MIDI Learn** command.
- Move a knob or slider on your MIDI controller (this can be a keyboard, a knob box, or any device that sends MIDI). This links the control of *Strum Session* to the MIDI controller you just moved.

To deactivate a MIDI link, simply right-click/Control-click on the corresponding control on the *Strum Session* interface and select the **MIDI Forget** command.

7.2.3 Creating a default MIDI Map

It is possible to define a set of MIDI links, called a MIDI map, that will be loaded automatically when *Strum Session* is launched. Once you have defined a set of MIDI links that you wish to save, click on *MIDI* button to open the *MIDI* configuration window and click on the **Save Current as Default** button.

If you make changes to MIDI links after opening the program and wish to revert to the default MIDI map click on *MIDI* button to open the *MIDI* configuration window and click on the **Load Default** button.

If you wish to deactivate all the MIDI links at once open the *MIDI* configuration window and click on the **Clear MIDI Map** button.

7.2.4 MIDI Program Changes

Strum Session responds to MIDI program changes. When a program change is received, the current program is changed to the program having the same number as that of the program change message in the currently loaded bank.

If you do not wish *Strum Session* to respond to MIDI program changes, open the *MIDI* configuration window by clicking on the *MIDI* button and uncheck the **Enable Program Changes** option.

7.2.5 MIDI Bank Changes

In general, MIDI bank numbers are coded using two signals: the MSB (most significant byte) and LSB (least significant byte) transmitted using MIDI CC (continuous controller) number 0 and 32 respectively. The way these signals are used differs with different manufacturers.

In the case of *Strum Session*, the value of the MSB signal is expected to be zero while the value of the LSB signal represents the bank number. Banks are therefore numbered from 0 to 127 with this number corresponding to the position of a bank within the list of banks as displayed by the Bank manager (see Section 4.3). For example, an LSB value of 0 corresponds to the first bank in the bank list while an LSB value of 10 corresponds to the eleventh bank in the list. Note that a bank change only becomes effective after the reception of a new MIDI program change signal.

If you do not wish *Strum Session* to respond to MIDI bank changes, open the *MIDI* configuration window by clicking on the *MIDI* button and uncheck the **Enable Bank Changes** option.

7.2.6 Pitch bend

The MIDI pitch wheel allows one to vary the pitch of *Strum Session*. The pitch wheel can be moved with the mouse but it is also automatically connected to the pitch wheel signal received from your MIDI keyboard.

The range of the pitch bend is 2 semi-tones up or down by default but can be changed. To adjust the range of the pitch bend, open the MIDI configuration window by clicking on the **MIDI** button located just below the MIDI led in the top part of the interface and use the **Pitch Bend Range** drop-down list to select the range in semi-tones.

7.2.7 Aftertouch

Strum GS responds to aftertouch. By default, the aftertouch signal is linked to pitch bend. To adjust the range of the pitch bend, open the MIDI configuration window by clicking on the **MIDI** button located just below the MIDI led in the top part of the interface and use the **Aftertouch Bend** drop-down list to select the range in semi-tones. Aftertouch can also be used to control any other parameter through the use of MIDI links as described in section 7.2.2. A triggering threshold can

also be applied to the aftertouch. This is adjusted using the **Aftertouch Threshold** drop-down list. The values in the list correspond to values of the aftertouch signal above which aftertouch will be triggered in *Strum GS-2*. When notes are played with aftertouch values below this threshold value, the aftertouch is not triggered. In order to always have aftertouch, set the threshold to a value of zero. Not that the behavior of this parameter may depend on the settings or sensitivity of your MIDI keyboard.

7.2.8 Live MIDI Loop Control

In order to make MIDI loops more expressive, *Strum Session* allows one to link two different MIDI controllers to some performance parameters such as playing softer or louder, slower or faster, increase or decrease the strumming span and putting emphasis on lower or higher strings when strumming. To access this modulation matrix, open the MIDI configuration window by clicking on the **MIDI** button located just below the MIDI led in the top part of the interface and assign effects to controllers in the Live MIDI Loop Control section of the window.

7.2.9 Modulation wheel

Strum Session responds to MIDI modulation (MIDI controller number 1) and controls vibrato by default. For more details, please refer to Section 5.2.11. The modulation wheel can also be used to control any other parameter through the use of MIDI links as described in section 7.2.2.

8 Using *Strum Session* as a Plug-In

Strum Session is available in VST, RTAS, AAX and Audio Units formats and integrates seamlessly into the industry most popular multi-track recording and sequencing environments as a virtual instrument plug-in. *Strum Session* works as any other plug-in in these environments so we recommend that you refer to your sequencer documentation in case you have problems running it as a plug-in. We review here some general points to keep in mind when using a plug-in version of *Strum Session*.

8.1 Audio and MIDI Configuration

When *Strum Session* is used as a plug-in, the audio and MIDI ports, sampling rate, buffer size, and audio format are determined by the host sequencer.

8.2 Automation

Strum Session supports automation functions of host sequencers. All parameters visible on the interface can be automatized except for the **Polyphony**, **Bank**, **Program** and **History** commands.

8.3 Multiple Instances

Multiple instances of *Strum Session* can be launched simultaneously in a host sequencer.

8.4 Saving Projects

When saving a project in a host sequencer, the currently loaded program is saved with the project in order to make sure that the instrument will be in the same state as when you saved the project when you re-open it. Note that banks of programs are not saved with the project which implies that if you are using MIDI program changes in your project, you must make sure that the bank you are using in your project still exists on your disk when you reload the project. The programs must also exist and be in the same order as when the project was saved.

8.5 Performance

Using a plug-in in a host sequencer requires CPU processing for both applications. The load on the CPU is even higher when multiple instances of a plug-in or numerous different plug-ins are used. To decrease CPU usage, remember that you can use the **freeze** or **bounce to track** functions of the host sequencer in order to render to audio the part played by a plug-in instead of recalculating it every time it is played.

9 Appendix - Lists of Chords Detected by *Strum Session*

We present here a list of the main chords recognized by *Strum Session* and for which it can find a guitar voicing. The following guidelines should be kept in mind:

- The chords are presented using C as the root. They can be transposed in any key.
- For all chords, inversions are recognized except if they conflict with another chord from this list.
- Inverted chords on the keyboard do not necessarily have a corresponding voicing on the guitar. In these cases *Strum Session* will still propose a voicing; it is not guaranteed that this voicing will be playable on the guitar and the *no match* sign will be lit in the chord display.
- When chords are played on the keyboard, the order of the notes above the root is not taken into account by *Strum Session*. This implies that you can play the chords as you know them on the keyboard without having to know or learn special voicings used by guitar players.
- Certain guitar voicings do not include all the notes played on the keyboard.
- In addition to the chords listed below, *Strum Session* can recognize other chords for which it has no guitar voicing. In these cases *Strum Session* will still propose a voicing; it is not guaranteed that this voicing will be playable on the guitar and the *no match* sign will be lit in the chord display.

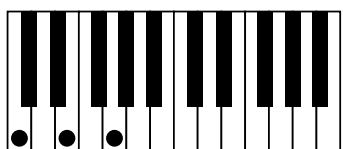
List of Chords recognized and voiced by *Strum Session*

C	Cm	Csus2
Csus4	Caug	Cdim
C(b5)	C6	Cm6
Cm6 (no 5)	C7	C7 (no 3)
C7 (no 5)	Cm7	Cm7 (no 5)
C7sus2	C7sus2 (no 5)	C7sus4
C7sus4 (no 5)	C7#5	Cdim7
C7b5	Cm7b5	CMaj7
CMaj7 (no 3)	CMaj7 (no 5)	CmMaj7

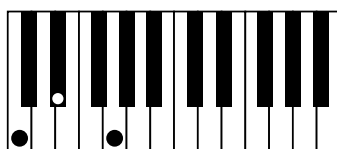
List of Chords recognized and voiced by *Strum Session*

CMaj7sus2	CMaj7sus4	CMaj7b5
Cadd9	C9	C9 (no 5)
Cm9	Cm9 (no 5)	C9sus4
C9sus4 (no 5)	C9#5	C9b5
CMaj9	CMaj9 (no 5)	CmMaj9
CmMaj9 (no 5)	CMaj9b5	C6 9
Cm6 9	Cm6 9 (no 5)	Cm11
Cm11 (no 5)	C13	C13 (no 5)
Cm13	Cm13 (no 5)	C13b9
C13b9 (no 5)	CMaj13	CMaj13 (no 5)
C7b9	C7b9 (no 5)	C7#9
C7#9 (no 5)	C7b5b9	C7b5#9
C7#5b9	C7#5#9	C7#11
Cm7 11	Cm7 11 (no 5)	CMaj7#11
C7b9#11	C7#9#11	C7#9#11 (no 3)
C7 13	C7 13 (no 5)	Cm7 13
Cm7 13 (no 5)	C7b13	CMaj7 13
CMaj7 13 (no 5)	C7b9b13	C7#9b13
C9#11	CMaj9#11	C9b13

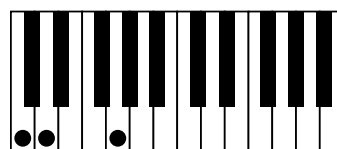
List of Chords - Example of a possible position on the keyboard.



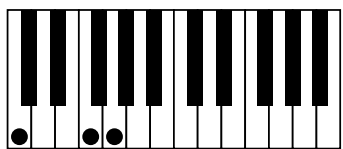
C



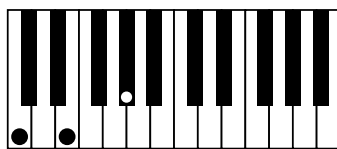
Cm



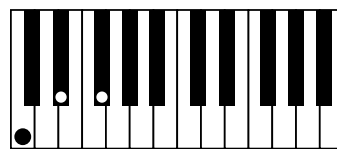
Csus2



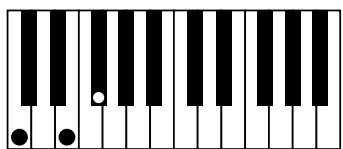
Csus4



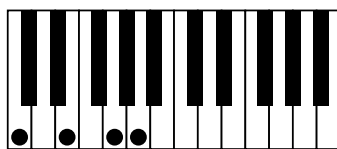
Caug



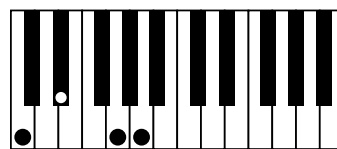
Cdim



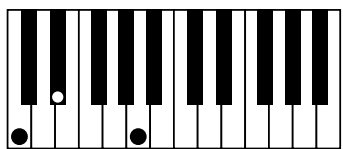
C(b5)



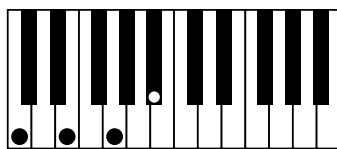
C6



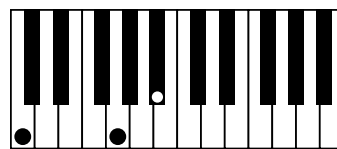
Cm6



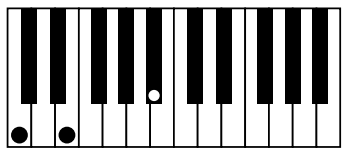
Cm6 (no 5)



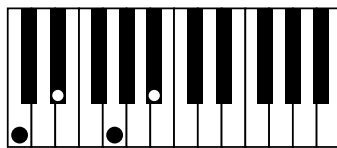
C7



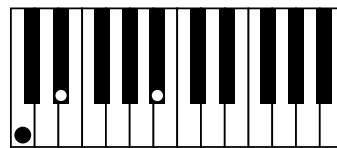
C7 (no 3)



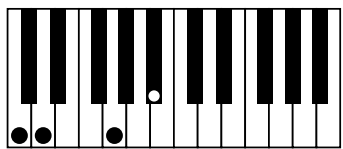
C7 (no 5)



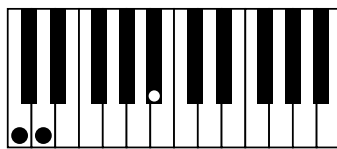
Cm7



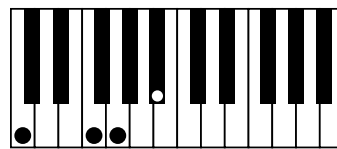
Cm7 (no 5)



C7sus2

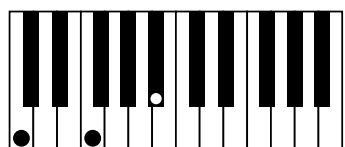


C7sus2 (no 5)

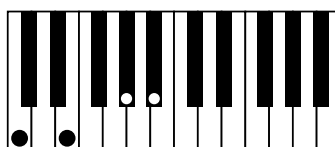


C7sus4

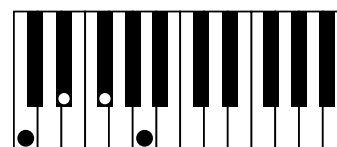
List of Chords - Example of a possible position on the keyboard.



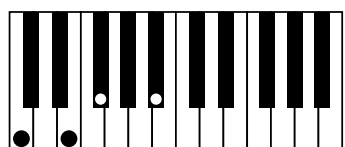
C7sus4 (no 5)



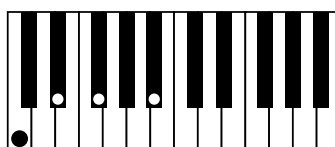
C7#5



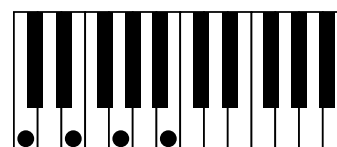
Cdim7



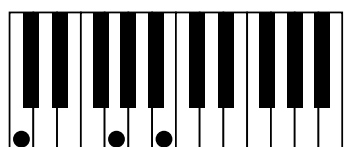
C7b5



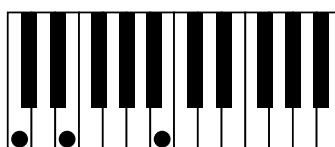
Cm7b5



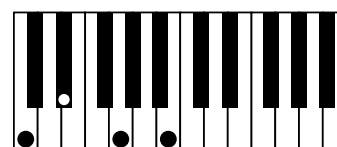
CMaj7



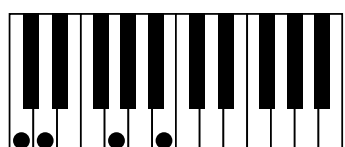
CMaj7 (no 3)



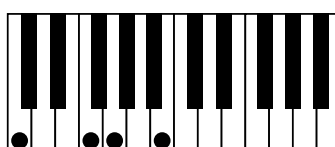
CMaj7 (no 5)



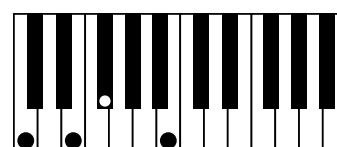
CmMaj7



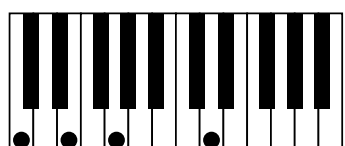
CMaj7sus2



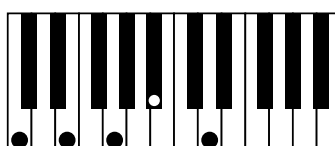
CMaj7sus4



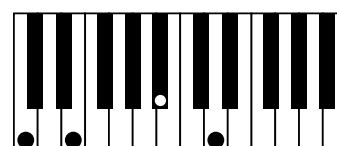
CMaj7b5



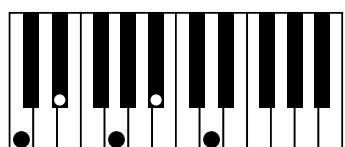
Cadd9



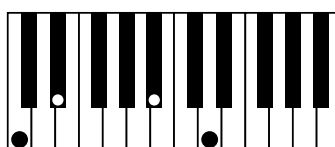
C9



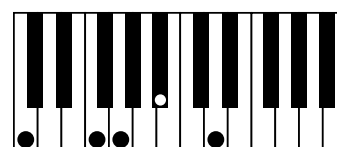
C9 (no 5)



Cm9

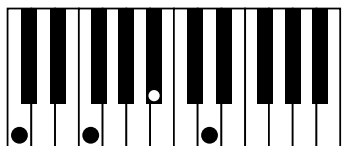


Cm9 (no 5)

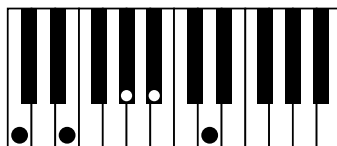


C9sus4

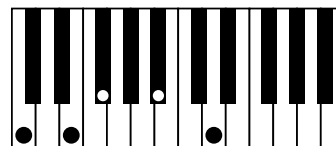
List of Chords - Example of a possible position on the keyboard.



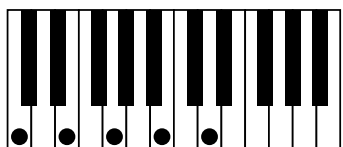
C9sus4 (no 5)



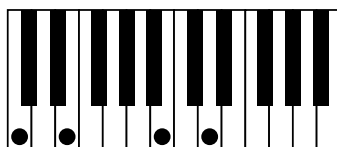
C9#5



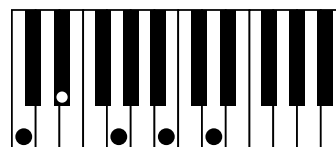
C9b5



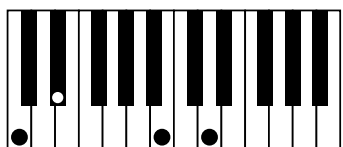
CMaj9



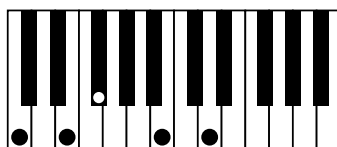
CMaj9 (no 5)



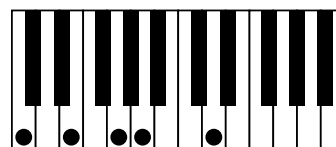
CmMaj9



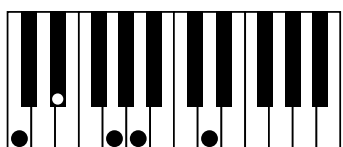
CmMaj9 (no 5)



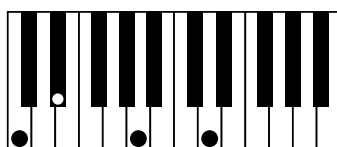
CMaj9b5



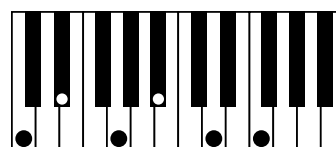
C6 9



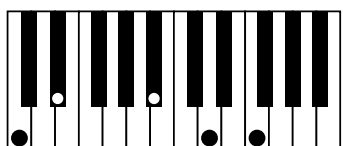
Cm6 9



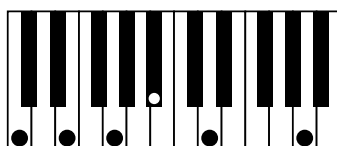
Cm6 9 (no 5)



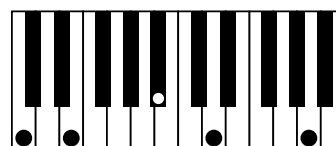
Cm11



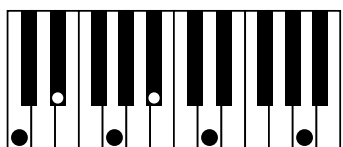
Cm11 (no 5)



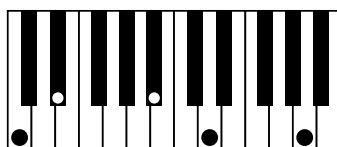
C13



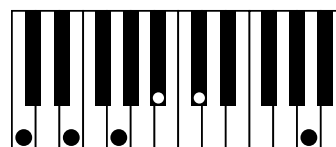
C13 (no 5)



Cm13

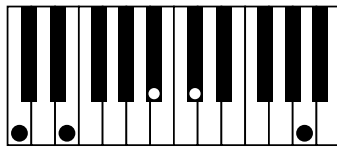


Cm13 (no 5)

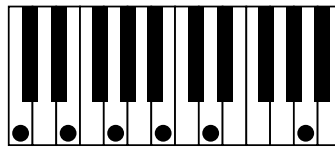


C13b9

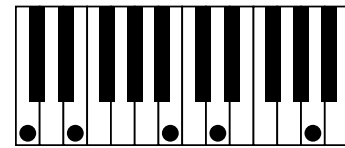
List of Chords - Example of a possible position on the keyboard.



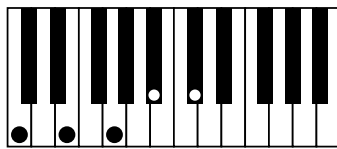
C13b9 (no 5)



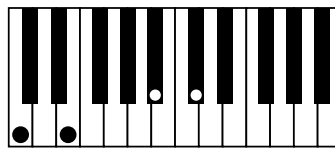
CMaj13



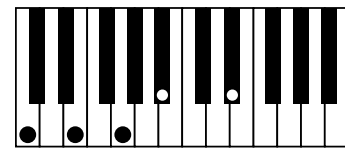
CMaj13 (no 5)



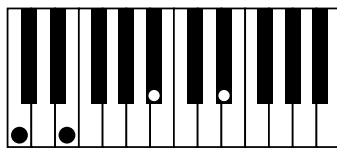
C7b9



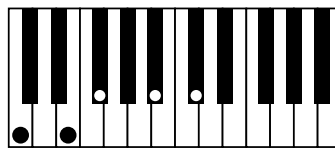
C7b9 (no 5)



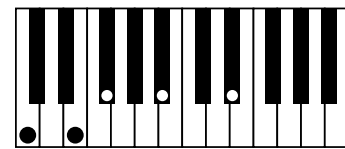
C7#9



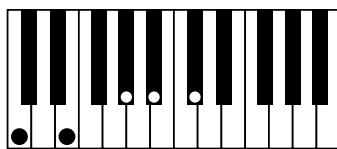
C7#9 (no 5)



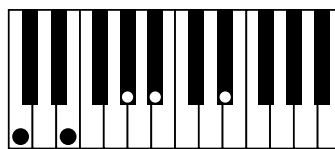
C7b5b9



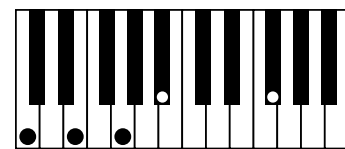
C7b5#9



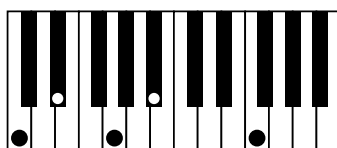
C7#5b9



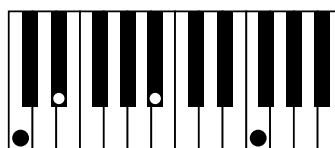
C7#5#9



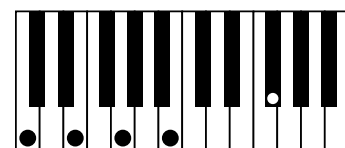
C7#11



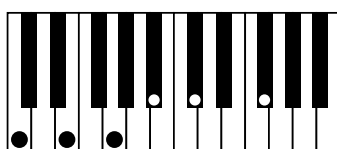
Cm7 11



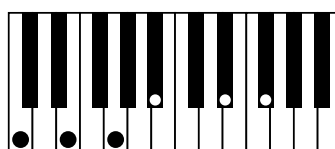
Cm7 11 (no 5)



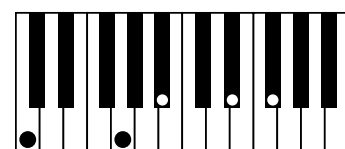
CMaj7#11



C7b9#11

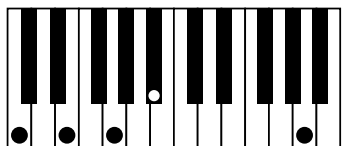


C7#9#11

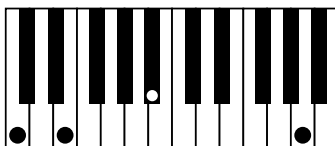


C7#9#11 (no 3)

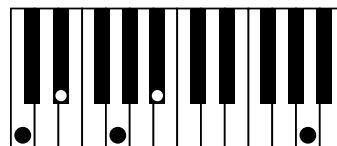
List of Chords - Example of a possible position on the keyboard.



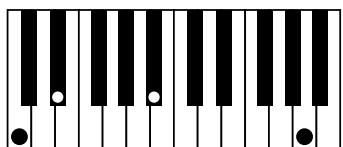
C7 13



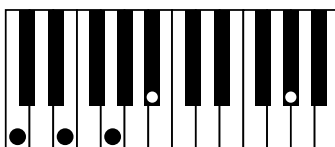
C7 13 (no 5)



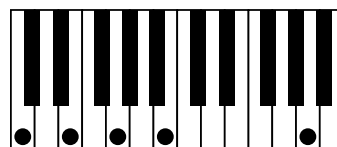
Cm7 13



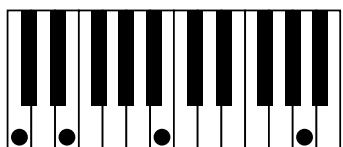
Cm7 13 (no 5)



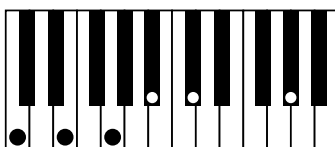
C7b13



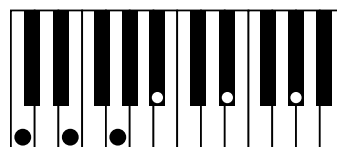
CMaj7 13



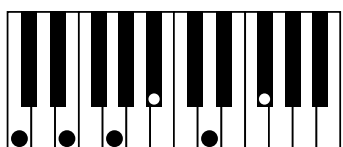
CMaj7 13 (no 5)



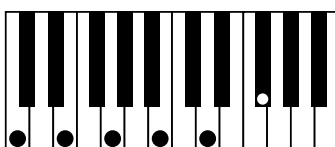
C7b9b13



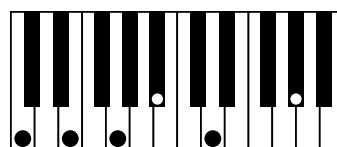
C7#9b13



C9#11



CMaj9#11



C9b13

10 License Agreement

IMPORTANT! CAREFULLY READ ALL THE TERMS AND CONDITIONS OF THIS AGREEMENT BEFORE OPENING THIS PACKAGE. OPENING THIS PACKAGE INDICATES YOUR ACCEPTANCE OF THESE TERMS AND CONDITIONS. IF YOU DO NOT AGREE WITH THE TERMS AND CONDITIONS OF THIS AGREEMENT, PROMPTLY RETURN THE UNOPENED PACKAGE AND ALL COMPONENTS THERETO TO THE PARTY FROM WHOM IT WAS ACQUIRED, FOR A FULL REFUND OF ANY CONSIDERATION PAID.

This software program, any printed materials, any on-line or electronic documentation, and any and all copies of such software program and materials (the “**Software**”) are the copyrighted work of Applied Acoustics Systems DVM Inc. (“**AAS**”), its subsidiaries, licensors and/or its suppliers.

1. **LICENSE TO USE.** The Licensee is granted a personal, non-exclusive and non-transferable license to install and to use one copy of the Software on a single computer solely for the personal use of the Licensee. Use of the Software is subject to this Agreement.
2. **RESTRICTIONS ON USE.** The Licensee may not nor permit third parties to (i) make copies of any portion of the Software, other than as expressly permitted under this Agreement; (ii) modify, translate, disassemble, decompile, reverse engineer or create derivative and/or competitive products based on any portion of the Software; (iii) provide use of the Software in a network, timesharing, interactive cable television, multiple CPU service bureau or multiple user arrangement to users not individually licensed by AAS, other than as expressly permitted by the terms of this license. The Software is licensed to you as a single product. Its component parts may not be separated for use on more than one computer.
3. **OWNERSHIP.** AAS retains title to the Software, including but not limited to any titles, computer code, themes, objects dialog concepts, artwork, animations, sounds, audio effects, methods of operation, moral rights, any related documentation and “applets” incorporated into the Software. AAS retains ownership of and title to all intellectual property rights in the Software, underlying technology, related written materials, logos, names and other support materials furnished either with the Software or as a result of this Agreement, including but not limited to trade secrets, patents, trademarks and copyrights therein. Licensee shall not remove or alter any copyright or other proprietary rights notices contained on or within the Software and shall reproduce such notices on all copies thereof permitted under this Agreement or associated documentation.
4. **LIMITED WARRANTY.** Except for the foregoing, THE SOFTWARE IS provided “AS IS” without warranty or condition of any kind. AAS disclaims all warranties or conditions, written or oral, statutory, express or implied, including but not limited to the implied warranties of merchantable quality or fitness for a particular purpose, title and non-infringement of rights of any other person. AAS does not warrant that THE SOFTWARE will meet the Licensee’s requirements or that the operation of the software will be uninterrupted or ERROR-FREE.

5. **LIMITATION OF LIABILITY.** TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, IN NO EVENT WILL AAS BE LIABLE TO THE LICENSEE OR ANY THIRD PARTY FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, INCIDENTAL OR EXEMPLARY DAMAGES WHATSOEVER, INCLUDING BUT NOT LIMITED TO LOSS OF REVENUE OR PROFIT, LOST OR DAMAGED DATA, BUSINESS INTERRUPTION OR ANY OTHER PECUNIARY LOSS WHETHER BASED IN CONTRACT, TORT OR OTHER CAUSE OF ACTION, EVEN IF AAS HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES, EXCEPT IN RELATION TO GROSS NEGLIGENCE OR WILFUL BREACH OF THIS AGREEMENT BY AAS. NO AAS AGENT, REPRESENTATIVE OR DEALER IS AUTHORIZED TO EXTEND, MODIFY OR ADD TO THIS WARRANTY ON BEHALF OF AAS. THE TOTAL LIABILITY OF AAS FOR DAMAGES, WHETHER IN CONTRACT OR TORT, UNDER OR RELATED IN ANY WAY TO THIS AGREEMENT SHALL BE LIMITED TO THE LICENSE FEES ACTUALLY PAID BY LICENSEE TO AAS, OR IF NO FEES WERE PAID, AAS' LIST PRICE FOR THE SOFTWARE COVERED BY THIS LICENSE. THE EXCLUSION OF IMPLIED WARRANTIES AND/OR THE LIMITATION OF LIABILITY IS NOT PERMITTED IN SOME JURISDICTIONS, AND SOME OR ALL OF THESE EXCLUSIONS MAY THEREFORE NOT APPLY.
6. **TERMINATION.** This License also shall extend to the Software and any updates or new releases thereof obtained by the Licensee, if any, subject to any changes to this License made by AAS from time to time and provided to the Licensee, provided AAS is under a separate obligation to provide to Licensee such updates or upgrades and Licensee continues to have a valid license which is in effect at the time of receipt of each such update or new release. This License shall remain in effect until terminated. The Licensee may terminate this Agreement at any time, upon notification to AAS. This Agreement will terminate immediately without notice from AAS if the Licensee fails to comply with any provision of this License. Any such termination by AAS shall be in addition to and without prejudice to such rights and remedies as may be available, including injunction and other equitable remedies. Upon receipt of notice of termination from AAS, the Licensee must (a) immediately cease to use the Software; (b) destroy all copies of the Software, as well as copies of all documentation, specifications and magnetic media relating thereto in Licensee's possession or control; and (c) return all original versions of the Software and associated documentation. The provisions of Sections 1, 3, and 5 shall survive the termination of this Agreement.
7. **GOVERNING LAW.** This Agreement shall be governed by and construed in accordance with the laws of the Province of Quebec, without regard to the United Nations Convention On Contracts for the International Sale of Goods and conflict of laws provisions, if applicable, and the parties hereby irrevocably attorn to the jurisdiction of the courts of that province. Les parties sont d'accord à ce que cette convention soit rédigée en langue anglaise. The parties have agreed that this agreement be drafted in the English language.
8. **SEVERABILITY.** If any of the above provisions are held to be illegal, invalid or unenforceable, such provision shall be severed from this Agreement and this Agreement shall not be

rendered inoperative but the remaining provisions shall continue in full force and effect.

9. ENTIRE AGREEMENT. This Agreement is the entire agreement between AAS and the Licensee relating to the Software and: (i) supersedes all prior or contemporaneous oral or written communications, proposals and representations with respect to its subject matter; and (ii) prevails over any conflicting or additional terms of any quote, order, acknowledgement, or similar communication between the parties during the term of this Agreement except as otherwise expressly agreed by the parties. No modification to the Agreement will be binding, unless in writing and signed by a duly authorized representative of each party.
10. NON-WAIVER. No delay or failure to take any action or exercise any rights under this Agreement shall constitute a waiver or consent unless expressly waived or consented to in writing by a duly authorized representative of AAS. A waiver of any event does not apply to any other event, even if in relation to the same subject-matter.

Index

- about, 46
- aftertouch, 29, 49
- alternate bass, 20
- amplifier, 38
- architecture, 11
- arp keys, 20
- arpeggios, 20, 26
- audio, 47
 - configuration, 9, 47
 - device, 47
 - latency, 47
- auto-strum, 12, 20, 42
- automatic chords, 17
- backup, 32
- bank, 30
 - backup, 32
 - create, 30
 - delete, 30
 - factory library, 33
 - rename, 30
 - sharing, 32
- bank manager, 30
- bass and chords, 28
- bend, 28
- buffer size, 9, 47
- chords
 - change, 21
 - change of pitch, 22
 - detection, 11
 - detection time, 43
 - display, 42
 - drop, 24
 - list, 52
 - lowest, 24
 - movable, 24
 - open, 24
 - root, 24
 - type, 43
 - voicing, 24
- chorus, 39
- clock, 35, 41
- coarse, 34
- compare, 45
- compressor, 36
- contact, 10
- default MIDI map, 48
- delay, 40
- distortion, 37, 46
- downstroke, 18
- drop chords, 24
- drop-down menus, 34
- effect
 - master clock, 41
 - synchronization, 35
- EQ, 36
- factory library, 9
- fine tuning, 34
- guitar mode, 11, 16, 40
- hammer-on, 16
- help, 9, 10
- history, 45
- individual strings, 20
- installation, 8
- inteface
 - resize, 15
- interface, 13, 34
 - drop-down menus, 34
 - knobs, 34
 - numerical displays, 34
 - switches, 34
- keyboard, 14, 44
- keyboard mode, 11, 16, 40
- knobs, 34

- latency, 47
- level meter, 46
- library
 - restore, 33
- limiter, 46
- loop, 11
 - control, 23, 50
 - creating, 23, 24
 - expressivity, 23, 50
 - modulation, 23, 50
 - using, 22
- loop mode, 22, 40
- loops module, 43
- manual, 46
- master clock, 41
- master volume, 46
- MIDI, 47
 - aftertouch, 49
 - bank change, 49
 - configuration, 9, 48
 - device, 48
 - link map, 48
 - links, 48
 - loop control, 23, 50
 - loop player, 43
 - pitch bend, 41, 49
 - program change, 30, 32, 49
 - receiving, 45
- MIDI controller, 48
- MIDI led, 45
- minimum system requirements, 7
- modulation, 34
- modulation wheel, 41
- modules, 34
- movable chords, 24
- muffled strokes, 19
- multi-effect
 - synchronization, 35
- mute all, 19
- muted strum, 27
- no 3, 42
- no 5, 42
- no match, 42
- numerical displays, 34
- open chords, 24
- overview, 11
- palm muting, 19
- parameters, 34
- partial strumming, 27
- peak, 46
- picking, 20
- pickups, 40
 - bridge, 40
 - neck, 40
 - position, 40
- pitch bend, 49
- pitch wheel, 29, 41
- play mode, 40
- play view, 35
- playing mode, 11
- plug-in, 10
 - audio configuration, 51
 - automation, 51
 - formats, 51
 - MIDI configuration, 51
 - multiple instances, 51
 - performance, 51
 - saving projects, 51
- polyphony, 8
- powerchords, 25
- programs, 9, 30
 - backup, 32
 - bank manager, 30
 - copy, 30
 - factory library, 33
 - move, 30
 - saving, 30
 - sharing, 32
- pull-off, 16
- range, 42
- resize, 15

- restore factory library, 33
- reverb, 40

- sampling rate, 8, 9
- save, 30
- save as, 30
- signal flow, 11, 12
- standalone, 9
- start-up, 9
- stationary bend, 28
- strum down, 21
- strum range, 42
- strumming, 12, 42
 - auto, 42
 - keys, 42
- strumming keys, 12, 16, 17, 20
- sustain pedal, 21
- switches, 34
- synchronization, 35, 41
- synchronization to a host, 41
- system requirements, 7

- transpose, 45
- tremolo, 39
- trills, 27
- tuning, 45

- upstroke, 18
- utility section, 45

- vibrato, 41
- voicing, 11
- volume, 46
- VU meter, 46