

a lgpt lazy dog.

Basic Editing & Navigation

ARROWS: In screen navigation.
 A: Insert Chain/Phrase/Note.
 A,A: Insert next unused Chain/Phrase/Instrument.
 LT+(B,A): Clone.
 B+A: Cuts the current Highlighted Item .
 A+ARROWS: Updates Highlighted Item value.
 A+UP/DOWN: +/- 0×10.
 A+RIGHT/LEFT: +/- 1.
 B+ARROWS: Rapid Navigation.
 B+UP/DOWN: Page up/down in Song Screen, Next/Previous Phrase in Current Chain in Phrase Screen. Navigation +/- 0×10 in Instrument/Table Screen.
 B+LEFT/RIGHT: Next/Previous Channel in Chain/Phrase Screen.
 Navigation +/- 1 in Instrument/Table Screen. Switch between Song and Live Modes in Song Screen.
 RT+ARROWS: Navigate between the Screens.

Selections

a few ways to make a selection:
 LT+B: Starts selection mode with only the data at the cursor selected
 LT+B+B: Starts selection mode with the cursor's row selected
 LT+B+B+B: Starts selection mode with the entire screen selected
 once a selection is started you can do a few more things:
 ARROWS: will make an existing selection bigger or smaller
 B: copy selection to buffer, or
 LT+A: cut current selection
 And then:
 LT+A: paste the clipboard content at current location

Playback

B+LT/B+RT : In song screen toggle between live & song mode

song mode:

START: In song screen, starts / stop the song In phrase,chain starts / stop playing the current chain/phrase if there is a blank spot in a row lgpt doesn't search up the column for a chain to to play.
 RT+START: In phrase/chain starts playing the full song from current position

live mode:

START: Queues the selected chain The queued chain will play when the chain which is currently playing finishes. If no chain is already playing on a channel, the next chain which finishes on any channel will start the queued one. Queued phrases are shown with a blinking '>'.
 Pressing START a second time will queue the position in immediate mode ... i.e the position will start as soon as the current playing phrase is finished. If a chain is in immediate queue the sign '>' blinks faster.

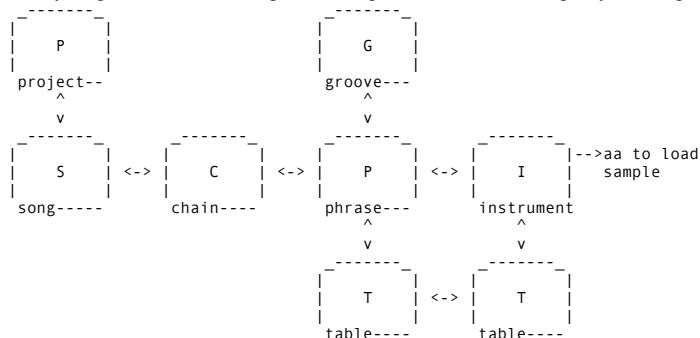
RT+START: Queues the current channel to stop as soon as the currently playing chain is finished. Pressing R+START a second time will toggle immediate queue where the channel stops as soon as the current playing phrase is finished.
 If queuing is enabled in the chain/phrase screen, the current edited chain will be queued

Muting

RT+B: Toggles mute/unmute of cursor channel
 if RT is released before B, channel stays mutes
 if B is released before RT, channel goes back to original state
 RT+A: Solo cursor channel
 if RT is released before A, channel stays solo'ed
 if A is released before RT, all channel go back to original state
 LT+RT: restore full playback on all channels
 RT+A,RT+B can be used in conjunction with selections. If a selection is present the toggle mute/solo action is done on all channels present in the selection

System Operations

Return to the song selection screen by pressing L_SHOULDER+R_SHOULDER+SELECT (Esc on windows). warning: all data is currently forgotten when resetting to the song selection screen. No pussy warning.



Commands

ARPG abcd cycle through relative pitches a, b, c, and d (starting with original pitch, then up a semitones, b semitones and so forth). The cycle loops if there's only zero's past a given post
 ARPG 3000: loops between original pitch and +3 semitones
 ARPG 4050: loops between original pitch, +4 semitones, +0 semitones, + 5 semitones
 speed of arpeggiator is constant and can not be changed
CRSH aabb with aa = pre crush drive (from 1 to 0xFF 00=no change) bb = crush setting (from 0 to 0xF 0×0= 1 bit // 0xF = 16bit)
FCUT aabb adjust the filter cutoff to bb at speed aa
 FCUT 0080 will instantly set the filter cutoff to 50%
 FCUT 1000 will close the filter entirely at speed 10
FLTR aabb lowpass filter, set absolute frequency value for cutoff aa & resonance bb
 FLTR 00FF is un-adulterated sound
FRES aabb adjust the filter resonance to bb at speed aa
 FRES 08FF will raise the resonance to screeching at speed 08
HOP --a play position will jump to the next phrase in a chain, jumping directly at position a in the phrase.
 hop is instant: instrument triggers and commands on the same row will be run.
 no effect on instruments
KILL -aa instrument will stop playing after aa ticks.
LEGA aabb performs an exponential pitch slide from previous note value to pitch bb at speed aa.
 00 is the fastest speed for aa (instant, useless)
 bb values are relative: 00-7F are up, 80-FF are down, expressed in semi-tones
 if LEGA is put on a row where a note is present and the pitch offset is 0 (e.g. C4 I3 LEGA 1000) the slide will occur automatically from previous note to the current one at the given speed.
 If an instrument is not triggered on the same row as LEGA, the command will re-trigger the previous instrument (unless the previous instrument is still playing).
 LEGA does exponential pitch change (i.e. it goes at same speed through all octaves) while PITCH is linear
LPOF aaaa Loop OFset: Shift both the loop start & loop end values aaaa digits
 LPOF 0001 adds one to both values, LFOF FFFF removes one
 reset everytime you start a new note (same as volume, pitch)
 LPOF is absolute
 you can't trigger a note with the LPOF, it has to be executed after a sample is playing
 every time you trigger a sample LPOF is set back to the instrument parameters
MDCC aabb
 Sends a MIDI "continuous control" message. aa is the control number and bb is the value. It will be sent on the MIDI channel of the currently running instrument.
MDPG ???? it sends a program change command
PAN aabb: where bb is the pan destination and aa is the speed to get there
PLOF aabb PLayer OFset virtually cuts any sample in 256 chunks. jump absolutely to chunk aa or relatively move forward/back bb chunks.
PTCH aabb
 PTCH is also time for the first two byte nibble
 PITCH is linear pitch change
RTRG aabb retrigger the sound by looping the from current play position over a certain amount of ticks.
 bb is the number of ticks used for the looping.
 aa is an additional value that allows to move the loop forward of aa ticks each time the loop has been done.
 RTRG 0001: loop one tick from current play position
 RTRG 0102: loop of two ticks but move the loop one tick every loop
 RTRG 0101: does not do anything because after looping one tick, you move forward one tick and therefore go back to the current position :)
TABL -aa triggers table aa
TMPO -aaa sets the tempo to hex value aaaa.
 TMPO 0000 is safe and doesn't effect the tempo at all.
 TMPO 003C (60bpm) is the lowest acceptable value and TMPO 0190 (400bpm) is the highest acceptable value
VOLM aabb starting from the instrument's volume setting, approach volume bb at speed aa. 00 is the lowest volume and 00 is the fastest speed (instant). to achieve sounds that grow in volume, make an instrument with volume 0 and then apply the VOLM command