The **Clipper** is a guitar-format digital instrument with 6 "Strings" (columns) x 24 "frets" (rows) of touch-sensitive keys for a total of 144 keys, (also available in 12- or 24-string formats). Each of the 144 keys is fitted with a 3-color LED which can output over 32,000 colors. The Clipper lighting system can be used for lighting effects as well as provide feedback from music programs such as Ableton Live. The key locations are mapped internally to MIDI Note-numbers on reserved MIDI channels. The colors are mapped to a 127-location color-table so they can be individually selected by the 7-bit MIDI note-velocity value attached to each note.

You can use the Clipper's special command set to create a variety of lighting effects, teaching applications, and interactive performances. These standard MIDI commands need no special coding from the device side other than a simple user-setup in the Events-Menu to make the assignments. Refer to Appendix 1 for a detailed listing. The default presets, or "Songs", in the Clipper top-menu include various lighting setups. Investigate the SENSORS/EVENTS menu details to learn how these setups work.

This document primarily describes the detailed User setups to the Clipper for use with Ableton Live. These setups are pre-loaded in your instrument, and this is your reference to the menu screens pertinent to the interactive lighting features.

For more detailed descriptions of the operation of the Ztar User Interface please refer to the ZPC2 Operation manual. There you will find complete discussions of Zoning and Tuning the fingerboard as well as the various MIDI assignments. This document concerns primarily setup and operations for the Lighting controls only

- "Ableton Live" property of Ableton Inc.

Contents:

Hardware setup

- The Clipper can be connected to:
    - Your computer wirelessly with a AirPower2 receiver or with a standard USB 2.0 cable
    - Your MIDI system wirelessly with the AirPower2 receiver or directly with a standard MIDI cable .
- At full brightness the LEDs require more power than a standard USB2.0 port can provide. In order to run the lighting The Clipper must be powered by Internal Battery or "Phantom power" through the MIDI cable. Phantom power is supplied to the MIDI cable by our PB-1 interface adapter and the AC/9VDC wall-adapter. You can still connect to your computer via the USB cable which will only support the MIDI control functions of the instrument.

Overview of the Ableton Live interface

Ableton Live (currently at Rev.9.6) is a popular music software application that allows a user to record and playback clips of music in a wide variety of ways, all synced to a master beat.
When configured for Ableton Live the Clipper functions much like other commercially available Clip grid controllers, but with added functionality.

On the Clipper the color of each LED represents the color of each respective clip in Ableton's Session View.   When a Clip is loaded in Ableton the fingerboard key-color assumes the clip's color.

- When a clip is loaded its LED is at half brightness.
- When a clip is cued to play its LED blinks between half brightness and off.
- When the clip launches its LED lights up at full brightness.
- When a Clip location is empty, the fingerboard key is either OFF or shows an assigned background color.

This allows the user to color code a Live set and not have to look at the computer screen throughout a performance. Thus the Clipper liberates the computer musician from the laptop stand.

If you're interested in programming the Clipper for your own lighting effects it's easy with our dedicated MIDI control command-set. Using standard MIDI Note and Continuous Control messages you can light the LEDs to any color, program and stream text and images, scroll the display, blink individual LEDs and perform other functions. Using the setups in the Clipper OS you can create Zones of color, Tuning maps for different guitar tunings, load and select images, and create animations. Refer to Appendix 1 for a listing of the command set.

Now that you're excited about your new Clipper fingerboard, let's get into the specifics.

Ableton Live Session and Clipper Live Zone Overview

The Ableton Live Session view displays an array of colored boxes that form the Clip Grid and change dynamically as you play. Notice that the Clipper fingerboard changes to match the computer screen.

- ***In Ableton Live the horizontal row is a Scene*** *and the* ***Column is a Track****. The analogy for the column is a "Bass-track" or "Piano-track", for example. The Scene then comprises a clip for each Track, or Instrument in this case.*
- *Only one clip can play per track, or column, at a given time*

3

The *RedBox* is the outlined active region of the Ableton Clip-Grid that displays on your Clipper fingerboard.

The *LiveZone* is the active area on the Clipper fingerboard that represents the Redbox in Ableton Live.

Ableton Live controls the Clipper lighting by use of the Clipper Python script available here.

http://www.starrlabs.com/media/software.php

The Clipper allows a total of over 32000 colors. When you are connected to Ableton, Live's color table is loaded into the Clipper so that you can match the LEDs to the clip-colors in your session.

Clipper LiveZone:

This is a fingerboard-zone of arbitrary size, created in the Clipper that reflects the Redbox on the Ableton Live screen. In performance mode, clips are loaded, recorded or launched.

When we set up the Ableton "Live Zone" on the controller we define X-Y boundaries on the fingerboard. The overall size is arbitrary, the maximum size being 5x23 for Clips with the sixth row for StopClips, and the twenty-fourth column for Scene Launch keys. Naturally the Live Zone can be much smaller to control a smaller Clip Grid. Ableton only controls the lighting for the "Live" Zone that is set up on the Clipper. All other fingerboard Zones are unaffected.

Launch Scene Keys:
This is the column of Green keys placed to the far right of the RedBox for a Zone of any size. The function of each of these keys is to trigger a Scene, or all clips in the corresponding row of the Clip Grid.

Stop Keys:
This is a row of keys placed below the clip keys and the launch scene keys. The function of each of a Stop key is to stop all clips in the corresponding column in the clip key array. The bottom key in the far right column is Stop All Clips.

Blinking stop clip button:
 The stop clip button should blink until the downbeat where it stops the playing clip (bringing it to half brightness) and returns the stop clip button to full brightness (non-blinking).

Moving the "RedBox"

You can access more clips in the Session view by adjusting the position of the Redbox.
This is done from the Clipper by sending these MIDI note commands:

Up	-----	Note 110	Channel 16
Down	-----	Note 109	Channel 16
Right	-----	Note 108	Channel 16
Left	-----	Note 107	Channel 16
Rotate 90°	-----	Note 84	Channel 16

[Note: the RedBox boundaries may also be changed within the Clipper's onboard user-interface by changing the Zone boundaries for the 'RedBox' zone.]

Setting up Ableton Live to work with the Clipper

The Clipper communicates with Ableton's Live API using MIDI Remote Scripts. These Python scripts interpret incoming standard MIDI messages and translate them to API calls within Live. The scripts are available on our website.

Script Installation:

Download the file from our website, ******* www.starrlabs.com\mystery-folder *****
Un-zip and copy entire "Clipper" folder into "MIDI Remote-Scripts" folder of Ableton Live:

Windows: /Program Files/Ableton/Live 9.x.x/Resources/MIDI Remote Scripts

Mac OSX: /Applications/Live 9.x.x/App-Resources/MIDI Remote Scripts
Ctl-click on Live in the Applications folder. Select " Open Package Contents" to view these folders.

 Once this folder has been copied, (re)start Ableton Live, open up the Preferences menu and navigate to the MIDI tab. There you will be presented with a grid of dropdown menus. In the control surface column open one of the dropdown menus and scroll down to "Clipper". In the next two columns to the right, select your input and output ports for the Clipper or AirPower2(version-X) if you are wireless.
In the lower half of the MIDI preferences window you will see a grid with options for all of the active MIDI ports. Make sure that both "Track" and "Remote" are set to "On" for your previously assigned input and output ports. The "Sync" option is for sending/receiving MIDI clock messages and can be left off for most use cases. We recommend leaving it off to reduce the amount of messages the Clipper has to handle.

Now Ableton is ready to interact with your Clipper, but we still need a "Song" preset for the Clipper. We have Ableton-ready presets available on our website which can be programmed into your Clipper via System Exclusive MIDI messages (SysEx).
http://www.starrlabs.com/clipper/presets (!!!)

You can also set up the necessary fingerboard zones using the following instructions.

Setting up the Clipper to work with Ableton Live

- *A variety of Clipper presets have already been included in your instrument for use with Ableton Live. The following instructions explain how the setups are created and the way the Clipper interacts with Ableton Live.*

- The LED hardware can be controlled externally with standard MIDI note and CC messages on MIDI channels 15 and 16. The Ableton live script communicates via USB or wirelessly with your Clipper to light up the Session view on your fingerboard. The library of LED control messages is listed in Appendix 1.

Zone setup:

So that we can individually address each LED location on the fingerboard, the Clipper fingerboard outputs note-messages on MIDI channels 15 and 16 for communicating the fret locations to the on-board LED-controller electronics, and also to Ableton Live. Our Clipper Ableton script sends back notes on channels 15 and 16 to control the LEDs.

 [In the case of a 12-string fingerboard, four Zones are needed for independent access to each key as we put both Channels 13 and 14 into use here.]

Pay special attention to the Lighting Zone format in the FRETBOARD/TUNINGS/KEYMAP editor. The "Ableton" keymap is assigned to one or more active Zones which are assigned to Channels 15/16 (13/14 added for 12-string) and sent to the "Right"port for onboard lighting or "Left" port for the external computer via USB.

Some terminology regarding Zone Boundaries:

The Clipper is modeled after a guitar fingerboard which is described in terms of Strings and Frets.

- A String is comprised of the 24 consecutive keys in each of the 6 rows running the full length of the fingerboard.
- A Fret is comprised of the 6 consecutive keys in each of the 24 columns running across the width of the fingerboard.

The interactive Live setup is composed of at least three Zones on the fingerboard. [You may also want to add additional Zones for playing "Guitar" (Bass, Piano, Drums, etc.) from the fingerboard.]

1) One "Live"-Zone to set the boundaries of the Clipper's Redbox for Ableton. (Zone1 for example.)
2) Two Zones to send Clip-control commands to Ableton. (Zones 2 and 3 for example.)

Live Zone setup:

Open the Clipper Zone menu. The "Tuning/Transposition" sub-menu has a "Color-Zone" field that may be set to *"RedBox"* or *"Local", or "Off".*

Set the CZone to 'RedBox' for Ableton Live:
Setting the soft-switch "CZone" to something other than Off means that the Zone will now send to Ableton a **MIDI sysex** message that contains the Zone's boundaries and color information whenever those values are changed. Also, this Sysex message can be sent when the Clipper "Song" (onboard preset) is selected by setting "P&VSend=Y".

If *"RedBox"* is set, Zone Boundary data are automatically sent via Sysex to MIDI-Out-L(eft) which goes both to the MIDI Out jack and also the USB port.

Note: For simply setting a block of color to a Zone, you can set the CZone to 'Local' for assigning a background color to a fingerboard Zone.
If "Local" is set, Zone Boundary and Color-number data are sent via Sysex to MIDI-Out-R which is connected internally to the LED controller in the fingerboard.

Use the R/L switch to convert the color zones, or to tell Ableton whether you're playing Right-handed or Left-handed. Find this in the FRETBOARD > TUNE/TRANS > MORE menu.

The ***only*** function of this option is to adjust the R/L flag that's sent out in the sysex message -- it has no effect on fingerboard tuning. You now press the softkey to select the field, and then use the up and down buttons to change from R/L to L/R. Note that the left/right swap in the TUNING menu will ***not*** affect this flag.

Create the Zone boundaries, MIDI Channel, and transposition:

Example for the full 24-fret LiveZone:

Zone1: Redbox. *(Set only one RedBox Zone.)*
Low Boundary= String1, Fret1; High Boundary = String6,Fret24.
In FRETBOARD>TUNE/TRANS> set CZone=Redbox.
Keys: Off; MIDI Channel and R/L Port setting: doesn't matter.
In the FRETBOARD/MIDI menu set "P&V Send=Y" (Patch and Volume Send=Yes)
so that these settings are recalled when you select your Live preset.

Zone2: Low Boundary= String1, Fret1; High Boundary = String5, Fret24.
In FRETBOARD>TUNE/TRANS> set CZone=Off.
In FRETBOARD>TUNE/TRANS>TUNINGS> select Map 8.
(-cont-)

Keys: Tap; Set this Zone to MIDI Channel 15:Left.
In FRETBOARD>TUNE/TRANS>More set the Transposition: -24. (The reason for this is explained in the Keymap Appendix.)

Zone3: Low Boundary= String6, Fret1; High Boundary = String6, Fret24.
In FRETBOARD>TUNE/TRANS> set CZone=Off.
Keys: Tap; Set this Zone to MIDI Channel 16:Left.
In FRETBOARD>TUNE/TRANS>More set the Transposition: -24.
In FRETBOARD>TUNE/TRANS>TUNINGS> select Map 8.

KEYMAPS

When playing the fingerboard, the Clipper uses a special "keymap" that assigns a unique channel and note-number combination to each note-location on the fingerboard. This is done so that each LED in the LED array on the fingerboard can be lit independently without regard to a guitar tuning. When plugged into an outboard computer his same map is used to control individual screen-grid locations in applications like Ableton Live.

Keymaps are selected, edited, and stored in the Fretboard/Tunings Menu where a unique set of eight full maps may be saved for each "Song", or master preset in the Clipper. To activate a keymap you must assign a Map Number to a Fretboard Zone in the TUNINGS menu.

The Lighting keymap is factory-installed to Map#8 in all of the onboard Songs in the Clipper. *(It may also appear at Map #3 where it has been loaded for convenience when editing your setups.)*The fingerboard key-array is ordered starting with "String#1, Fret#1" which is the lowest fretted note on a guitar for a right-handed player. The highest note-location on String#1, Fret#24, is LED-location #24.

On the input side, the same range of messages on Channels 15 and 16 are used to address the individual LEDs. Also there are a variety of embedded functions in the Clipper that may be accessed by other Note- and CC-messages as listed in the Clipper LED Command sheet.

MIDI Merge settings

You can assign any of the following Merge settings to each of the MIDI ports independently.

1. *The Clipper has (2) full MIDI Ports, Left and Right.*
2. *The Left MIDI port Input is our standard MIDI In connector on the Clipper.*
3. *The Left MIDI port Output drives our normal MIDI Out connector and the USB Output.*
4. *The Right MIDI port Input is attached to the USB input.*
5. *The Right MIDI port Output drives the lighting-data to the LED controller board.*

Go the **UTILITIES/More>MIDI-IN** menu and set the Merge settings. *The RedBox Zone setting should make these settings for you automatically.]*

The Merge settings are made for the Left and Right MIDI Ports separately.

Merge: *Off/Notes/Sysex/Both/NotesX,SysexX/BothX. ("X" = "Crossed")*
Notes: Only merge incoming notes from the selected input port to the same corresponding output port.
            (MIDI Left-In to MIDI Left-Out or MIDI Right-In#2 to MIDI Right-Out.)
Sysex: Only merge incoming Sysex data to the selected output.
Both: Merge both Notes and Sysex data to the selected output.
NotesX: Merge only incoming Notes to the opposite MIDI Port number. (MIDI Left-In to MIDI
Right-Out or MIDI Right-In#2 to MIDI Left-Out.)
SysexX: Merge only incoming Sysex data to the opposite MIDI Port number.
BothX: Merge both Notes and Sysex data to the opposite output.

Note: **If _any_ zone is enabled as a Redbox zone, the MIDI Merge assignments will be set so that both MIDI In and USB In will communicate with the LEDs.**

That automatic Redbox Merge setup is as follows: (**MIDI Left Input = crossed, MIDI Right Input = normal**). This takes either the MIDI or USB inputs and routes them to the LED controller. If all zones are either "Off" or "Local", Merge is disabled.

*T*he automatic setting is only applied when you change the CZone type in the Fretboard > Tune/Trans menu, so once you have set up your zones, you can then go to the MIDI-RX menu and change the merge options manually and independently to the two ports if you wish. But the next time you change a CZone setting it will reset the merge options.

→ **If you are not using Ableton, or you simply want to play guitar on the Clipper and have the LEDs light up the keys that you play:**

Set the Fingerboard to use three Zones :

Zone1: (This will be your "music" zone for the notes that you hear.)
Low Boundary= String1, Fret1; High Boundary = String6, Fret24. (Set up to use the full fingerboard.)
In FRETBOARD>TUNE/TRANS> set CZone=Off.
In FRETBOARD>TUNE/TRANS>TUNINGS> select Map 1 or anything other than Map 8.
        Set up "GUITAR" or any tuning of your choice.
In FRETBOARD>MIDI set the MIDI channel to the Left Port and channel 1 or any channel other
        than 15 or 16.
In FRETBOARD set Keys: Tap or Fret depending on your choice.;

Zone2: Low Boundary= String1, Fret1; High Boundary = String5, Fret24.

  In FRETBOARD>TUNE/TRANS> set CZone=Local.

  In FRETBOARD>TUNE/TRANS>TUNINGS> select Map 8.

  Keys: Tap; Set this Zone to MIDI Channel 15:Left.

  In FRETBOARD>TUNE/TRANS>More set the Transposition: -24. (The reason for this is explained in the Keymap Appendix.)

  Zone3: Low Boundary= String6, Fret1; High Boundary = String6, Fret24.

  In FRETBOARD>TUNE/TRANS> set CZone=Local.

  Keys: Tap; Set this Zone to MIDI Channel 16:Left.

  In FRETBOARD>TUNE/TRANS>More set the Transposition: -24.

  In FRETBOARD>TUNE/TRANS>TUNINGS> select Map 8.

  ****

Color Settings

Ableton Live currently uses a color-picker from either the Mac or Windows OS that offers 60 color choices. These 60 values are mapped into our available 7-bit color value and selected from an embedded table in the Clipper. When the Live session starts the Ableton color-map is sent to the Clipper as the colors are used. Other color maps may be loaded into the Clipper as well, and several alternative maps are included.

Ableton Live carries internal variables that store all of its dynamic status including current color-status for each on-screen clip-location among other things. That information is available using calls to the Python-based LiveAPI. Using this information we are able to extract the dynamic clip/color status and output to the Clipper via MIDI data to update its LED array.

Setting the Color Number:

The Color number is used by the ClipperOS and the Clipper software to assign a given color to a zone of keys when playing the Clipper as a musical instrument, other than the Ableton session usage. Simply set the **CZone= Local** and change the color number to view the colors in your selected color map.

- *The Color-number adjustment in this menu is always visible but you cannot adjust it when in the Redbox mode. You need to have Czone:Local or Czone:Off in order to adjust the color value.*

Color-Zone Sysex Message format

When the Zone-boundaries are changed for the RedBox Zone the ClipperOS will send a Sysex to Ableton Live to reset the size of the RedBox to match. The following information is sent to Live:

Transmits the boundaries of zone 'z' (1-32) in the current song, as a Sysex message in the format:
   $F0 $00 $01 $42 pp nn nn ll ll ZZ LS LF HS HF RL CC csum $F7
  where:
    pp = packet type, (7-bit unpacked data) = $40
    nn nn = subtype number (14-bit)   = $00 $03
    ll ll = data length (14-bit)   = $07
    ZZ = zone number
    LS = lo string
    LF = lo fret
    HS = hi string
    HF = hi fret
    RL = R/L switch , 0=right, 1=left
    CC = color index (7-bit)

IMAGES

The Clipper currently can hold 10 full fingerboard-frame images which may be uploaded using the companion LED control software for the PC. Images are stored as MIDI SysEx files which may be dumped to the Clipper using any standard MIDI SysEx utility. The capacity will be upgraded to 32 images and more in the future.

TUNING MAPS

The Clipper stores Tuning maps which are used to represent "fixed" images that may reside on the fingerboard underneath any dynamic lighting effects that may be active. Generally these are used to display a tuning or other visual reference for notes on the fingerboard.

For 12-string instruments:
1. Clear: No background lighting.
2. Fret Marks: shows guitar fret marks at the standard fret locations.
3. Spectral fourths: 7 colors are assigned to the natural-notes, the white keys of the piano. The piano's black keys are defaulted to OFF. Straight Fourths tuning.
4. Spectral fourths/Minus-6 : 7 colors are assigned to the natural-notes, the white keys of the piano. The piano's black keys are defaulted to OFF. Straight Fourths tuning. Expands the range of the 12-string Clipper.
5. WhiteBlack Fourths: Two colors as on the Piano. Straight Fourths tuning starting at a low F natural.
6. WhiteBlack Fourths/Minus-6: Two colors as on the Piano. Straight Fourths tuning starting at a low B natural.

Using the proper MIDI commands, select the desired Tuning Map and then use the SetToBackground command.
Example [Set Piano-style with a Low B natural]:

NOTE-OFFS

The Clipper deals with note-off messages according to the mode of operation.

In normal run-mode, depending on assertion of the Enable-Background command, a Note-off corresponds to either the selection of Color-#0, or the pixel value of a previously set background. Color-#0 is normally Black(OFF), however any other color may be set into the Color#0 location of the Clipper by loading a new color map.

Fade-time: when the Fade-time value is set to >0, the MIDI Note-Off triggers a Fade-timer for each pixel location. When the Fade-timer expires the lit LED will extinguish.

Latch-Mode: When Latch-Mode is set the Note-Offs are ignored.

Charts and appendices

LED Fingerboard Command Specification

Special MIDI-Note Functions on Channel 16 :

[Notice that the minimum velocity value available for your use is the number 1. Velocity = 0 is reserved for the Note-Off function.]

Command	#	Comments
CLEAR FRAME	127	Ignore velocity, clears to 0 unless clear to background mode set
INIT ZONES	126	ignore velocity
REFRESH	125	ignore velocity
ENABLE ZONE	124	velocity = ZONE 1 - 31, bit6 1 = enable 1 = disable (Displays Zone lighting. Bring to foreground)
LEFTRIGHT	123	velocity 2 = LEFT, 1 = RIGHT, mirror about the Y-axis
SHIFT RIGHT	122	velocity offset 0:no effect, =>1:Shift Right one position Bit6=0: Rotate (Wrap) image, Bit6=1: Do not wrap image. Rotate Right: vel<64; Shift Right: vel =>64D
SHIFT LEFT	121	velocity offset 0:no effect, =>1:Shift Left one position Bit6=0: Rotate (Wrap) image, Bit6=1: Do not wrap image. Rotate Left: vel<64; ShiftLeft: vel =>64D
SCROLL UP	120	velocity offset: 0:no effect, =>1:Shift Up one position Bit6=0: Rotate (Wrap) image, Bit6=1: Do not wrap image. Rotate Up: vel<64; Shift Up: vel =>64D
SCROLL DN	119	velocity offset: 0:no effect, =>1:Shift Down one position Bit6=0: Rotate (Wrap) image, Bit6=1: Do not wrap image. Rotate Down: vel<64; Shift Down: vel =>64D
BLINK ON(upper 16)	118	velocity + 128 = led#
BLINK ON(lower 128)	117	velocity = led#
BLINK OFF(upper 16)	116	velocity + 128 = led#
BLINK OFF(lower 128)	115	velocity = led#
BRIGHT(upper 16)	114	velocity + 128 = led#
BRIGHT(lower 128)	113	velocity = led#

DIM(upper 16)	112	velocity + 128 = led#
DIM(lower 128)	111	velocity = led#
RedBox Scroll Up	110	Velocity: Ignore - not used by ledfb firmware, output to Ableton
RedBox Scroll Down	109	Velocity: Ignore - not used by ledfb firmware, output to Ableton
RedBox Scroll Right	108	Velocity: Ignore - not used by ledfb firmware, output to Ableton
RedBox Scroll Left	107	Velocity: Ignore - not used by ledfb firmware, output to Ableton
- Unused -	106	- unused -
REVERSE LEDS	105	Velocity: 0-64 = Normal, 65-127 = Reverse
TABLE TOP GUITAR	104	Mirror about X-axis. Velocity 7F=Tabletop: body is low LED number Velocity 00=Guitar: head is low LED number
Set Char (lower 128)	103	velocity = text character in lower 128 of ASCII table
Set Char (upper 128)	102	velocity + 128 = text character in upper 128 of ASCII
SET CHAR DIRECTION	101	velocity sets the direction Chars scroll from, 0 from head, >0 from body
SELECT SCRIPT [Available scripts are listed in a table below.]	100	Velocity selects a Script, 0 aborts. A Script is an embedded macro that can display images, text, and/or lighting effects. The opening splash screen is a Script. Any script is aborted by asserting Select-Script with a value=0.
ENABLE BACKGROUND	99	Enable Background: velocity 1 = clear to black, velocity 2 = Enable background. When an LED is turned on, then turned off according to some dynamic function, the off-state may be either an image saved to the background or simply black.
SET COLOR MAP	98	velocity is color map number; Max = 4. *See Pg.18, Maps*
SELECT IMAGE	97	velocity is the number of the image to display. Image# 0-31 are RAM, images 32-128 are ROM based images
SET STRING 6	96	Set all LEDs on a string to the same color. Velocity = color index
SET STRING 5	95	Velocity = color index
SET STRING 4	94	Velocity = color index
SET STRING 3	93	Velocity = color index
SET STRING 2	92	Velocity = color index
SET STRING 1	91	Velocity = color index
Display Tuning Map	90	Velocity = zone number*
Send To Background	89	Send the foreground image to the background. Verlocity is ignored.
Set Tuning	88	Pre-defined Tuning Maps are stored in ROM. The full list is provided below. Velocity = Tuning map number. Guitar, Piano, etc.

		1=ClearTuning, which clears the background. The full list is described on Page 11.
SCROLL-IMAGE	87	A selected image is scrolled over the short dimension of the fingerboard. Velocity = step timing for the Scroll delay.
SET FOREGROUND COLOR	86	Velocity = color map index# of the foreground color.
SET BACKGROUND COLOR	85	Velocity = color map index# of the background color.
Rotate RedBox 90°	84	Velocity: Ignore - not used by ledfb firmware, talks to Ableton. Rotates the Ableton RedBox
SYNC	83	Reset the Blink timer and other internal step-timers.

MIDI-Note Messages on Channel 16 continued:

OFFSET FOREGOUND COLOR INDEX	82	Velocity is added to all color selections for indexed-color pixels, i.e., foreground colors. If the resulting color-index exceeds the length of the color map, the value wraps around to the beginning of the table.
SET FRETMARKS OVERLAY COLOR	81	Velocity = color for the Fretmarks
ENABLE FRETMARKS OVERLAY	80	Velocity = 1 disables the Fretmarks. Velocity = 2 enables the Fretmarks

MIDI-CC Messages on Channel 16 (Channel 15- Unused):

Latch LEDs	64	Off < 65, On > 64; Once set, all LEDs will remain lit until the command is reset.

MIDI-CC Messages used by the LED fingerboard. Only on channel 16

BRIGHTNESS	7	Scale global brightness from Master Volume, Channel 16
SCALE RED	20	Adjust the brightness for all Red LEDs on the
SCALE GREEN	21	Adjust the brightness for all Green LEDs on the
SCALE BLUE	22	Adjust the brightness for all Blue LEDs on the
SCALE SATURATION	23	0-127; Scales the intensity of all colors. Applies to images, tuning

		maps, and foreground colors.
VU String 1 *	24	Fires all of the LEDs on String #1 in sequence (Data value= length of the lit-string. All LEDs are lit from Fret1 to the scaled data-level) CC-data values of 0-127 scale from no lights to all 24 columns lit. All LEDs will light from Fret1 up to the scaled data-value. Each Fret (column) corresponds to a data increment of a little more than 5.
VU String 2	25	Fires all of the LEDs on String #2 (in sequence, value= length) lights sequential number of frets according to MIDI data 0-127
VU String 3	26	Fires all of the LEDs on String #3 (in sequence, value= length) lights sequential number of frets according to 0-127
VU String 4	27	Fires all of the LEDs on String #4 (in sequence, value= length) lights sequential number of frets according to 0-127
VU String 5	28	Fires all of the LEDs on String #5 (in sequence, value= length) lights sequential number of frets according to 0-127
VU String 6	29	Fires all of the LEDs on String #6 (in sequence, value= length) lights sequential number of frets according to 0-127
VU All Strings	30	Fires all of the LEDs on all Strings (in sequence, value= length) lights sequential number of frets according to 0-127
VU Fret 1 **	31	Lights LEDs sequentially up fret 1 like a VU meter. CC-data values of 0-127 scale from no lights to all 6 rows lit in 12 increments. From bottom to top: Rows 1-4= green, Row5=yellow, Row6=Red. Each key lights at half-brightness for the lower half of its own range. Any given data value will light all keys from the bottom to the scaled level. Each lighting increment corresponds to a data increment of about 10 and a half. For example, a datum110 (decimal) lights Row6 (Red) at half-brightness, Row5 (Yellow) full-brightness, and Rows4-1 (Green) full-brightness.
VU Fret 2	32	Lights LEDs sequentially up fret 2 like a VU meter. Value of 0-127 scales from no lights to all lit, fading in brightness and color.
VU Fret 3	33	Lights LEDs sequentially up fret 3 like a VU meter. Value of 0-127 scales from no lights to all lit, fading in brightness and color.
VU Fret 4	34	Lights LEDs sequentially up fret 4 like a VU meter. Value of 0-127 scales from no lights to all lit, fading in brightness and color.
VU Fret 5	35	Lights LEDs sequentially up fret 5 like a VU meter. Value of 0-127 scales from no lights to all lit, fading in brightness and color.
VU Fret 6	36	Lights LEDs sequentially up fret 6 like a VU meter. Value of 0-127 scales from no lights to all lit, fading in brightness and color.
VU Fret 7	37	Lights LEDs sequentially up fret 7 like a VU meter. Value of 0-127 scales from no lights to all lit, fading in brightness and color.

VU Fret 8	38	Lights LEDs sequentially up fret 8 like a VU meter. Value of 0-127 scales from no lights to all lit, fading in brightness and color.
VU Fret 9	39	Lights LEDs sequentially up fret 9 like a VU meter. Value of 0-127 scales from no lights to all lit, fading in brightness and color.
VU Fret 10	40	Lights LEDs sequentially up fret 10 like a VU meter. Value of 0-127 scales from no lights to all lit, fading in brightness and color.
VU Fret 11	41	Lights LEDs sequentially up fret 11 like a VU meter. Value of 0-127 scales from no lights to all lit, fading in brightness and color.
VU Fret 12	42	Lights LEDs sequentially up fret 12 like a VU meter. Value of 0-127 scales from no lights to all lit, fading in brightness and color.
VU Fret 13	43	Lights LEDs sequentially up fret 13 like a VU meter. Value of 0-127 scales from no lights to all lit, fading in brightness and color.
VU Fret 14	44	Lights LEDs sequentially up fret 14 like a VU meter. Value of 0-127 scales from no lights to all lit, fading in brightness and color.
VU Fret 15	45	Lights LEDs sequentially up fret 15 like a VU meter. Value of 0-127 scales from no lights to all lit, fading in brightness and color.
VU Fret 16	46	Lights LEDs sequentially up fret 16 like a VU meter. Value of 0-127 scales from no lights to all lit, fading in brightness and color.
VU Fret 17	47	Lights LEDs sequentially up fret 17 like a VU meter. Value of 0-127 scales from no lights to all lit, fading in brightness and color.
VU Fret 18	48	Lights LEDs sequentially up fret 18 like a VU meter. Value of 0-127 scales from no lights to all lit, fading in brightness and color.
VU Fret 19	49	Lights LEDs sequentially up fret 19 like a VU meter. Value of 0-127 scales from no lights to all lit, fading in brightness and color.
VU Fret 20	50	Lights LEDs sequentially up fret 20 like a VU meter. Value of 0-127 scales from no lights to all lit, fading in brightness and color.
VU Fret 21	51	Lights LEDs sequentially up fret 21 like a VU meter. Value of 0-127 scales from no lights to all lit, fading in brightness and color.
VU Fret 22	52	Lights LEDs sequentially up fret 22 like a VU meter. Value of 0-127 scales from no lights to all lit, fading in brightness and color.
VU Fret 23	53	Lights LEDs sequentially up fret 23 like a VU meter. Value of 0-127 scales from no lights to all lit, fading in brightness and color.

VU Fret 24	54	Lights LEDs sequentially up fret 24 like a VU meter. Value of 0-127 scales from no lights to all lit, fading in brightness and color.
LATCH MODE	64	(MIDI Sustain) When Latch Mode is On, incoming Note-Off messages will be ignored rather than turning off their intended LEDs. Velocity < 65 = Off, velocity > 64 = On.
FADE TIME	72	CC-data sets the duration the LEDs remain illuminated after being lit from a Note-On event. Similar to a synthesizser's ADSR Release time.
SET-SCRIPT-CLOCK-COARSE	75	Sets the coarse timing delay for clocked script-looping functions like the Swirl effect in 10 millisecond increments. Delay = velocity x .01 seconds.
SET-SCRIPT-CLOCK-FINE	76	Sets the fine timing for clocked script-looping functions like the Swirl effect in 0.1 millisecond increments. Delay = velocity x .0001 seconds. The coarse and fine values are additive.
STORE-RAM-IMAGE	77	The dynamic LED data frame buffer is copied to a RAM image location. Velocity = Image number (v = 0-9) The duration of the lights from pressed keys; similar to a synth "release time".

Clipper Setup Charts

Embedded lighting Scripts:
[Set-script command: Note 100]

0: Null script
1: Abort Script. Aborts any active script.
2: Grand Demo. Combines several scripts and the Starr Labs sign-on screen.
3: Product_ID. Current Clipper OS version#.
4: Null
5: Spiral Effect. Single shot event.
6: Spiral Script. Loops the spiral effect.
7: Swirl Effect. Single shot event.
8: Swirl Script. Loops the swirl effect.
9: Boustrofedon. Serpentine effect.
10: Boustrofedon Script. Loops the single shot event.
11: Show current color map.
12: Scroll RAM Images.
13: Scroll ROM images.
14: Cylon script

Embedded Color Maps:

[One of these Color Maps is loaded into the default 127-location Color Map table. When you apply a MIDI note to light up a location on the fingerboard, the velocity selects a color from the default table.]

0: Null Map, blank
1: Ableton Live color map. 60 colors. Duplicated to (mostly) fill the 127 location map.
2: Spectral color map. Full color in the traditional rainbow sequence.
3: Tertiary color map. The basic 12 colors resulting from the simple combinations of the R,G, and B LEDs.
4: Velocity color map. A four-color map to clearly distinguish MIDI velocity levels.

TUNING MAPS: [maps of 7/7/17]

1. Clear: No background lighting.
2. Fret Marks: shows guitar fret marks at the standard fret locations.
3. Spectral fourths: 7 colors are assigned to the natural-notes, the white keys of the piano. The piano's black keys are defaulted to OFF. Straight Fourths tuning.
4. Spectral fourths/Minus-6 : 7 colors are assigned to the natural-notes, the white keys of the piano. The piano's black keys are defaulted to OFF. Straight Fourths tuning. Expands the range of the 12-string Clipper.
5. WhiteBlack Fourths: Two colors as on the Piano. Straight Fourths tuning starting at a low F natural.
6. WhiteBlack Fourths/Minus-6: Two colors as on the Piano. Straight Fourths tuning starting at a low B natural.

Using the MIDI command Note 88/velocityN, select the desired Tuning Map and then use the SetToBackground command. DisplayTuningMap Note- 90 (only needed one time)

Here is an example to set up a 12-sttring Clipper with a Spectral Tuning (rainbow colors for the natural notes.).

```
Refer to the manual to see the command chart where these commands reside.

Low 6 strings:

Set the Tuning map to Spectral 4ths/ -6 (Low B):
Go to an unused Sensor-Pad.
Set the Pad to Notes-Basic or -All.
In Event #1 set it to Channel 14, Note 88 and for the velocity, instead of
"PAD" set it to 4.
In the Output screen, set it to MIDI: R
Press the Pad.
If it doesn't take at first try setting the pad to Note 90 and press the Pad..
Repeat if necessary.

For the high 6 strings:

Do the exact same thing as for the Lower 6 strings, but for two changes:
1)change the MIDI channel to 16.
2)change the TuningMap selection to Note88-Velocity 3.
```

LIGHTING SETUPS

These are a few things you can set up for your Clipper to create some interesting lighting effects. You can create these setups in the Sensor menu or if you have an external MIDI controller or sequencer sending commands to the Clipper you can have virtually unlimited variety in the effects you choose.

Spread the lights up the length of the fingerboard with the velocity from a Pad:

Here's a cool thing you can do with a Drum Pad that you can link to the sound or not as you prefer.
You're going to set a pad -hit to blast the lights up the fingerboard

Set Pad1:
SENSORS/EVENTS menu
Pad1
PlayFunction: All
Event1: ControlChange, Channel 15:CC#24 - (VU String #1)
Event2: ControlChange, Channel 15:CC#25 - (VU String #2)
Event3: ControlChange, Channel 15:CC#26 - (VU String #3)
Event4: ControlChange, Channel 15:CC#27 - (VU String #4)
Event5: ControlChange, Channel 15:CC#28 - (VU String #5)
Event6: ControlChange, Channel 15:CC#29 - (VU String #6)
Sensors/Output Screen: Set the Right Port [Lights]

TIMING screen options:
If HitMode= TOUCH the harder you hit the Pad, the farther the Lights travel up the fingerboard.
If HitMode= TIMED, you can set it to Loop-repeat a few times for more effect.

You can change the Pad's curve or simply give the Pad-hit a fixed velocity value if you want to always light up a fixed amount of LEDs. For instance, always lighting all the LEDs on a string would be done with a velocity Fixed at 127.

Use this same idea to light up an entire fret by using the VU Fret commands 1-24 (Ch 16: CC#31-54)

This is a cool effect using a pitchwheel or joystick too.

LIGHT UP an entire string at once:

This is similar to the VUFret command but differs in that the entire string always lights and the velocity now modifies the color value.

Light the entire fingerboard with one command from a Pad-hit.
PlayFunction: Basic
Event1: Note Channel16; Note 86; Velocity = Color (1 = Red)

SETUP a Pad to cycle between various effects:

PlayFunction: ALT-UP (a Step-sequence of up to 8 events. Each time you hit the Pad, you step to a new Event)
Event1**: Load image;** Note: Ch.16; Note#97; Velocity = image number
Event2: **Load image;** Note: Ch.16; Note#97; Velocity = image number
Event3: **Full Fingerboard Color;** Note Channel16; Note 86; Velocity = Color (1 = Red)
Event4: **Full Fingerboard Color;** Note Channel16; Note 86; Velocity = Color (2 = Orange)
Event5: F**ull Fingerboard Color;** Note Channel16; Note 86; Velocity = Color (3 = Purple)
Event6: **Full Fingerboard Color**; Note Channel16; Note 86; Velocity = Color (4 = Cyan)
Event7: **Full Fingerboard Color**; Note Channel16; Note 86; Velocity = Color (5 = Green)
Event8: **Clear Fingerboard**; Note Channel16; Note 127; Velocity is ignored

Create a Dancing Pattern:

PlayFunction: Notes sequential (a Step-sequence of up to 8 events. Each time you hit the Pad, to fire all of the events in the sequence. Hit the Pad again to toggle it OFF)
Event1**: Shift Right 2 spaces;** Note; Ch.16; Note 122; Velocity = 2
Event2: **Scroll Up 1 space;** Note; Ch.16; Note 120; Velocity = 1
Event3: **Shift Left 1 spaces;** Note; Channel16; Note 121; Velocity = 1
Event4: **Scroll Down 1 space;** Note; Channel16; Note 119; Velocity = 2
TIMING menu: HitMode= Timed; LOOP=ON

SCROLL the Lights down the fingerboard in streaming fashion:
SENSORS/EVENTS menu
Pad1
PlayFunction: All
Event1: Note, Channel 16:Note#122- (Shift Right; Velocity=1 shifts one space)
TIMING menu: Set HitMode = Timed; LOOP=ON.

Run a Lighting Script, chase the LEDs around the fingerboard:
SENSORS/EVENTS menu
Pad1
PlayFunction: All
Event1: Channel 15 (also 13 for the 12string) Note 100, velocity 6. This loops the Spiral Effect.

In the SENSOR/Events menu set up two knobs or pedals (breath, etc…) with Channel 16 MIDI CC-numbers 75 and 76 which are the Coarse and Fine settings for the Script-Clock rate, or the speed of the Lighting effect.

Abort the script with a Note 100/Velocity-0 command.

Add your own Lighting Setups here:

Appendix 1

Setting up Live in the Mac to work with the Clipper

1. Load Live
2. Load or create your session clips and other EFX mappings
3. Copy the Clipper-Live Python script into your Control Surfaces folder.
4. Go to " LIVE > Preferences" and select the "Clipper-MIDI" as your controller.
5. In "MIDI Input" tab, select "Starr Labs MIDI"
6. In "MIDI I Output" tab, select "Starr Labs MIDI"

Appendix 2

Ableton Live Operation Notes:

Button Flashing: Ableton buttons flash when they are Armed, that is, the clip is ready to Play or Record but is waiting for the actual Start-time, to come around before firing. The start time is usually the "downbeat" but other Launch Quantizations are available.

Recording Mode:

When recording to a Clip, color is Red. We aim to represent the status of the little Clip arrow at the left of the Scene (row), which is sometimes solid and sometimes blinking.
The fingerboard Clip button becomes Red and is steady-state or flashing depending on the Recording-status.

1. Recording may be started by pressing a Record button in which case the recording starts immediately and the arrow glows solid Red.
2. Recording may be "Armed" with a 2-bar count-in where the LED flashes and when the recording actually begins the LED glows solid Red.

Playback Mode:

The fingerboard Clip button is steady-state or flashing depending on the Playback-status. The Ableton Record/Playback Arrow shows the status.

1. Playback may be started by pressing the play button in which case the playback starts immediately and the button glows at full intensity with the assigned color.

2. Playback may be "Armed" with a 2-bar count-in where the LED flashes and when the recording actually begins the LED glows its solid color.

Clip Launch modes:
(These are setups performed directly in Ableton Live's Clip Launch Mode setup dialog)

Trigger - once the button is pressed it latches and plays the clip until it reaches it's end or a **Stop** button is pressed. This will usually be the stop clip button at the bottom of the track, but can be any blank clip with an active stop button in that track. *Note-offs are ignored.*

Gate - F*ull intensity when active.* Momentary operation. You must hold down the button to play the clip. This is handled by Note-On on button-press and Note-Off issued button-release. Normal keyboard mode.

Toggle - *Full intensity when active.* Press once for the clip to play and repeat, press again for it to stop. Ignore Note-Off. Successive Note-On's toggle the function.

Repeat - mouse down/note on repeatedly plays the clip, retriggering at the time interval of the clip's launch quantization. When released the clip plays out like Trigger mode.

Appendix 3

MIDI message Format:

Each MIDI channel can address 128 note-numbers.

A full MIDI Note-On message is composed of (3) 7-bit bytes.

Byte #:1 Status byte (in binary) 0101nnnn where 09h = MIDI Note-On, and nnnn is a 4-bit channel number.

Byte #2: MIDI Note-number, 0nnnnnnn, where nnnnnnn is a 7-bit note-number. This is used to address each LED location. Each key on the fingerboard has a unique channel/note address.

Send MIDI Note-Off to turn off an individual LED. Both a literal Note-Off and a Note-On velocity=0.
Byte #3: Velocity value, 0nnnnnnn, where nnnnnnn is a 7-bit value. This normally encodes the force of a key-strike. In our case this is used to encode the color-value for each LED pixel.

Appendix 4

How colors are selected:

A full frame of color data assigns a 7-bit value to each fingerboard location. (144) 7-bit bytes.
These color values are index-numbers into an array on the device that holds a list of values to create 128 unique colors (including OFF = 0 = no light)*.
*Note there are two ways to turn OFF an LED.
Additionally, once a MIDI status byte has been issued from a MIDI controller, the channel number is then established, and most controllers will enter what is called **"Running Status"** which dispenses with the status byte and sends only 2-byte messages containing just the note-number and velocity information. When the channel-number must be changed or a different type of message must be sent, then the controller drops out of Running Status, sends a new Status byte as necessary, and re-enters Running Status. In the case where there are more than 127 LEDs in the array, a channel switch will be necessary to address the full array.

Create the special and extremely important Lighting Keymap : *[this is pre-loaded into the RAM of your Clipper but if you re-initialize the instrument or lose your onboard backup battery, you will lose this keymap as well. Always keep a sysex image of your Clipper presets (Songs in Ztar lingo).]*

[a Keymap is a map showing the note-assignment to every key on the Clipper fingerboard. When you set the TUNING for a Zone, you are creating a new Keymap. Each Zone may have its own keymap. A very powerful feature that enables chording among other things. The Map# for each Zone is shown in the Zone>Key/Trans>Tunings menu. Hit the TUNINGS/EDIT softkey to edit the selected map by individual keys.]

- In our LED fingerboard instruments, this has been created as a default and you should not need to create this map yourself. It is also available as a Sysex file on our website.

String#1,Fret#1- Fret#24 : LED locations 1-24 ; Channel 15 MIDI Notes 24-47(*)
String#2,Fret#1- Fret#24 : LED locations 25-48 ; Channel 15 MIDI Notes 48-71
String#3,Fret#1- Fret#24 : LED locations 49-72; Channel 15 MIDI Notes 72-95
String#4,Fret#1- Fret#24 : LED locations 73-96; Channel 15 MIDI Notes 96-119
String#5,Fret#1- Fret#24 : LED locations 97-120; Channel 15 MIDI Notes 120-144(**)
String#6,Fret#1- Fret#24 : LED locations 121-144; Channel 16 MIDI Notes 24-47 (*)

*Due to the way Octaves are implemented on the Clipper, in order to enter a MIDI Note#-0 (for any channel), you must set Note-24 as the lowest available note#, then transpose the ZONE down by two octaves, or -24 half-steps to offset the entire keymap to start at MIDI Note-0..

**While ordinarily "127" (decimal) is the highest allowable note# in MIDI, because of the aforementioned reason, the top note on String#5 becomes "144" and the Zone-offset of -24 adjusts the top-note back to a correct value of "120".

<div style="border:1px solid blue; background:#e0e0e0;">

Terminology

Color Map: A defined palette of colors.

Indexed Color: An arbitrary palette of colors whose various colors may be selected by an index number. Color Map Index: A number that selects a given color in a color map. In the case of the Clipper the indexed color map consists of 127 colors and the MIDI velocity is the color-index. The Ableton Live color map is an indexed color map of just 60 colors.

Direct Color: The color hue is directly defined by its component Red, Green, and Blue intensity values. The Clipper uses 15-bit color resolution which may define 32,000 different colors. 127 of these colors are loaded into the Clipper's color index and are selected by MIDI velocity data.

</div>

TIP and Notes:[TBD]
Clearing a local color-zone
12-string Clipper uses MIDI Channels 13,14,15,16 to interface to the LED array.

Clipper Operation Notes and Bugs:

The Ableton Live color-map from your computer is not a contiguous spectral array of colors but is pretty random in its color assignments. When these colors are selected by sending ordinary MIDI note-messages the colors may appear random. A linear spectral map is included in the Clipper.

Sometimes when switching between songs with different RedBox dimensions some of the LEDs won't refresh. Usually pressing the song selection button a couple extra times will remedy this.

Sometimes your MIDI Merge setting will get messed up even though you don't remember changing them. This will become apparent if you stop seeing visual feedback from the computer on the fingerboard LEDs. Double check your Merge settings if you lose communication from Live to your Clipper.

In the Events screen sometimes the Velocity value does not appear when setting the Event. Scroll to a different pad and back to revive the velocity field.

Index: