MidiSizer

Got MIDI?

MidiALF Manual

Firmware version: 0.95

There are two types of UI pages: top level pages and command pages. The top level pages are:

- note/mute
- velocity/skip
- gate/legato
- cc1/send
- cc2/send
- sequence settings
- progression settings
- system settings
- CC LFO1
- CC LFO2

Top level pages can be selected by rotating EncB at the right of the screen. First 8 top level pages can also be selected directly by pressing one of the 8 step switches while holding down SEL button (at the bottom on the right side).

Command pages are shown by selecting a command from the Command page that could be activated by pressing and releasing SEL button alone. These are the currently available command pages:

- Randomize
- Rotate
- Initialization
- Load
- Save

- Send sysex

Load and Save pages could be invoked directly by pressing SEQ and SEL+SEQ buttons respectively.

(https://midisizer.files.wordpress.com/2013/01/midialf_cmd.jpg)

When the top level page is shown, rotating EncA at the left of the LCD screen changes all steps of the currently selected sequence and clicking EncA allows copying of the currently selected sequence to another one. When command page is shown, rotating EncA results in page specific action, however clicking it closes the currently shown page returning to the previously shown top level page (aka Escape).

Rotating EncB at the right of the LCD screen while the top level page is shown cycles through top level pages and clicking EncB shows the Note page or the Sequence Settings page if the Note page is already shown. When command page is shown, rotating EncB typically selects an item while clicking it carries out the page specific action (aka Enter).

Global keys

EncB rotation cycles between the top level pages: Note, Velocity, Gate, CC1/CC2, three Settings pages and two LFO pages.

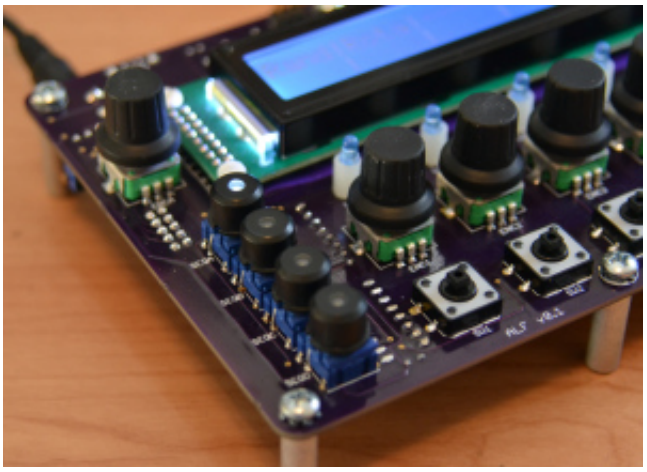
EncA click shows Copy Sequence page if a top level page is shown, or closes the currently shown command page.

EncB click shows the Note page or Sequence Settings page if the Note page is already shown.

Step Encoder rotation changes the step value.

Step Encoder click “plays” the step sending it’s note and CCs to MIDI OUT.

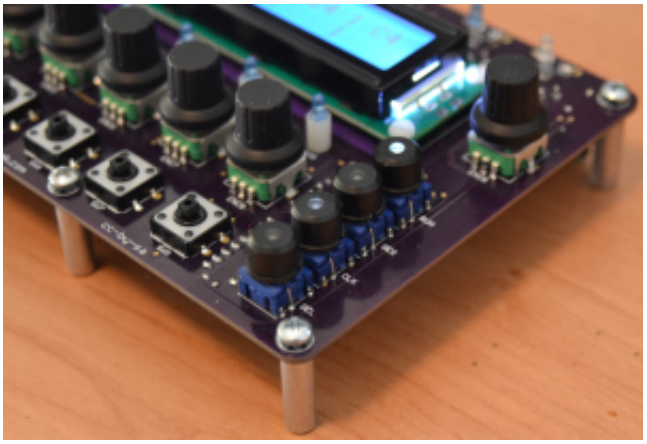
Left side switches select sequence A, B, C or D.


(https://midisizer.files.wordpress.com/2013/01/midialf_left.jpg)

RUN starts and stops sequencer.

SEQ shows the Load page.

CLK cycles Clock Division slowing down the sequence by 2 or 4 times.


(https://midisizer.files.wordpress.com/2013/01/midialf_right.jpg)

If the SEL switch is held down, the following switches' action is altered:

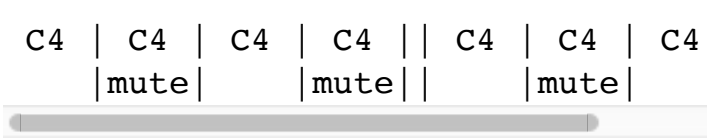
– SEQA sets set Link Mode to None;
– SEQB sets Link Mode to 16 and selects sequence A;
– SEQC sets Link Mode to 16 and selects sequence C;
– SEQD sets Link Mode to 32;
– RUN toggles Recording mode;
– SEQ shows the Save page.
– CLK toggles "step hold" mode.

Pressing the step switches 1 to 8 while holding down SEL switch shows Note, Velo, Gate, CC1/CC2 and settings pages.

Rotating encoder while holding down SEL switch increases increment to 10 or an octave.

Lone SEL switch shows Command page.

Note page

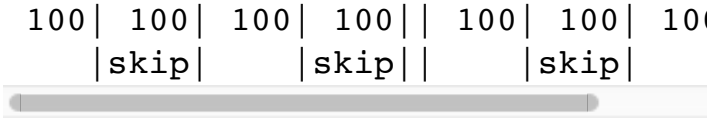


EncA transposes notes for all steps.

Step Encoders transpose the step note.

Step Switches toggle “mute” option for the step.

Velocity page

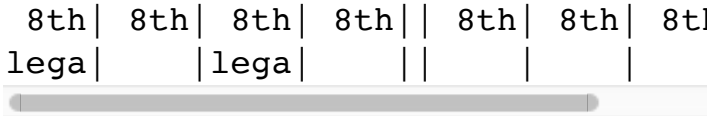


EncA changes velocity for all steps.

Step Encoders change the step velocity.

Step Switches toggle “skip” option for the step.

Gate page



EncA changes gate duration for all steps.

Step Encoders change the step duration.

Step Switches toggle “legato” option for the step.

CC1/CC2 pages

```
100| 100| 100| 100|| 100| 100| 10
cc1| cc1| cc1| cc1|| cc1| cc1| cc
```

EncA changes CC value for all steps.

Step Encoders change CC value for the step.

Step Switches toggle “send” option for the step.

Sequence settings page

```
Temp| Div| Grv|Amnt||Root| CC1| CC
 120| 8th|swng| 0  || C4 | 14 | 15
```

|Temp| – sets sequencer tempo from 25 to 250 beats per minute.

| Div| – sets sequencer step duration (from 64^{th} to 2 bars).

| Grv | – controls type of shuffle/jitter to apply to the clock, same as on Mutable Instruments MIDIPal (http://mutable-instruments.net/midipal).

|Amnt| – controls the amount of shuffle/jitter to apply to the clock.

|Root| – sets root note of the sequence. Press the step switch to “learn” last received note.

|CC1| – controls CC1 number. Press the step switch to “learn” last received CC number.

|CC2| – controls CC2 number. Press the step switch to “learn” last received CC number.

|Chan| – controls sequencer MIDI channel, 1-16.

Clicking step encoder on any of the settings pages restores default setting value.

Progression settings page

```
 Dir|Forw|Back|Repl||Intv|Rept|Ski
forw|  0   |  0   |  0   ||  0   |  0   |  0
```

|Dir| – controls sequence direction: forward, backward, pendulum, random.

|Forw| – controls how many steps to go forward before jumping back or replaying.

|Back| – controls how many steps to jump back.

|Repl| – controls how many times to replay the steps above.

|Intv| – controls interval for the repeat and skip parameters.

|Rept| – controls how many times to repeat the steps in the interval.

|Skip| – controls how many steps to skip after the interval.

|Link| – sets sequence link mode: 4×8, 2×16 or 1×32.

Nonlinear sequence progression parameters are borrowed from MIDIBox SEQ v4 (http://www.ucapps.de/). Thank you, Thorsten, these are a lot of fun!

System settings page

```
 Clk|SeqX|Xpos|----||PrgC|CtlC|---
 int|  imm|note|    ||none|none|
```

|Clk| – controls clock source: internal/external.

|SeqX| – sequence switch mode: immediate/on sequence end.

|Xpos| – sequence transposition mode: none/on note on (v0.95).

|PrgC| – controls Program Change handling: none, receive, send, receive & send.

|CtlC| – controls Control Change handling: none, receive, send, receive & send.

Note: Control Change Send functionality is not implemented in v0.92

LFO1/LFO2 pages

```
LFO1| CC |Reso|Amnt||Cent|Rate|Syn
 tri|none| 64 | 0  || 63 | 16t|fre
```

|LFOx| – controls LFO waveform, see Mutable Instruments MIDIPal (http://mutable-instruments.net/midipal) CC LFO app for details.

| CC | – controls LFO CC number. Press the step switch to "learn" last received CC number.

|Reso| – controls LFO resolution.

|Amnt| – controls LFO amount.

|Cent| – controls LFO center.

|Rate| – controls LFO rate.

|Sync| – controls LFO synchronization: free running, reset on start, on sequence start, or on sequence step.

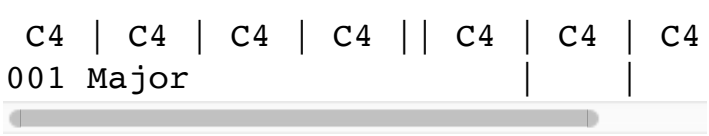
Command page

```
Rand|Rota|----|----||Init|Load|Sav
    |    |    |    ||    |    |
```

This page is shown when the SEL button is pressed and released without pressing any other switches.

Step Encoder click selects the command.

Randomize page



This page is shown when Rand command is selected on the Command page.

EncA transposes notes for all steps according to selected scale.

EncA click shows the previous page or sets all notes to the root note if SEL is down.

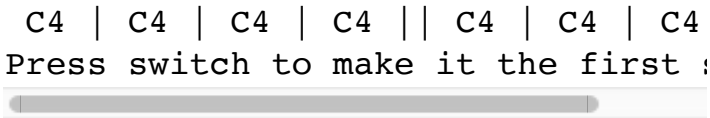
EncB selects a scale, scales collection is courtesy of MIDIBox.org (http://www.midibox.org).

EncB click randomizes sequence notes according to the selected scale or shuffles existing note order if SEL switch is down.

Step Encoders transpose the step note within selected scale.

Step Switches randomize step note according to the selected scale.

Rotate page



This page is shown when Rota command is selected on the Command page.

EncA transposes notes for all steps.

EncB shifts steps left or right.

Step Encoders transpose the step note.

Step Switches shift sequence so that it starts with the step.

Initialization page

```
Initialize:
001 Berlin theme          [Seq][Pgm
```

This page is shown when Init command is selected on the Command page.

EncB selects program slot.

EncB click initializes selected program slot.

Enc6 click initializes currently loaded sequence.

Enc7 click initializes currently loaded program.

Enc8 click initializes all programs on the device.

Load page

```
Load program from:
001 Berlin theme
```

This page is shown when Load command is selected on the Command page or SEQ button is pressed.

EncB selects program slot to load.

EncB click loads the selected program.

Save page

```
Save program 'Flashback2'
001 Berlin theme
```

This page is shown when Save command is selected on the Command page or SEQ button is pressed while holding down SEL button.

EncA changes name editing cursor position.

EncB selects program slot to save to.

EncB click saves the selected program.

Enc1 click moves cursor to the left.

Enc2 click moves cursor to the right.

Enc1 changes current editing character.

Enc2 changes current editing character in uppercase letter range.

Enc3 changes current editing character in lowercase letter range.

Enc4 changes current editing character in digit range.

Enc8 copies current sequence name into the slot name.

Step switch 1 toggles between space and the original character.

Step switch 2 toggles between previous and the original character.

Step switch 3 toggles between upper and lowercase character.

Step switch 7 inserts a character at the cursor position.

Step switch 8 deletes a character at the cursor position.

Send sysex page

```
Send sysex with:
001 Berlin theme          [Seq][Pgm
```

This page is shown when SysX command is selected on the Command page.

EncB selects the program slot.

EncB click sends the selected program slot.

Enc6 click sends currently selected sequence.

Enc7 click sends currently selected program.

Enc8 click sends all programs on the device.

Advanced features

Control Change messages

MidiALF recognizes the following control messages:

CC	Value	Action
106	0-3	Set sequence
107	0-3	Set direction
108	0-3	Set link mode
109	0-2	Set clock division
110	> 0	Increase tempo
111	> 0	Decrease tempo
112	BPM / 2	Set tempo
113	> 0	Toggle run mode
114	> 0	Stop sequencer
115	> 0	Start sequencer
116	> 0	Toggle recording mode
117	> 0	Stop recording
118	> 0	Start recording

System exclusive message format

[TBD]

Firmware update

MidiALF firmware update steps:

1. Backup your sequences to sysex file using Send Sysex page [All]
2. Connect your MIDI interface output to MidiALF MIDI IN
3. Power on MidiALF while holding down the EncoderA. MIDI LEDs will blink shortly then MIDI IN LED will stay steadily lit. The display will remain blank or contain a line of box characters.
4. Send firmware update sysex file (*.syx) to your MIDI interface port with 250ms delay between sysex buffers. Elektron's C6 works great on OS X and Windows. Classic MIDI-OX is also good on Windows. MIDI LEDs will be blink while firmware sysex is being received.

Firmware update takes a couple of minutes, MidiALF will automatically reset when the update is done. Check the version number display on MidiALF startup page.

Most firmware updates do not require MidiALF memory to be reset. However, if you don't find your parts and songs after the firmware upgrade, please restore them by sending the sysex back up you created at step 1 to MidiALF MIDI IN.

Device reset

Power on MidiALF while holding down EncoderB to reset it to initial state. Note that this will wipe out device memory without any confirmation. Use this to restore MidiALF operation as the last resort.

44 thoughts on "MidiALF Manual"

Mark Jarzewiak says:
on January 24, 2013 at 5:41 pm
Stellar work Peter!

Reply
John donson says:
on June 26, 2013 at 1:15 am

I think this sequencer is really nice, i think i will ask for pcb and atmega, anyway I always miss a chord mode

Reply

kvitekp says:
on June 26, 2013 at 6:05 am
Thank you.

What exactly is the chord mode?

Reply

Saddox says:
on November 10, 2013 at 11:05 pm
Chord mode.. like on some old synthesizers: Press "D3" and you get a D-minor and so on (or set to major you get a D-major). Nice to have sometimes.

kvitekp says:
on November 11, 2013 at 6:30 am
Yeah, that may be useful sometimes. However, it's not clear how this is related to the step sequencer.

Andrew Scheidler says:
on August 31, 2013 at 5:18 pm
This looks like a great sequencer, I think I need to build this 🙂
Would it be possible for this design to be expanded to *two* displays and sets of knobs for a full 16 steps of display/control? (or maybe the CPU has all its pins used up?)
Another project to build! Andrew

Reply

kvitekp says:
on August 31, 2013 at 11:54 pm
I doubt … it's much easier to use two MidiALF's synced to the same clock.

Reply

kvitekp says:
on September 8, 2013 at 5:56 am
Not really, the MCU is pretty busy serving 8 steps. However, i'm planning to experiment with "expanded" mode when 2 MidiALFs are chained in series each skipping every other cycle.

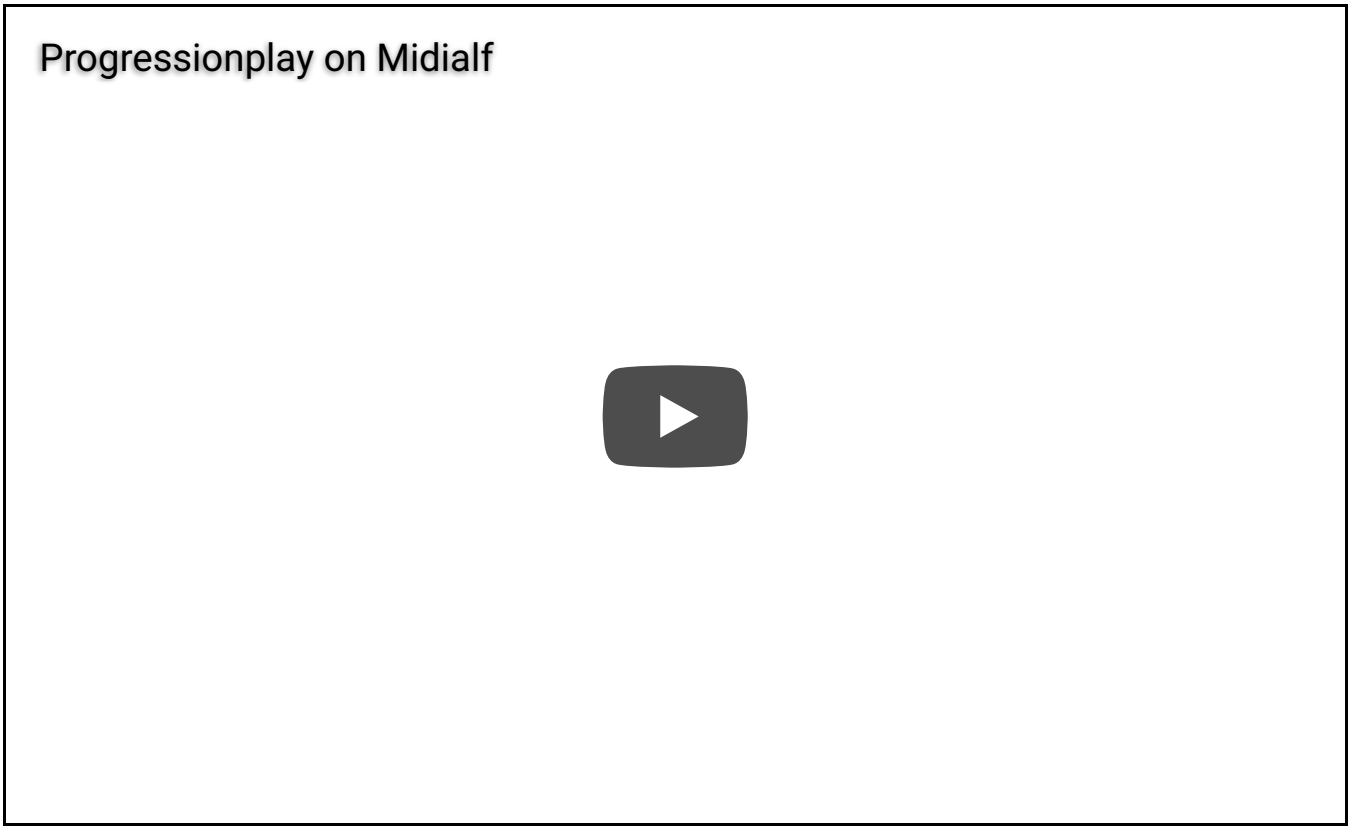
Reply

borfo says:
on January 5, 2014 at 10:58 am

> It'd be great if this "expanded mode" could allow two Midialfs to run two sequences and control two synths simultaneously – instead of skipping every other cycle, they could alternate between serving the two sequences…

saddox says:
on November 11, 2013 at 6:36 am
It is just sending a chord instead pf a note. Maybe not the most important feature but fun

Example when sending note to synthesizer with three oscillators playing a chord.



Reply

> kvitekp says:
> on November 11, 2013 at 6:47 am
> Sounds like an interesting feature for my MidiMate project.
>
> Reply

Peter Ullrich says:
on January 1, 2014 at 2:05 pm
Hi Pete! My MIDIAlf works fine, the only disturbing behavior for me is that switching between sequences takes place suddenly and not after finishing the current sequence (like on the GORF sequencer and also in Ableton live). Would it be possible to implement this mode so that each sequence is played complete and then the change takes place?
Or is this feature already implemented and I just over read this mode setting?

Reply

kvitekp says:
on January 3, 2014 at 6:11 am
I'll implement quantized sequence switching in one of the next versions, thank you for the suggestion.

Reply

borfo says:
on January 5, 2014 at 11:00 am
I like the instant switch between sequences – maybe this could be set up as an option, so users could have it whichever way they prefer?

kvitekp says:
on January 5, 2014 at 11:50 am
This option will be configurable.

Peter Ullrich says:
on January 5, 2014 at 1:09 pm
Thanks Peter, configurable switching behavior sounds perfect!

Reply

borfo says:
on January 6, 2014 at 11:00 am
Posted this in the forum a few days ago, but maybe this is a better place for it since the forum is pretty quiet.

Feature request: Restrict to Key/Scale
It'd be great if there could be a global setting key/scale setting that would restrict MIDIalf to the selected key/scale – pretty much the way things work when you're in randomize mode… If enabled, no matter what mode you're in, turning the encoders would only produce in-key notes.

Another feature that could be added if this was enabled is a "note shift within key" function – a mode where pressing one of the encoder buttons or one of the bottom buttons would shift the pitch of each note in the pattern up or down by a certain number of steps within the key – ie: if you're in Cmajor with a pattern CCCDCCCE, pressing button 2 would change the pattern to DDDEDDDF, pressing button 5 would change it to GGGAGGGB, and pressing button 1 again would bring it back to CCCDCCCE.

Maybe this note shift function could be activated by pressing and holding the right encoder button + one of the bottom buttons. While the right encoder button was pressed, the step indicator LEDs could indicate the current note shift.

Reply

kvitekp says:

on January 7, 2014 at 7:49 am
Sounds interesting, however introducing this kind of global “states” in the UI makes it harder to understand. I’ll see what can be done.

Reply

borfo says:
on January 7, 2014 at 8:57 am
Cool. Thanks.

borfo says:
on January 9, 2014 at 10:06 pm
One more feature idea… Loading and switching to a saved pattern happens pretty fast, but the UI’s a bit tough to get around for quickly switching between saved patterns on cue.

It’d be great if you had the option to specify a CC # to trigger each saved pattern, and when that CC is received on the MIDIalf’s control channel, MIDIalf would load and play that pattern. Then you could use an external MIDI controller to trigger between different sequences on the MIDIalf. Maybe the load/save screens could have an option on it to specify a CC # for each pattern. Or even better, maybe there could be a MIDI learn function that would let you click the controller button you want to use to trigger that pattern.

I’ve been working my way through the source code, trying to get myself familiar enough with it to take a stab at implementing some of this stuff myself, but I’m new to the whole hardware programming thing, so the learning curve’s a bit steep. Haha. One thing I’m having trouble wrapping my head around is how to figure out how much memory might be available for adding stuff to the MIDIalf – the .hex file seems to be bigger than the 64kb that’s available on the atmega644p, so I guess you can’t figure it out that way… Could you give me some idea of how “close to capacity” the MIDIalf is with the present firmware?

Reply

kvitekp says:
on January 10, 2014 at 3:51 pm
I’ll be expanding list CCs supported by MidiALF in a few days, and ability to load sequences is on the list of planned extensions.

MidiALF uses about 89% of available flash memory, you can check this with this command:

/WinAVR/bin/avr-size -C –mcu=atmega644 build/midialf/midialf.elf

Reply

borfo says:
on January 10, 2014 at 7:23 pm
Thanks. There isn’t some similar command to estimate ram usage, is there?

kvitekp says:
on January 11, 2014 at 6:42 am
Same command gives RAM usage, see Data: section. It says 35% is used for static data, the rest is used for stack.

Scott says:
on March 16, 2014 at 9:59 am
is there a way to control the number of times each step repeats? I'm curious if the midialf could be used like a midi version of the Intellijel Metropolis – http://www.intellijel.com/eurorack-modules/metropolis/ (based on the Ryk M-185 sequencer)

Reply
kvitekp says:
on March 16, 2014 at 5:10 pm
Yeah, check progression settings: you can set how many times a step or a set of consecutive steps will be repeated.

Reply
Scott says:
on March 19, 2014 at 12:08 pm
Can the sequence be transposed via midi keyboard?

Reply
kvitekp says:
on March 19, 2014 at 5:17 pm
Of course!

Reply
Matt says:
on August 22, 2014 at 12:58 am
I have two of them now, great machine. One thing should be add – Selection for Global MIDI channel for notes (1-16)

Reply
kvitekp says:
on August 22, 2014 at 3:29 am
Are you saying that the channel notes are sent out should be different from global midi channel MidiALF is set to?

Reply
Matt says:
on August 22, 2014 at 4:25 am
hi, sorry I think cubase redirect all to one channel 😦 I check with midi monitor and sequencer send everything ok .

Reply
zardoz says:
on September 10, 2014 at 1:35 pm
hi,
very nice sequencer, a lot of fun playing with.
There is just two missing options for me :
An option to disable midi thru.
An option to disable the sequence transposition.

Reply
kvitekp says:
on September 10, 2014 at 1:42 pm
Makes sense, i'll consider implementing those in future firmware revisions.

Reply
zardoz says:
on September 10, 2014 at 2:00 pm
great ! thanks.

Olivier Biron says:
on October 23, 2015 at 7:26 am
The chords mode is awfully missing 😦
Is this in the air ?
Take a look at Zaquencer or Engine.
I have the Zaquencer and the chords feature is useful.
Not for everyone but it is a nice feature.

Reply
kvitekp says:
on October 23, 2015 at 7:34 am
Yeah, it may useful, however i don't think chord mode should be part of a sequencer like MidiALF. Instead, chord mode is a natural part of a polyphonic sequencer like MidiSeq: https://midisizer.com/midigal/midiseq-sequencer/

Reply
mgavette says:
on January 16, 2016 at 7:16 am
Just got an ALF and all I can say is WOW. Awesome work, Peter!

One thing I am wondering — how hard would it be to add a smoothing parameter to the CC sequences? The sharp jump works great for some things, but for others it's rather abrupt and would be better served by a transition from one step to the next. It would make the CC sequences like ultra-configurable LFOs!

Reply

kvitekp says:
on January 17, 2016 at 8:03 am
This is planned for ALF2.0, however in the current firmware i'm not sure it si possible because there is not enough CPU clocks left to update DACs at the rate required for smooth transitions.

Reply

mgavette says:
on January 17, 2016 at 12:11 pm
Meaning new firmware on the existing hardware?

kvitekp says:
on January 18, 2016 at 8:11 am
yeah, hardware is the same.

mgavette says:
on January 19, 2016 at 4:35 am
Amazing — looking forward to it!

One more question if I could — is it possible in the current firmware to trigger sequence on/off with note on/off? I noticed above that someone asked about transposition so I thought perhaps I was missing it somehow. Thank you!

kvitekp says:
on January 19, 2016 at 6:56 am
I'm not sure what you mean by "trigger sequence on/off" — do you mean start stop the sequence with on/off ? If so, no.

mgavette says:
on January 19, 2016 at 7:04 am
Right. Like upon hitting a note on the keyboard the sequence would start and when released the sequence would stop. This would be great as a sort of arpeggiator (if it transposed based on the note message) but also as an extra synth modulation source if you muted all the note steps and just used the ccs or lfos (set to sync to start as available in the current firmware) mapped to parameters on your external hardware.

Reply
julian says:
on September 8, 2016 at 1:44 am
hi, i didnt find, if there was an clock diveide function, like 1/3, 1/7. only seen the through 2 or 4. thank you

Reply

kvitekp says:

on September 8, 2016 at 5:43 am
Click division could be the following:

1; / / 64th Triplet
2; / / 32nd Triplet
3; / / 32nd
4; / / 16th Triplet
6; / / 16th
8; / / 8th Triplet
9; / / 16th Dotted
12; / / 8th
16; / / 4th Triplet
18; / / 8th Dotted
24; / / 4th
32; / / 2nd Triplet
36; / / 4th Dotted
48; / / 2nd (half)
64; / / 1 Bar Triplet
72; / / 2nd Dotted
96; / / 1 Bar (whole)
128; / / 2 Bars Triplet
144; / / 1 Bar triplet
192; / / 2 Bars (double)

Reply