DSEQ32 (aka yet another TR clone) is based on the classic TR drumsequencers

see trsequencer for more drumsequencer designs




# Features

- 32 step 8 track drum seq, each step has trig/acc/roll
- one track can be used as note track for bass sequencing
- independant length/tempo divisor for each track, all can be synced to a master track
- 8 banks of 16 patterns
- patterns are saved on a bankstick
- settings are stored on internal eeprom
- song mode
- all functions can be executed while running
- forward/backward playing direction
- 24 PPQ resolution
- midi clock master/slave (maybe also taptempo)
- sysex functionality

# Interface

- lcd: displaying (bpm pat trknr edited_par)
- 32 step buttons + stepleds, leds are dual color to display trig/acc
  - step edit behaviour: step off/trig/trig+acc
  - all functions are accessed with a stepkey + shift combination
  - pattern selection: stepkey + pattern button
  - track select/mute: stepkey + track button

- rotary encoder for bpm, also used for editing values
- Step button layout:
  - Track layer

trk button + track select: solo track track select + play button: manually trigger note

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Select | Select | Select | Select | Select | Select | Select | Select | Select | Select | Select | Select | Select | Select | Select | Select |
| Trk1 | Trk2 | Trk3 | Trk4 | Trk5 | Trk6 | Trk7 | Trk8 | Trk9 | Trk10 | Trk11 | Trk12 | Trk13 | Trk14 | Trk15 | Trk16 |

| 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Mute | Mute | Mute | Mute | Mute | Mute | Mute | Mute | Mute | Mute | Mute | Mute | Mute | Mute | Mute | Mute |
| Trk1 | Trk2 | Trk3 | Trk4 | Trk5 | Trk6 | Trk7 | Trk8 | Trk9 | Trk10 | Trk11 | Trk12 | Trk13 | Trk14 | Trk15 | Trk16 |

- Pattern layer

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Select | Select | Select | Select | Select | Select | Select | Select | Select | Select | Select | Select | Select | Select | Select | Select |
| Bank A | Bank B | Bank C | Bank D | Bank E | Bank F | Bank G | Bank 8 | | | | | | | | |
| 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 |
| Select | Select | Select | Select | Select | Select | Select | Select | Select | Select | Select | Select | Select | Select | Select | Select |
| Pat 1 | Pat 2 | Pat 3 | Pat 4 | Pat 5 | Pat 6 | Pat 7 | Pat 8 | Pat 9 | Pat 10 | Pat 11 | Pat 12 | Pat 13 | Pat 14 | Pat 15 | Pat 16 |

- Function layer

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Cut | Copy | Paste | Delete | NoteNr | Length | Divisor | Normal | Accent | Cut | Copy | Paste | Delete | Format | Midi | Fwd/Backwd |
| Track | Track | Track | Track | Track | Track | Track | Velocity | Velocity | Pattern | Pattern | Pattern | Pattern | Bankstick | Mode | Playdir |

| 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Fill | Shift | MidiCh | MidiCh | MidiCh | Offset | Notelen | Sync to | Sync Ld | | Dump cur | | Dump BS | | LCD | Reset |
| Track | Track | Drum | NoteSeq | CC | CC | NoteTrk | Track | Pattern | | Pattern | | Pattern | | Mode | Seq |

# Progress

- first prototype
- new layout design
- ledmatrix/buttonmatrix finished (see matrix page)
- frontpanel finished/wired
- first take on controll surface handler (step, track and pattern handling)
- sequencer works, timing is not stable yet
- made a temporary case
- fixed flickering leds when pushing a button
- bankstick error handling added
- default pattern is loaded when no BS is available at startup
- independant length/divider for each track implemented
- LCD optimalisations
- loading and saving of patterns works now
- backward playing direction implemented

- midi mode implemented with simple slave mode (no interpolation)
- improved bpm handling (pattern bpm is saved with pattern)
- fill function implemented
- CS interface improved (the edit functions are now accessed with stepbutton+shift)
- settings are saved in internal eeprom
- notetrack implemented, this track will be used to program basslines
- changed data structure, new pattern is 256bytes
- cs is updated for notetrack (shift + track selects notetrack)
- shift + stop resets the song position
- due to code/ram space limitations the project is ported to a pic18f4620
- the core board was modified so that the backlight can be controlled by the software
- new bankstick routines to handle the larger pattern size
- 2 new functions added: Seq Reset and Backlight mode
- CC layer added
- tracks don't go out of sync after changing divisor
- the note track now has a note length parameter
- pattern banks are implemented
- new function: shift track
- the note-on status of tracks is displayed in the track layer
- note-off handling when seq is stopped added
- the pattern bpm is now loaded when the seq is not running
- 0.7e
    - pattern banks implemented, bug fixed in bankstick routines
    - manual trigger and solo implemented
    - basic sysex support
    - songmode now works
    - note length implementation for note track
    - it is now possible to sync pattern changes to begin of pattern
    - lots of core tick optimalisations

# Software

Most of the code is written in C

the software will be based on the midibox clockbox example [uCApps]

- Data structure
- Application code
    - Bankstick code (updated)
    - Control surface code
    - Sequencer core code
    - Sequencer get/set and helper rountines
    - control surface code
- you can find DSEQ32 v0.7e here
- the doxygen generated documentation is available here

# Hardware

Components:

- CORE module CORE Module uCApps
- DOUTX3 DOUT Module uCApps
- DINX1 DIN Module uCApps
- BANKSTICK BANKSTICK Module uCApps

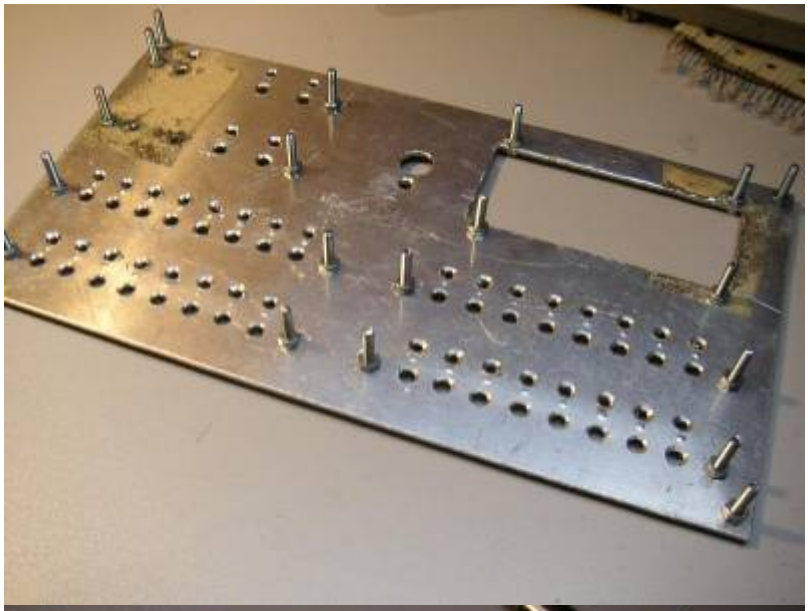
The leds/buttons are wired in a matrix configuration to reduce the number of DIN/DOUT modules needed DSEQ32_matrix

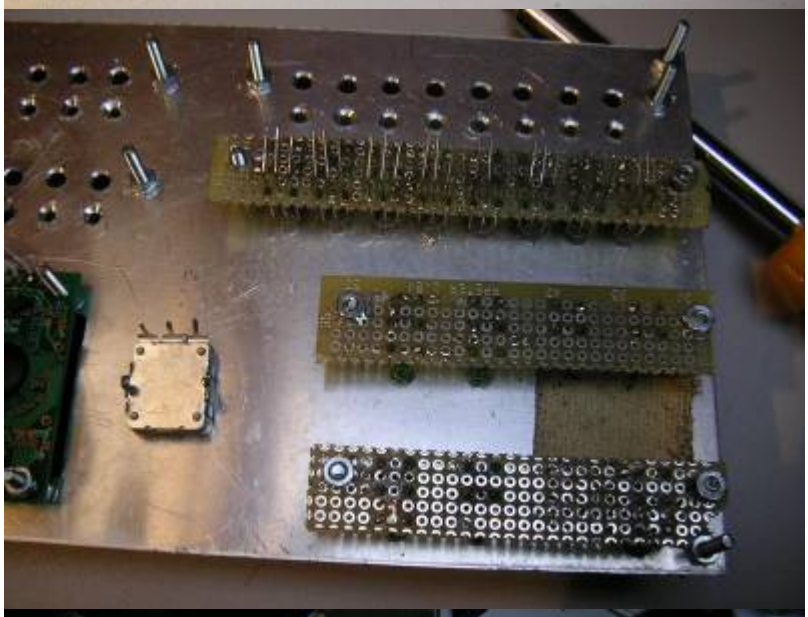
note: this description is out of date!!

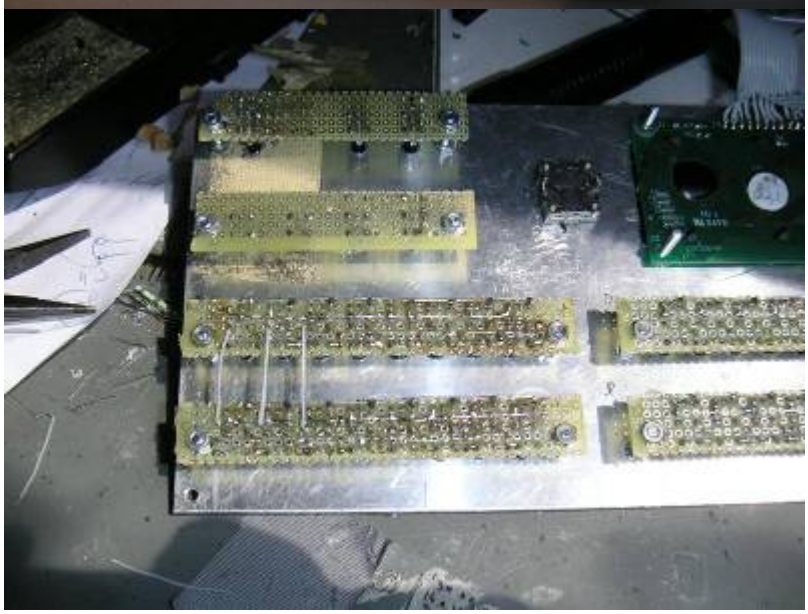
# Construction

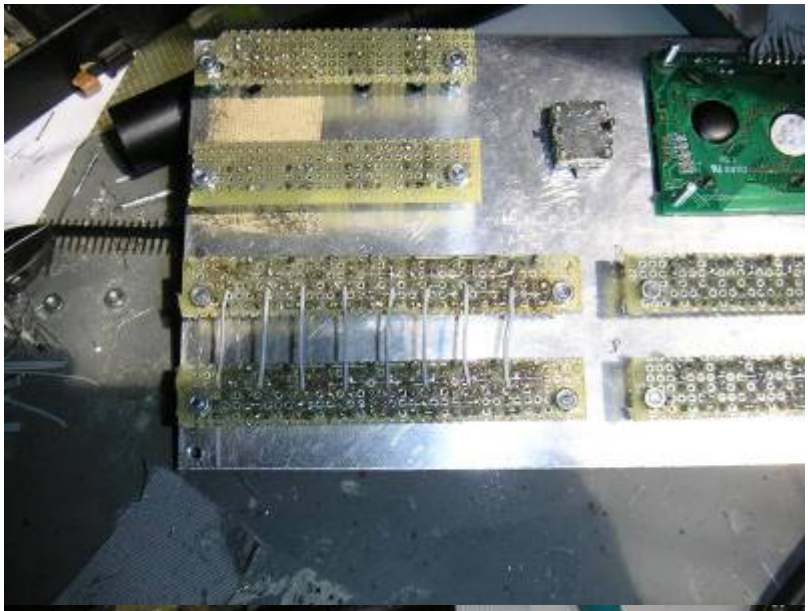
- Assembly of frontpanel

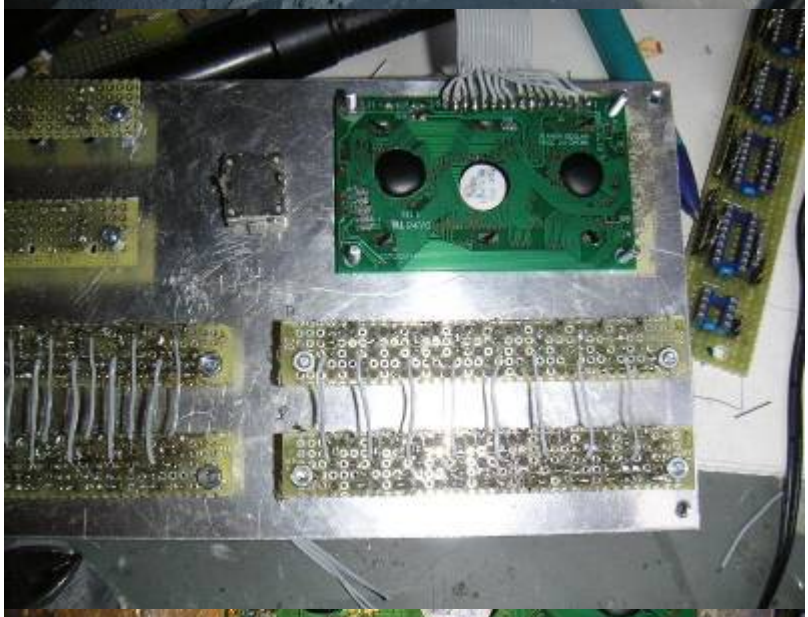


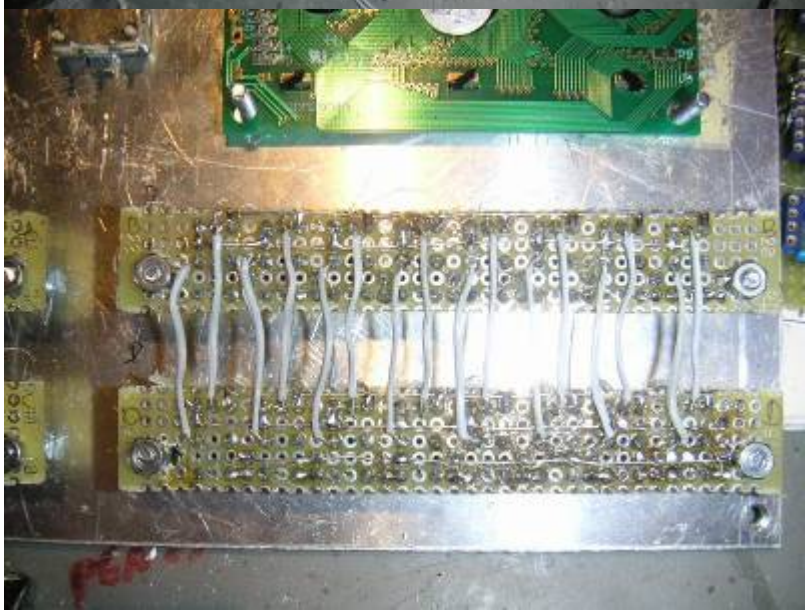




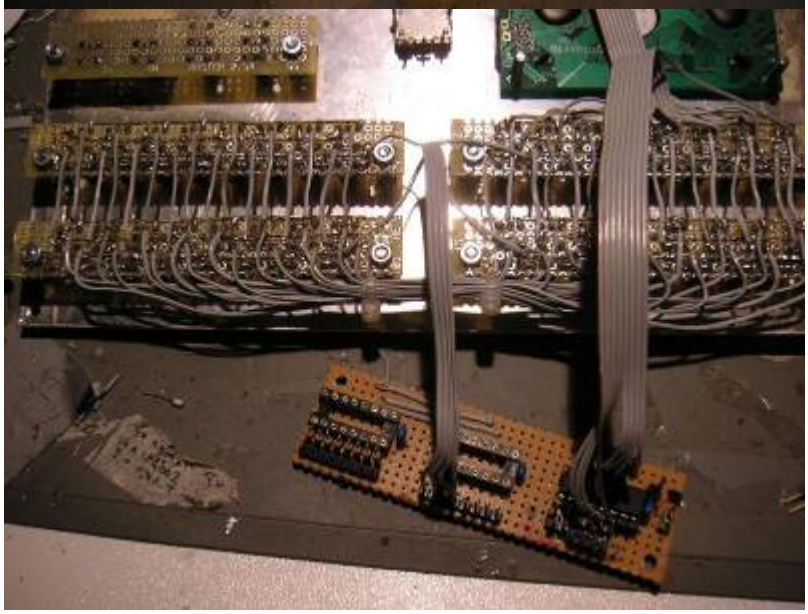




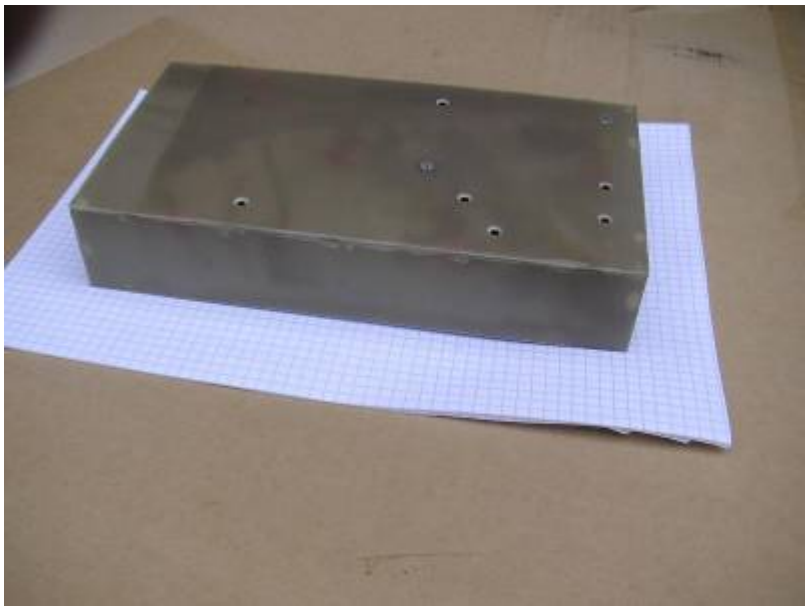


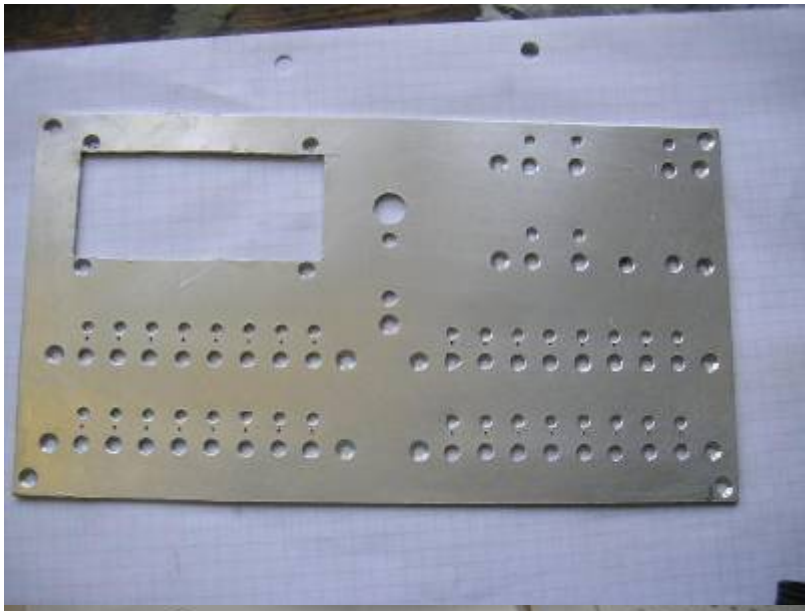




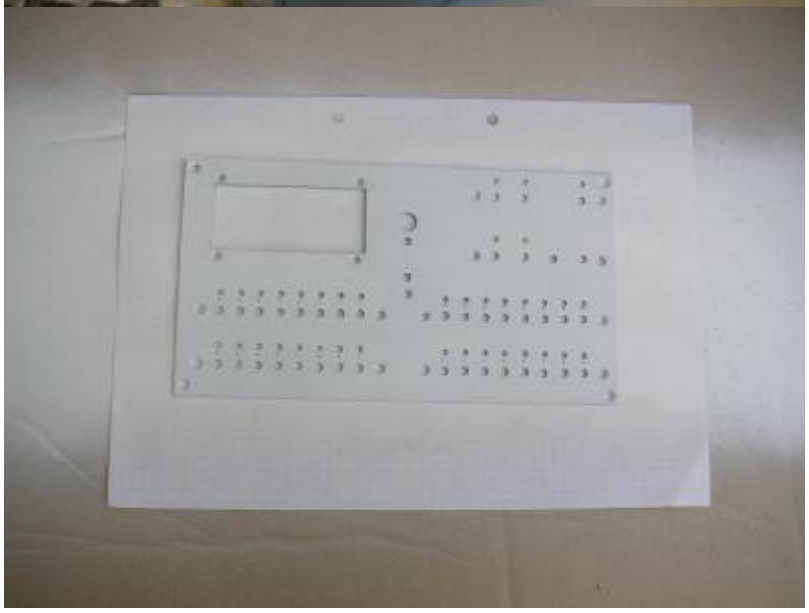


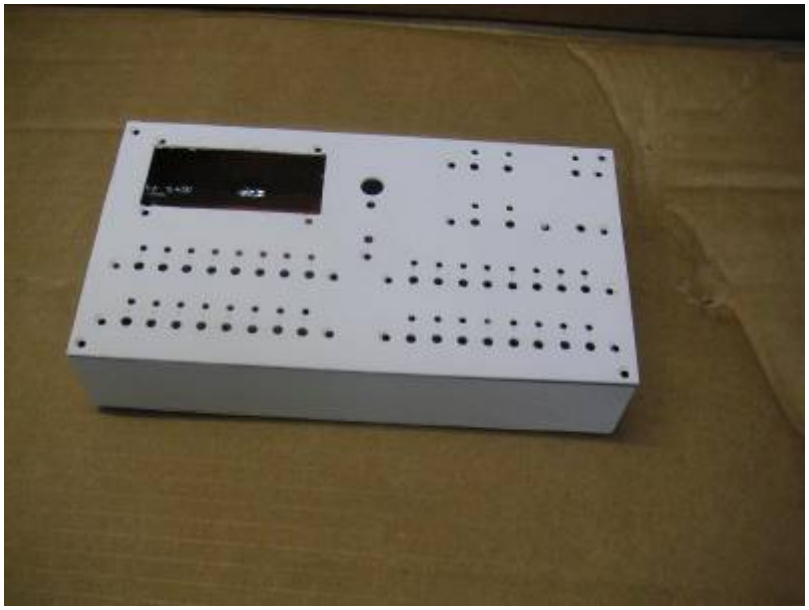
- Painting the case









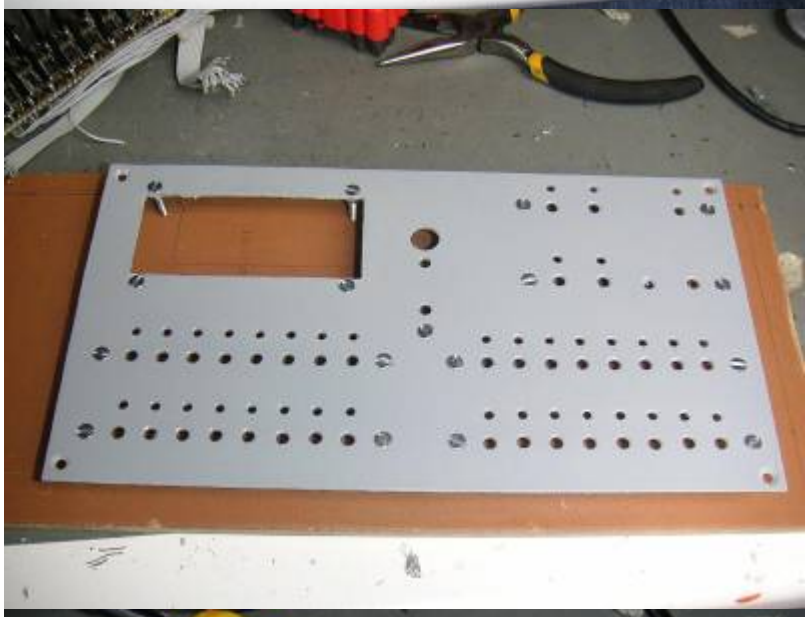


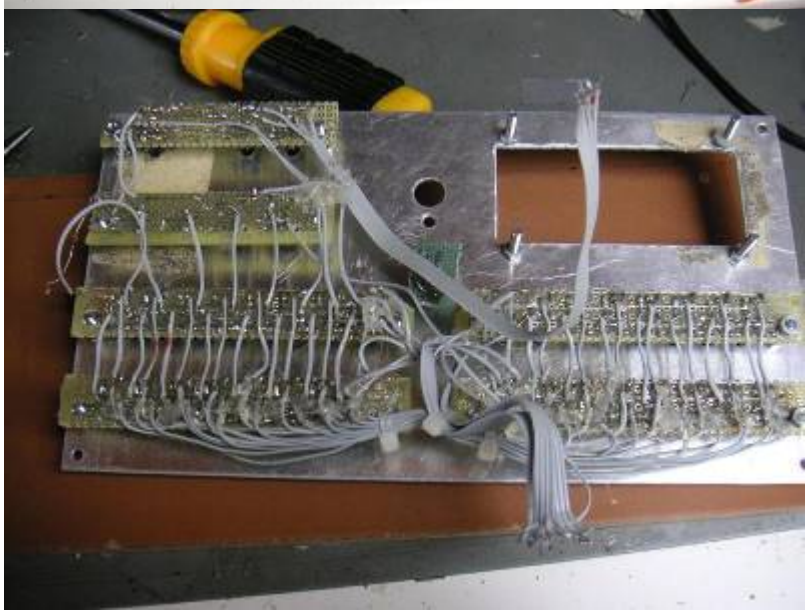
- Assembly after painting

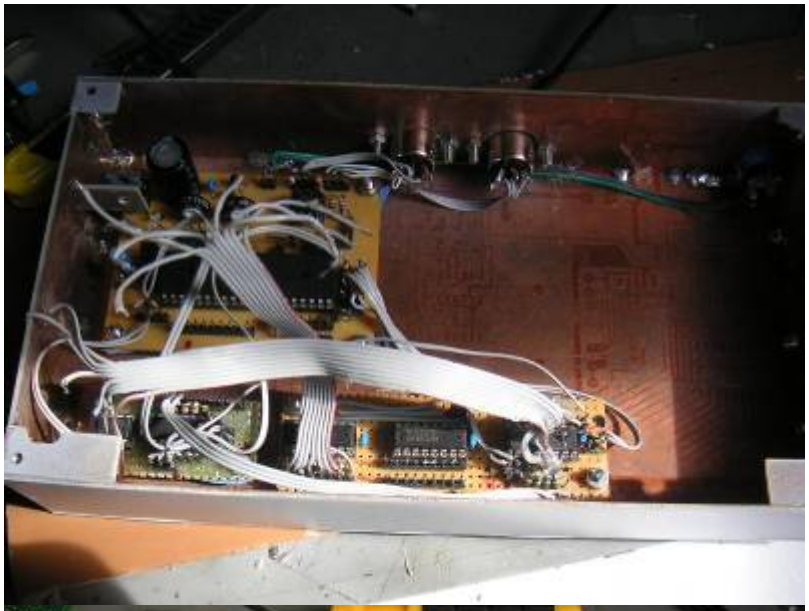














- Finished hardware

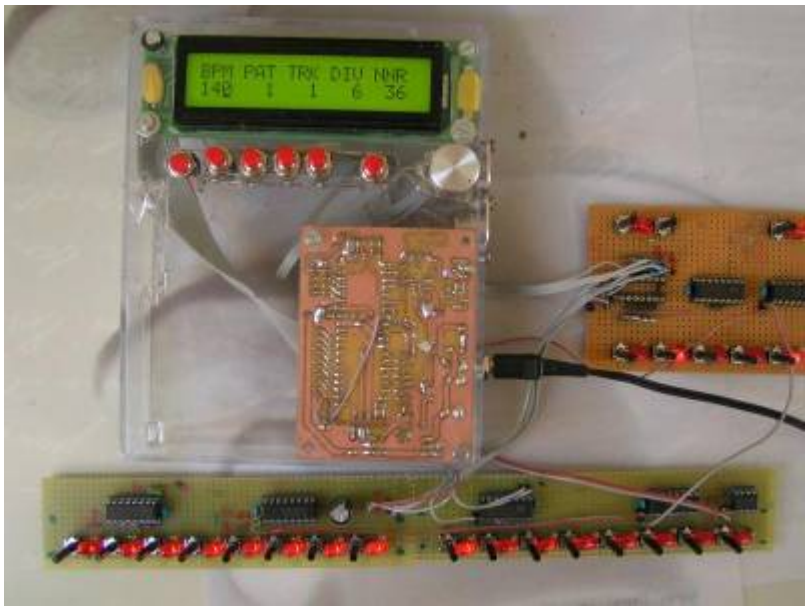




# Development

First prototype:

forum topic




used for development of data structure, bankstick code, sequencer core

New version:

- Construction of frontpanel
- Wiring of the 32 step leds/buttons
- Testing of the step buttons with the SM_example application
- Testing of the step leds with modified SM_example app (modified version includes leddriver)
- Development of user interface code
- Button and led handling finished

- The flickering led problem is solved with MIOS v 1.9c
- Encoder handling works, but there's still a small bug
- Updated LCD layout for new functions
- The leds to indicate play/save/vel/special function button work now
- Some LCD optimalisations

- Development of seq code
- Sequencer edit functions work as expected
- Sequencer runs!
- Fixed sequencer mute track bug
- Some tracks didn't send the note off messages correctly, corrected
- Bankstick error handling added
- Default pattern is loaded when no BS is available at startup
- Note length/divider is implemented
- Independant length/divider for each track

back to user_projects

more project by mess

From:
https://midibox.org/dokuwiki/ - **MIDIbox**

Permanent link:
**https://midibox.org/dokuwiki/doku.php?id=dseq32&rev=1175602479**

Last update: **2007/04/03 12:15**