LUT University
LUT School of Energy Systems
Degree Program in Electrical Engineering

Master's Thesis

AFFORDABLE CUSTOMIZABLE MIDI KEYBOARD WITH BACKLIT JANKÓ LAYOUT

Examiners: Prof. Pertti Silventoinen
D. Sc. Mikko Kuisma

Jere Peltokoski 2019

ABSTRACT

Jere Peltokoski

Affordable Customizable MIDI Keyboard with Backlit Jankó Layout

LUT University
LUT School of Energy Systems
Degree Program in Electrical Engineering
Master's Thesis, 2019
64 Pages, 26 Pictures, 1 Tables.

Examiners: Prof. Pertti Silventoinen
D. Sc. Mikko Kuisma

Keywords: alternative, Jankó, musical, keyboard, layout, MIDI, DIY, modular, Arduino

For a general-purpose electrical keyboard instrument the piano layout is cumbersome and long outdated. Alternative keyboards are superior and easier to master but have not gained popularity and are not available in general. This work describes an electronic MIDI keyboard with a hexagonal Jankó layout featuring velocity sensitivity, aftertouch and backlighting. The mechanical and electrical structure is presented along with the software overview. The keyboard can be built and repaired even with rudimentary garage tools. It is modular, expandable, customizable and offers low-cost entry. It can be retrofitted to most of the existing keyboards or run on its own with an Arduino microcontroller. The keyboard is open-source hardware and software, and manufacturer-independent. It can enter the market gradually unlike most of the other exotic instruments.

TIIVISTELMÄ

Jere Peltokoski

Affordable Customizable MIDI Keyboard with Backlit Jankó Layout

Lappeenrannan–Lahden teknillinen yliopisto LUT
LUT School of Energy Systems
Sähkötekniikan koulutusohjelma
Diplomityö, 2019
64 sivua, 26 kuvaa, 1 taulukkoa.

Tarkastajat: Prof. Pertti Silventoinen
D. Sc. Mikko Kuisma

Hakusanat: vaihtoehtoinen, Jankó, kosketin, soitin, layout, MIDI, tee-se-itse, moduuli, Arduino

Yleiskäyttöiseksi kosketinsoittimeksi piano-kosketinasettelu on kömpelö ja kauan sitten vanhentunut. Vaihtoehtoiset koskettimistot ovat parempia ja helpompia hallita, mutta ne eivät ole yleistyneet eikä niitä ole yleisesti saatavilla. Tämä työ kuvailee sähköisen MIDI-koskettimiston kuusikulmioisella Jankó-kosketinasettelulla sekä velocity-, aftertouch- ja taustavalotoiminnoilla. Mekaaninen ja sähköinen rakenne esitetään, sekä myös ohjelmiston yleiskatsaus. Koskettimiston voi rakentaa ja korjata jopa alkeellisilla autotallityökaluilla. Se on modulaarinen, laajennettavissa ja mukautettavissa, sekä tarjoaa edullisen aloituskynnyksen. Sen voi jälkiasentaa useimpiin olemassaoleviin koskettimistoihin, tai sitä voi käyttää omillaan Arduino-mikrokontrollerilla. Koskettimiston latteisto ja ohjelmisto on avointa lähdemateriaalia, ja se on riippumaton valmistajasta. Se voi astua markkinoille vähitellen, toisin kuin useimmat muut eksoottiset soittimet.

ACKNOWLEDGEMENTS

Writing of this thesis started as a hobby project that evolved into master's thesis in LUT University, Lappeenranta, and was brought to a conclusion in Turku.

At the very first I thank the examiners, Prof. Pertti Silventoinen and D. Sc. Mikko Kuisma, for the encouraging and humorously positive attitude towards science and studies in general, and for the genuine interest and partaking in wild ideas and projects such as this. Just a little push to the right direction can mean a lot.

Thanks to Ken Rushton for letting me use some of his pictures, and for the interesting and detailed blog concerning alternative keyboards. The deeper dive into these instruments gave me new insight, even though I still chose to hold on to this implementation.

Thanks to Wataru Ohkawa for sending me the Chromatone unit, a Jankó keyboard of his invention. Thank you for the rare privilege to have firsthand experience of this layout in action. I really wish you all the best and hope to see more of your keyboards out there.

Thanks to every single one of my friends with whom I have discussed about the subject, sometimes even torturing with it. Especially all of my bandmates, thank you for the music and the underlying motivation for this instrument. If I can I will do it all over again.

Thanks to my family for understanding and giving me space to finish this work. You mean more to me than I give you credit for.

Last but not least, I would like to thank my wife Merja for bearing with me, and believing in me. Thank you for showing that the concepts of true love and normal life are still incomprehensible. By being you, you never cease to amaze and amuse me. Or our unbelievable daughter. I love you! May your wisdom shine to the academic world as well!

Thanks to you, the reader, for increasing the awareness of the subject and bringing the goal of this work a little closer. I hope this work will inspire and benefit other music enthusiasts.

Jere Peltokoski
08.03.2019

CONTENTS

LIST OF SYMBOLS, ABBREVIATIONS AND TERMS

MIDI	Musical Instrument Digital Interface. Elementary serial bus for electrical communication in musical environment.
TTL	Transistor-to-Transistor Logic. Old circuit technic using 5 volts and bipolar transistors.
UART	Universal Asynchronous Receiver Transmitter. A very basic method of serial communication between two devices using start bit (low) and end bit (high) framing the data.
OSC	Open Sound Control. A high-level message-based communication protocol for networked audio and multimedia devices.
12-TET	12-Tone Equal Temperament. Tuning in which the octave is divided into 12 equally spaced tones (logarithmically on frequency scale).
GIF	Graphics Interchange Format. A long-lived image file format supporting 256 color palette, lossless compression and animation.
SPI	Serial Peripheral Interface. A fast synchronous serial interface using 4 wires (master-to-slave, slave-to-master, clock and slave select). One master and multiple slave devices.
I2C	Inter-Integrated Circuit. A serial bus using two wires (data and clock) and data frames. Multiple master and slave devices are possible with addressing.

Octave	The difference between a frequency and twice the frequency.
Tone	Certain sound having constant properties.
Pitch	Certain frequency or frequency group, often on logarithmic scale.
Note	Certain pitch used in music or a notation for it.
Harmony	Ratio between frequencies of two pitches.
Chord	Combination of notes that form certain harmonies.
Voice	A distinct tonal content of a specific physical or virtual instrument.
Tuning	Set of all available notes in an instrument.
Microtonal	Something outside of 12-TET tuning system.

1 INTRODUCTION

For an interface between a musician and a computer, a digital musical keyboard instrument is an excellent choice. For this general-purpose tool, the piano keyboard layout is imperfect. Dragging the burden of history along with it, the piano keyboard has evolved into its current form following mechanical limitations, musical choices, and conventions that do not apply anymore. There are alternative keyboard layouts that are easier and more logical, like the Jankó layout. Despite the indisputable superiority, the alternative keyboards have not gained popularity, and they are difficultly available if at all, and even more so at affordable price. However, it is possible to build one even by oneself without compromising features that exist in most of the commercial piano keyboards, yet with backlighted keys. An expandable modular structure enables low-cost entry and easy customization. The keyboard may be controlled by an Arduino microcontroller for stand-alone operation or retrofitted to an existing keyboard. This keyboard can replace most of the electrical piano keyboards, literally.

1.1 Background

This work continues from the author's previous work that studied history of keyboard instruments, tuning systems and different keyboard layouts. After being disappointed in the piano keyboard's capabilities in playing and composing along with the amount of extra effort to develop above mediocre skill level, the author sought for easier ways to play. The result was a keyboard layout that was already invented and called the Jankó layout. The author could not find anyone to build an instrument with the characteristics he desired, so he designed an instrument that everyone can build and modify to meet characteristics they desire.

1.2 Objective and research questions

The purpose of this work is not necessarily to create a new all-inclusive keyboard instrument that is the best in every aspect yet, but to set the new minimum level for keyboard instrument technology. Progress is also made when low-tech meets high-tech.

The research aims to find answers to at least these questions:
- How do the current keyboards work, and how can a new keyboard work?
- How can the new keyboard be built from the beginning, and what features will it have?
- Why is there no alternative keyboards available, and how to change it?

1.3 Scope of thesis

Even when it is possible to build the complete instrument following these guidelines, some prototyping and adjustments are needed for the final product. Also, the controller software is presented as an overview and needs to be transcribed into code and tested, which requires some additional work.

The keyboard type is limited to a keyboard with separate keys and discrete notes or pitches. Even if continuous key surface with continuous pitch offers more freedom in playing, it poses certain challenges and disadvantages.

This work covers the basic knowledge about MIDI and MIDI keyboard implementations necessary to build the keyboard, but determining exact design values and utilizing special features require reference to the original multipage standard definition.

1.4 Structure of thesis

The introduction chapter describes the starting point and motivation for the new keyboard. Chapter 2 explains how the MIDI (Musical Instrument Digital Interface) communication works on hardware and software level. Chapter 3 presents different options for sensing the key movement, including the popular implementation in current keyboards. Chapter 4 shows the mechanical implementation of the new keys, beginning from rudimentary key archetypes. The circuit board and component layout are also shown. Chapter 5 covers the electrical implementation. Wiring of the key matrix is pondered and the main components are chosen.

Software structure is outlined. Chapter 6 ponders the shortcomings with previous keyboard proposals and presents a strategy for the new keyboard to gain foothold and popularity. Chapter 7 presents results and conclusions. The features and characteristics are summarized and the status of the keyboard is reviewed. Chapter 8 is conversation about the subject.

2 MIDI PRINCIPLES

MIDI (Musical Instrument Digital Interface) is a unidirectional packet-based serial bus, that was developed in 1980s. The original physical connection suits well to a keyboard or other electrical musical instruments, even though it may get too slow for controlling complex synthesizer and sequencer setups. MIDI is also suitable for use on top of a faster connection such as USB. Advanced digital studio equipment often utilizes Ethernet for multitude of purposes. Using MIDI on software level only is also not uncommon. Other uses of MIDI includes show control (controlling lights, effects, pyrotechnics, etc.), game controllers and various electronics projects by hobbyists. Further on in this document MIDI refers to the MIDI 1.0 specification with the original physical connector unless otherwise specified.

The chosen connector for MIDI is a proprietary 5-pin DIN connector, of which 2 pins are used for communication. The traditional MIDI circuit diagram is shown in illustration 1. The connectors in the picture are the device's female connectors seen from outside of the device. 5V TTL logic is used and the signaling takes place with 5mA current when pin 5 is pulled low. The specification was updated in 2014 to include 3.3V signaling also. The signal is galvanically separated with an optocoupler. (The MIDI Association, 2018.) (Sparkfun electronics, 2015.)

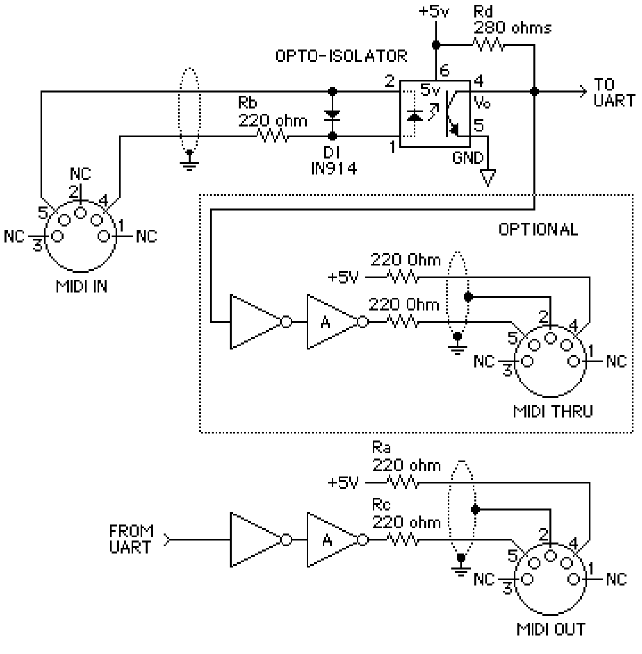

Illustration 1. Original electrical specification diagram (The MIDI Association, 2018).

Illustration 2 represents a MIDI-packet. In the image, the transmission progresses from right to left. Logic 0 corresponds to current on in the MIDI cable. Transmission speed is 31 250b/s. The MIDI-packet consists of a start bit (always 0), 8 bits of payload, and a stop bit (always 1). When idle, there is no current (Logic 1). The MIDI byte (payload) is always sent the least significant bit first. (The MIDI Association, 2018.)

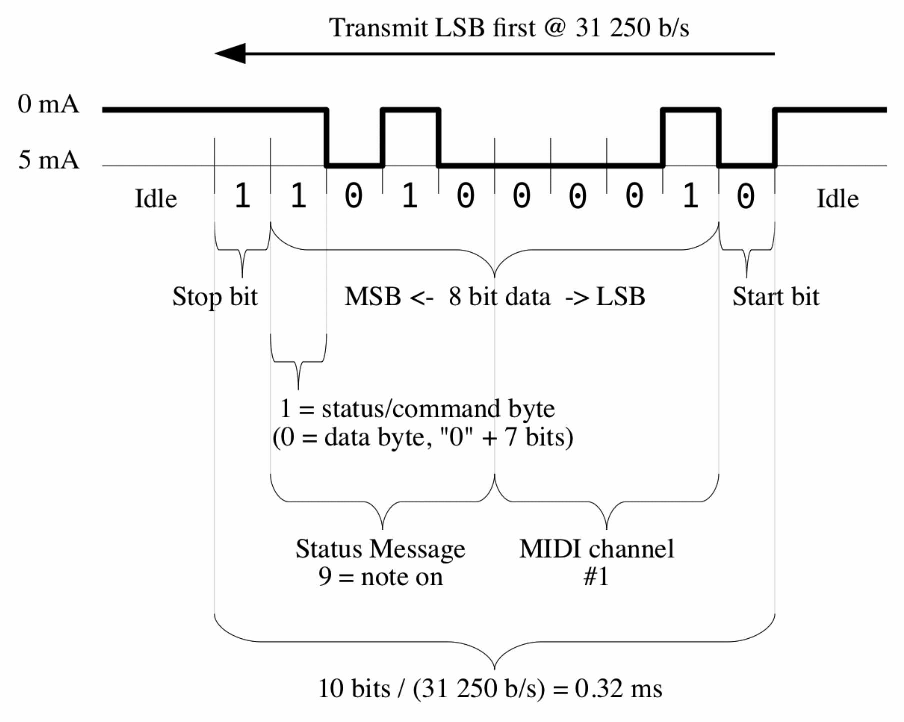

Illustration 2. A MIDI-packet. The signaling in the MIDI cable is inverse of the UART.

A MIDI byte is either a command byte (i.e., status byte) or a data byte, and it is indicated by the most significant bit (on the left in the image). If in question is a data byte, the most significant bit is zero, and 7 bits are left to be freely used for the data. If in question is a command byte, the 4 most significant bits give the specific MIDI command (i.e., status message) and the 4 least significant bits give the used MIDI channel. When sending MIDI messages, the command is sent first and then followed by the related data. Table 1 represents the MIDI messages. (The MIDI Association, 2018.) (Sparkfun electronics, 2015.)

Table 1. MIDI messages.

Status byte D7...D0	Data byte(s) D7...D0	Message	Description
1000nnnn	0kkkkkkk 0vvvvvvv	Note Off	k = key/note#, v = velocity, n = MIDI channel
1001nnnn	0kkkkkkk 0vvvvvvv	Note On	k = key/note#, v = velocity
1010nnnn	0kkkkkkk 0vvvvvvv	Polyphonic Key Pressure	k = key/note#, v = pressure value
1011nnnn	0ccccccc 0vvvvvvv	Control Change	c = controller# (1...119), v = controller value #120...127 reserved for channel mode messages
1100nnnn	0ppppppp	Program Change	p = new program#
1101nnnn	0vvvvvvv	Channel pressure (AfterTouch)	v = pressure value
1110nnnn	0lllllll 0mmmmmmm	Pitch Bend Change	l = LSB, m = MSB Center value (no pitch change) is 2000H
11110...	....	System Common Messages	System Exclusive, MIDI Time Code Quarter Frame, Song Position, Song Select, Tune Request, End Exclusive
11111...		System Real-Time Messages	Timing Clock, Start, Stop, Continue, Active Sensing (for connection keep-alive), Reset

Note on and *note off* mean that a key is pressed or released. The data bytes tell which key and how fast it was done. 128 notes are available, which results in little over 10 octaves or a little more than the human hearing range with the popular 12-TET tuning system. *Polyphonic key pressure* or polyphonic aftertouch tells how forcibly a key is held down. *Control change* tells the new position of a control such as a modulation wheel, pedal, knob or slider. Many control numbers are pre-defined, and some are reserved for channel mode messages that can be used to reset the channel or to assign voice polyphony. *Program Change* changes between patches or preprogrammed sound setups. *Channel pressure* or (monophonic) aftertouch tells how forcibly all the keys are held down. It is equivalent to polyphonic key pressure of the key that is held down with the greatest force. *Pitch Bend Change* (usually) tells the position of the pitch bend wheel, and has double precision data with only little overhead compared to the other controls. *System Common Messages* are higher level commands for intelligent synthesizers. System Exclusive can be used to send any data in MIDI data bytes. It requires an end message. A lengthy MIDI time code is sent in smaller pieces and has its own message. Song Position and Song Select are used for playback. Tune Request can be used to ask analog synthesizers to tune their oscillators. *System Real-Time Messages* are mainly used with sequencer functions. They are 1 byte long and override all other messages. (The MIDI Association, 2018.)

A typical MIDI message contains 3 bytes, so its transmission as such takes about 1ms (see illustration 2). MIDI messages can be shortened to save bandwidth. Table 2 shows an example of MIDI traffic. The left column is the long form. If the same command recurs, the command is not needed to be resent, but only the data bytes suffice (middle column). *Note on* and *note off* commands alternate often, and therefore *note off* can be substituted with *note on* whose velocity is zero (right column). (The MIDI Association, 2018.) (Sparkfun electronics, 2015.)

Table 2. Example of MIDI command shortening. All 3 columns are valid MIDI messaging.

`1001nnn 0kkkkkkk 0vvvvvvv`	`1001nnn 0kkkkkkk 0vvvvvvv`	`1001nnn 0kkkkkkk 0vvvvvvv`
`1001nnn 0kkkkkkk 0vvvvvvv`	`0kkkkkkk 0vvvvvvv`	`0kkkkkkk 0vvvvvvv`
`1001nnn 0kkkkkkk 0vvvvvvv`	`0kkkkkkk 0vvvvvvv`	`0kkkkkkk 0vvvvvvv`
`1000nnn 0kkkkkkk 0vvvvvvv`	`1000nnn 0kkkkkkk 0vvvvvvv`	`0kkkkkkk 00000000`
`1000nnn 0kkkkkkk 0vvvvvvv`	`0kkkkkkk 0vvvvvvv`	`0kkkkkkk 00000000`
`1000nnn 0kkkkkkk 0vvvvvvv`	`0kkkkkkk 0vvvvvvv`	`0kkkkkkk 00000000`
`1001nnn 0kkkkkkk 0vvvvvvv`	`1001nnn 0kkkkkkk 0vvvvvvv`	`0kkkkkkk 0vvvvvvv`
`1000nnn 0kkkkkkk 0vvvvvvv`	`1000nnn 0kkkkkkk 0vvvvvvv`	`0kkkkkkk 00000000`

Let us examine the MIDI performance in numbers. Sending of a single packet takes 0.32ms, and the throughput is 3125packets/s. It is more than enough for playing, even if every *note on/off* message requires a few packets. If 10 keys with all 10 fingers were pressed down exactly at the same time (with no other ongoing MIDI traffic), the worst-case latency would be 0.32ms+10*(2*0.32ms) = 6.72ms. In practice, it is unlikely to hit the keys this precisely, and it would only affect the last key so dramatically. For comparison, the speed of sound in dry air is 343m/s at 20°C, which means that the latency heard from a speaker is 1/343m/s ≈ 2.92ms/m. Therefore it is safe to say that the old MIDI connection handles a single keyboardist okay. Swinging the *pitch bend wheel* to its extremes and back and sending every intermediate value would take 0.32ms+(2*0.32ms)*2*(2^14-1) ≈ 21s. *Modulation wheel* and other double precision controls would take twice as much, if the double resolution is used. Therefore the controls need a properly chosen scanning speed to avoid flooding the bus. Aftertouch (*Channel Pressure*) should be also taken into account, especially if the polyphonic aftertouch (*Polyphonic Key Pressure*) is used.

To support novelty instruments, an extension has been added into the MIDI, called *MIDI Polyphonic Expression*, which enables every key press to be assigned to its own channel (The MIDI Association, 2018). This way the pressed keys can independently produce messages that would otherwise affect all keys, such as the pitch bend. The use of this extension is likely to need a faster connection and thoroughly planned hardware implementation.

Several (transmitting) MIDI devices can be daisy-chained into the same bus with a multiplexer. Some keyboards have in/out ports and internal multiplexers, but there are also several external multiplexers or suitable microcontrollers commercially available. Being of newer technology than the original MIDI, they often work with very small latency. Since MIDI messages are simple and do not contain any device-specific information (except for *System Exclusive* messages), it is easy to join several MIDI devices into one expandable modular instrument. It is also possible to use MIDI-USB converters for the modules, but it may require a custom converter or some rerouting in software, because MIDI ports are often separated at the USB side and show as different MIDI buses or devices, even on a same multi-port converter.

The original MIDI with the DIN connector is a safe fallback option and is therefore advisable to implement, even if it is not the primary communication protocol. And in some cases the more sophisticated solutions are an overkill. It is simple and requires very little in hardware and software. Almost all existing MIDI keyboard controller hardware will work as an expansion module, and custom ones can be hand-made with a simple microcontroller that has a UART capability. The USB - with MIDI or other protocol on top of it - requires a little more intelligent protocol software and a little more complex hardware in all connected modules, although USB MIDI class compliant devices work without any additional proprietary drivers. The different USB versions may pose a challenge also. The USB is not galvanically isolated by nature like the original MIDI, which should be taken into account in the musical environment. Another advanced option would be using Ethernet and TCP/IP network, but it requires even more capable software and hardware, in addition to setting up of the network. Wireless connections inside a modular instrument may not be beneficial. There is also an extreme option of using Open Sound Control. OSC messages resemble more symbolic programing language rather than MIDI messages, it is very scalable and has capability- and documentation inquiry system. Among other things, it can transport audio as well as MIDI, and it can even replace MIDI altogether. The drawbacks are that some features need to be reinvented at all ends because of the loose guidelines, and there is no obvious performance benefits over MIDI in its application area, assuming that the rest of the implementation is done properly.

At the time of writing, an updated MIDI 2.0 specification is just about to be released. It will offer many improvements along with downwards compatibility.

3 KEY SENSING PRINCIPLES

3.1 Typical MIDI piano keyboard

Illustration 3 shows a typical key implementation of a MIDI keyboard with a piano layout. The key is a lever pivoted at the other end and restricted in movement with a cushioned sliding guide at the other. The spring near the pivot returns the key back up and holds the key in place. The electronic components are two switches near the center and a force sensitive strip at the front edge. The switches are rubber dome switches that differ in height. When the key lever is pressed, the two switches close at different points of time, and the time difference is interpreted as the velocity. The pressed key is received by a dampener foam strip, and when enough of additional force is applied, it is relayed to the force-sensitive resistor (FSR) strip underneath the foam. The dropping resistance is interpreted as the (monophonic) aftertouch. Some early keyboards may have utilized a form of single pole double throw switch and a discharging capacitor to convert the velocity into voltage level, and an elastic tube together with an air pressure sensor for the aftertouch.

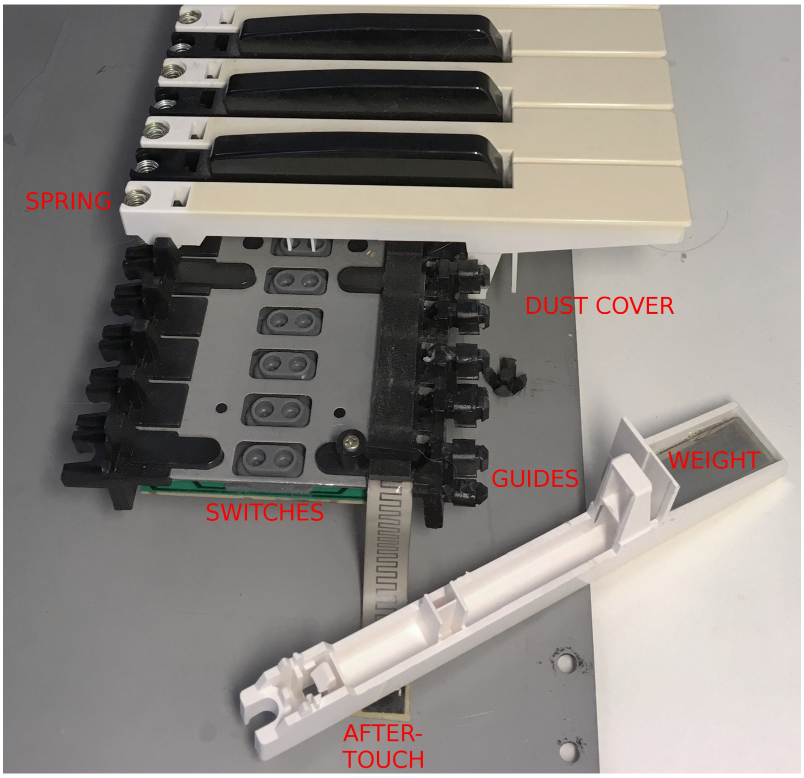

Illustration 3. A typical key implementation of a MIDI keyboard.

3.2 Lever arms with Jankó keyboard layout

Previous study has shown that the Jankó layout is more suitable to a multi-purpose keyboard instrument than the piano layout (Peltokoski, 2014). The piano layout and the related notation make barely sense in equal temperaments, that is to say, tunings with evenly spaced notes. The Jankó keyboard layout is shown in illustration 4. Here the keys or notes are numbered from low to high for clarity. The pitch rises when going from left to right. Two adjacent rows correspond to the piano layout but with no arbitrary gaps or missing black keys. The 2 rows are multiplied by 3 to make every note have 3 instances on the depth axis.

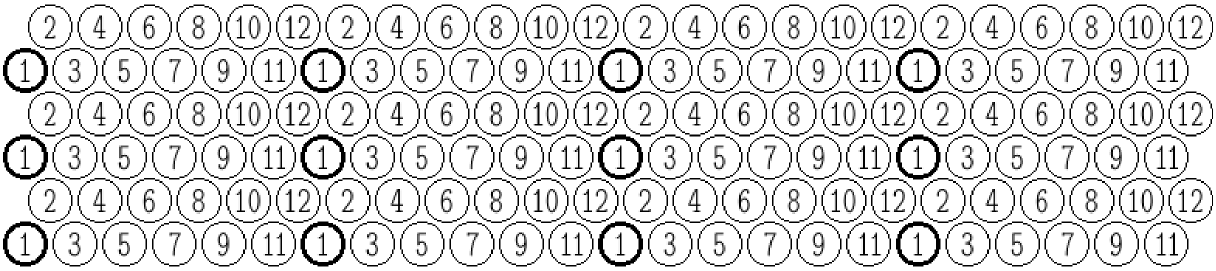

Illustration 4. Jankó keyboard layout. The pitch rises to the right. First note in an octave is marked with thick borderlines.

Illustration 5 shows a rudimentary schematic for a lever implementation of a Jankó keyboard with ascending (upper picture) or flat keybed (lower picture). There are a few guidelines to follow in the designing. To make the rows behave similarly, the lever arms should be as long as possible, and ideally the pivot point should reside vertically in the middle of the key travel. Here the maximum sinkage of the front row is 1cm, 0.5cm for the back row, and horizontally the pivot point is at twice the length of the keybed. The sensors (black area in the picture) should be located on the rotation radius of the lever to avoid any lateral or sideways movement, and also far from the pivot to have better accuracy for the key travel sensing.

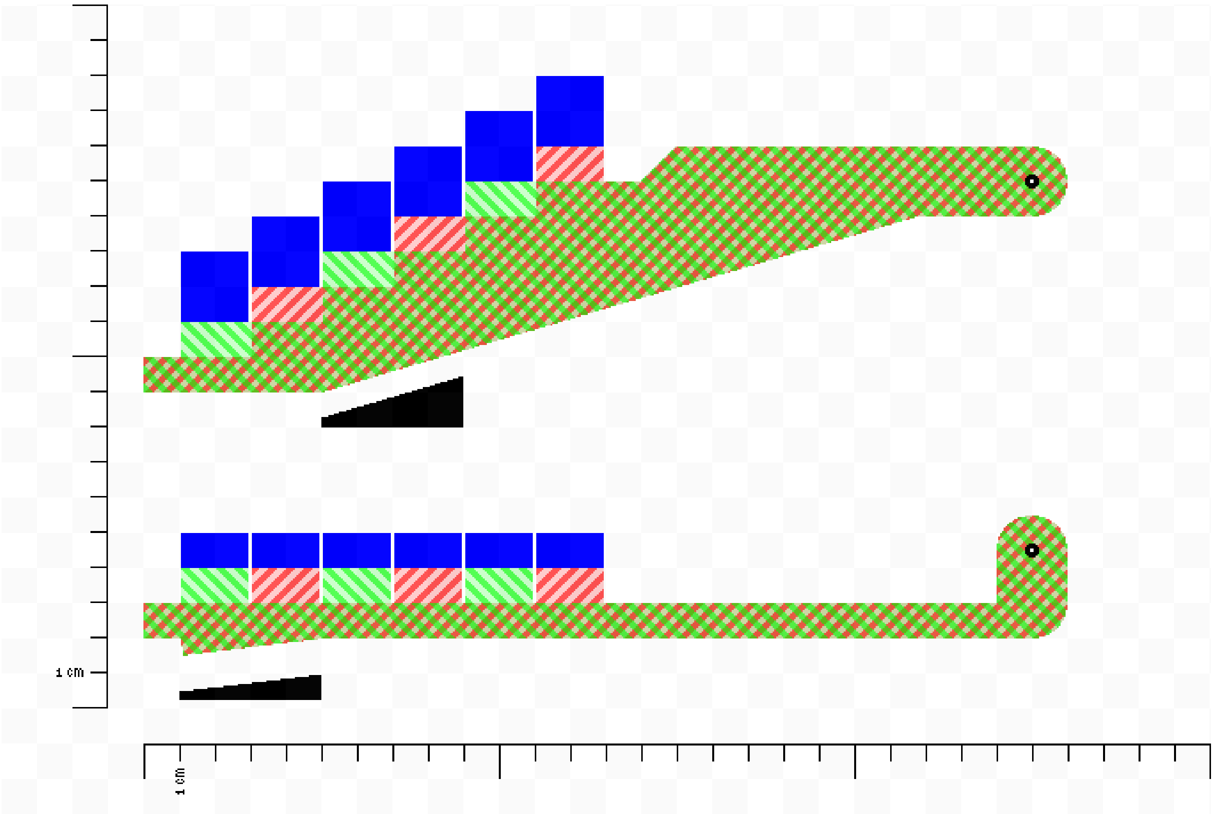

Illustration 5. A lever implementation for Jankó keyboard with ascending (upper picture) and flat keybed (lower picture). Alternating rows are marked with red and green, key tops with blue, and sensor area with black.

Long levers take up a lot of space and they weigh also. On the other hand, some moment of inertia is desirable for the feel of a semi-weighted keyboard. The levers need a proper support structure, which also adds to weight and complexity. It could be for example a grid of vertical rods or a comb at the front and the middle. The support is crucial for the rigid feel, and therefore the levers in the schematics may well need more vertical dimension in certain parts. The high pivot point poses demands for the housing and placing of the electronics. It can be moved down if the keys are allowed some back and forth movement. Backlighting of the keys is somewhat challenging. Transparent plastic like polycarbonate may be used to have sufficiently even lighting, but metal would be more rigid. The keys could be longer especially in the ascending version, but it would necessitate the levers to be longer too. A more sophisticated fine-mechanical structure with independent rows could well replace a mechanical piano keyboard, but it is easily out of the conditions set in this work. The lever implementation is possible to manufacture by hand, but involves a great amount of work.

3.3 Rubber dome switch

These switches are elastic domes usually made of silicone. The underside is made partly conductive by adding graphite, and it connects contacts printed straight on the circuit board. Illustration 6 shows a typical rubber dome. Sometimes membrane switches are used instead of the circuit board and the domes are not conductive. For keeping the domes in place they usually have small pegs that enter holes in the circuit board. The shape of the dome helps to return the switch back up, and often there is no need for an additional spring for the key. The same silicone cast can serve as the whole key in addition to the switch. The shape also provides a hysteresis, meaning that the switch tends to throw itself either up or down and not stay in between. This leads to better switching transients, and also enables some tactile feedback. Here it is possible to use two separate switches, or they can be combined side by side like in illustration **3**, or they can be combined coaxially like in illustration **7.**

Illustration 6. A typical rubber dome. The black area is conductive. Contacts on the circuit board are not shown.

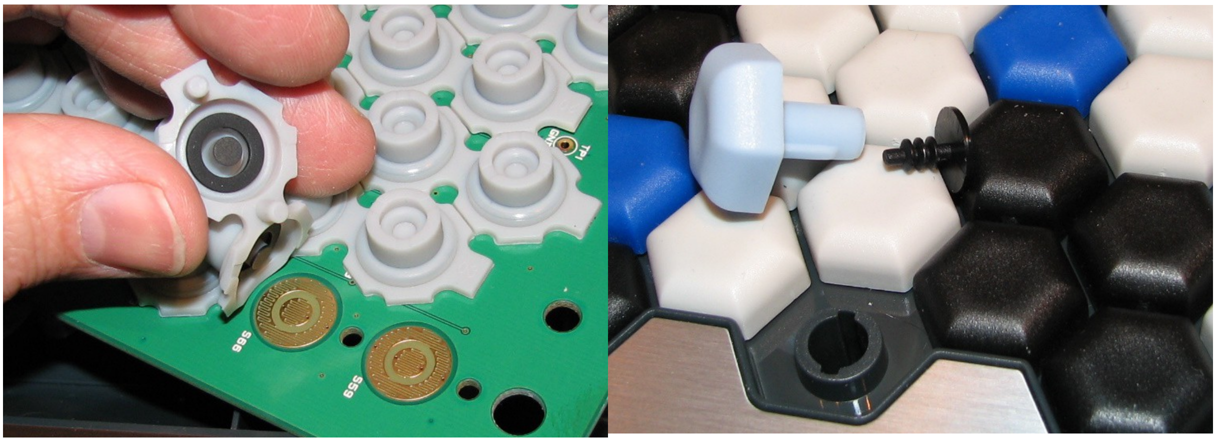

Illustration 7. Key structure of C-Thru Music Axis keyboards with coaxial rubber dome double switches (MusicScienceGuy, 2009).

There are some drawbacks to be taken into account. For the backlighting the domes can be made of transparent silicone, but the conductive parts will most likely need to be opaque. The aftertouch is difficult to implement underneath the domes. All-silicone keys with embedded switches have too poor feel for high-end keyboards. Rubber dome switches are cheap to manufacture industrially but somewhat challenging by hand. Conductive rubber is expensive and scarcely available. There has been progress in the hobbyist scene involving the use of 3D-printed molds and doping of silicone with graphite or carbon fibers.

3.4 Membrane switch

A membrane switch consists of two plastic membranes or films that have thin conductors attached on facing sides, and a third membrane with a hole at the switch region that keeps the conductors apart until force is applied. The conductors are usually printed with conducting rubber or other slightly elastic conductor material, and there may be multiple switches on the same membrane. The most familiar use of membrane switches is low-to-medium-priced computer keyboards and laptop keyboards, where they are used together with mechanical keys. They are well suited for example for toys, keytars and rollable keyboards. They can also be transparent for light effects. The drawbacks are that velocity and aftertouch are difficult to implement, and the tactile feedback is poor without additional mechanics. The durability may become an issue too. Membrane switches are difficult to make by hand, but they are very cheap using proper industrial equipment.

One way to provide velocity sensing using membrane switches is to use two switches on top of each other. They could be separated with a soft foam layer that has holes at the switch locations. A force-sensitive membrane or strip could be added underneath for aftertouch as well. The distance between the switches, that is, the foam layer thickness, is limited by the membranes' mechanical durability and affects the velocity reading accuracy. Thinking of a keyboard key, it may be reasonable to omit the membrane switches completely and utilize the force sensor only.

3.5 Simple wire switch

A very simple way to make a switch is to solder two elastic wires to a circuit board like in illustration 8. The two wires connect when an edge of a key moves downwards and pushes the protruding bend of the rear wire forwards. The two switches for the velocity can be of different height, or the key may be shaped suitably. The switch takes very little horizontal space on the circuit board.

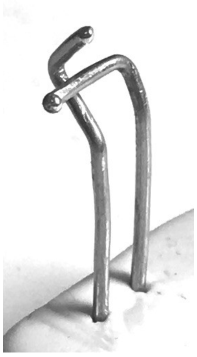

Illustration 8. A draft of a wire switch.

Although not so different from commercially available switches, the open design is prone to dirt and oxidization. This should be taken into account when choosing the material, along with its elasticity and solderability. Parallel capacitors and suitable spray-on coatings may improve the switching performance. Durability may become an issue especially at the soldering points. This switch is very cheap and easy to make and repair both industrially and by hand.

3.6 Slider switch

One solution would be using vertical strips or rails of conductors on sides of the moving key, and brushes or wipers on the frame. The rails and brushes could swap places, and they could also locate on the inner wall of a hollow key. This kind of sliders would also enable the backlight to be installed into the moving part by providing the contacts. The (polyphonic) aftertouch could also be implemented with short rails made of resistive material at the other end of the key travel. Illustration 9 shows a rudimentary schematic of the sliders. The slider implementation would move all electronics out of the way and save space for the mechanics.

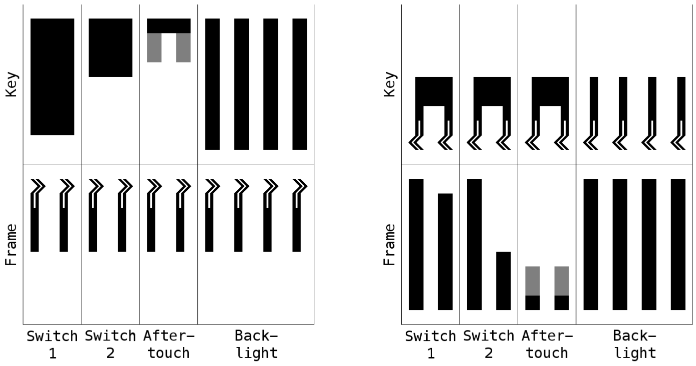

Illustration 9. A draft of slider switch implementation. Rails on the key (left) or brushes on the key (right).

The sliders require somewhat accurate placing of the conductors and the mechanics in the vertical direction. Especially the resistor strip for the aftertouch is sensitive about the placing and the resistance of the material. The sliders are prone to dirt and oxidization and may crackle a little bit, which may especially prevent the use of integrated LED chips that include control circuitry and are daisy-chained in series. The slider keys are laborious to manufacture by hand and also industrially in the beginning.

3.7 Force-sensitive resistor

Illustration 10 shows a typical force-sensitive resistor, here used for the aftertouch. The bottom layer is a film with printed conductors just like in a membrane switch. The important part is the plastic sheet on top of it. In it, conducting particles are suspended in a resistive material, and applied force brings the particles closer together to touch or close enough for quantum tunneling to occur. The adhesive tape on the sides also holds the sheet of variable resistance apart from the underlying conductors when non-operational. The top layer is plastic foam for protection and distributing the pressure more evenly. By printing a resistive track and conductors suitably on the bottom layer, it is possible to make a force-sensitive potentiometer that senses the force and also its location in case of a single touch point. Using only the force-sensitive resistors even for velocity necessitates some additional scanning electronics. They may also not eliminate the need for mechanical keys due to the lack of tactile feedback. Force-sensitive resistors are practically impossible to make by hand, but they are relatively easily available and reasonably priced. Using separate smaller sensors in every key would be very expensive, but they are cheap if they are printed on the same sheet industrially.

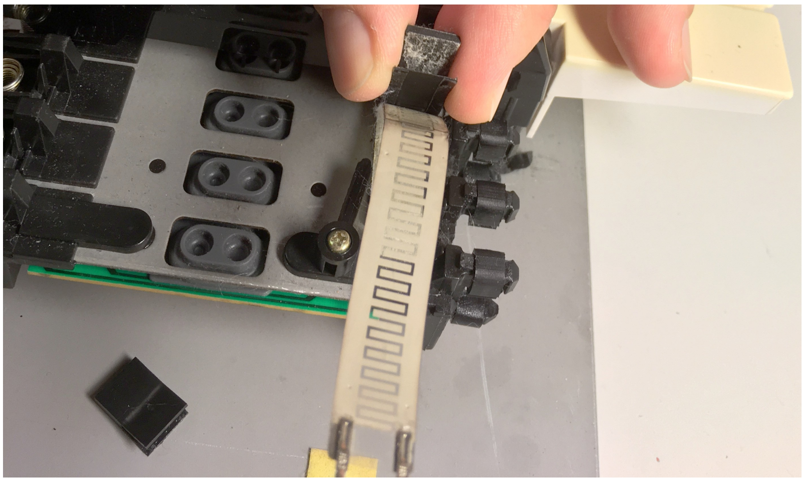

Illustration 10. A typical force-sensitive resistor strip. When pressed, the middle layer touches the conductors on the lower level, and its resistance drops gradually from megaohm range to practically zero.

3.8 Alternative key implementations

The only limit for exotic keyboard implementations is imagination. The ones covered previously represent the more traditional approach. The more refined switches may offer undeniable benefits, but very often they become expensive in numbers. The acoustic piano usually has 88 keys, and the more compact electronic keyboards usually have 61 keys. The Jankó layout triples the number of keys that have two switches each. Thus the piano width requires 88*3*2=528 switches, and the short keyboard 61*3*2=366 switches. Another factor is that the more sophisticated keys need more sophisticated electronics to scan them, which complicates retrofitting to existing keyboard electronics and may necessitate special hardware and software. The power consumption is also a matter worth considering, especially if the power source is shared with other electronics, or the power is drawn from a powered bus like USB. There are a few proposals for exotic implementations below.

Optical infra-red photo interrupter sensors are very robust, durable and easily available. They are an attractive choice for some keyboardists, but they cost as much as a commercial MIDI controller with a piano keyboard. They take up some space physically and they need power and possibly additional components. They can be turned on and off to reduce power consumption and heat generation, but it will cause a small delay in every switch, which may limit the scanning speed of the whole keyboard.

Accelerometers cost roughly the same as the photo interrupters. They are fairly accurate and fast, and they often have some processing in them. They consume only little power. A drawback is that this many devices need processing power for the data and a fast bus or other means to communicate. The electrical connection between a moving key and the circuit board is also a challenge. Furthermore, the environment of a musical instrument includes a lot of vibration, shocks and changing orientations, so the accelerometers may not be the obvious choice.

3.9 About continuous touch surface and microtonality

A natural upgrade and the ultimate alternative for the mechanical keybed is a keyless continuous touch surface. The touch surface is not chosen for the instrument presented in this work because it is significantly more complex to implement both in hardware and software. However, it is worth given a thought.

Let us define that the x-axis denotes the direction from left to-right, y-axis from front to back, and z-axis from top to bottom, or the pressure. The x-axis corresponds to pitch just like in the Jankó or piano layouts, but smoothly and including the pitch bend control. The z-axis or the pressure corresponds to the aftertouch, with the velocity being the initial pressure. The y-axis corresponds to the modulation control. Each touch point or finger may have their own x,y,z coordinates, which is supported by the MIDI polyphonic expression extension.

The challenge is to create a surface that can be read fast and accurately with modest computing power and price. Some very imaginative proposals exist, and there is plenty of room for more.

A disadvantage with the continuous surface is that it requires great accuracy and practice from the player. If the octave width is the same as in the Jankó layout, the note distances are halved. If the octave width is stretched, the reach or span of a single hand will suffer. To facilitate playing in tune, the touch points may be quantized, but it gives rise to new problems. Like how to perform a pitch slide with a little misplaced triad; should the endpoint be quantized for mistakes as well, or is it interpreted as an intentional effect? The physical keys with their quantized notes always allow some correction in the finger placement. Similarly, the fingers are positioned unequally on the y-axis. Therefore the usable range of the y-axis must be less than the total length. The longer the y-axis of the surface the harder it is to hit correctly on the x-axis.

A continuous surface must always keep track on the touch points and distinguish between them. This requires also some computing. An alternative solution is to use sensors at the fingertips. This way only ten sensors are needed to form the instrument, and they can work almost independently. The touch surface itself may be kept simple, and the fingertip sensors may interpret their location from small radio- or infra-red beacons located under the surface for example.

The continuous touch surface can be played relatively easily by both piano and Jankó keyboardists alike.

Playing microtonal music in tune, that is, hitting discrete notes outside the 12-TET tuning can be very challenging with the continuous surface whereas the hexagonal (Jankó) keyboard is somewhat usable for some microtonal experimenting. With for example the 31-TET tuning, with 6 rows of keys and 31 keys per octave, an octave span would still be achievable with one hand. The 31-TET tuning is the next closest equally tempered tuning system to 12-TET that performs at least as well in approximating the simplest fundamental harmonies available in

12-TET (1:2, 2:3, 3:4=4:3=2:3, 4:5, 5:6, 9:8) and also supplements the series (6:7, 7:8). This also means that 31-TET can be used to play 12-TET music with slight detune.

The Jankó keyboard does not allow smooth pitch glides like continuous pitch instruments, but the shortcoming may be helped with software. Most MIDI capable synthesizers offer at least monophonic glide mode, in which a new key press triggers a glide from the previous or currently held down note to the newer note at predetermined speed. There are also products that utilize accelerometers and gyroscopes as wearable sensors like rings and wristbands that may be used to control glide or pitch bend. Nevertheless, a Jankó keyboard allows the player to easily perform a glissando that goes through every intermediate note, which is practically impossible on a piano keyboard. Basing on this characteristic, it is possible to interpret a glide from the MIDI data to some extent.

4 NEW MECHANICAL IMPLEMENTATION

4.1 Key width

The most versatile keyboard is a hexagonal grid. Here the chosen key width, or the distance of parallel sides of a key, is 22 mm. Most keyboards with piano layout have approximately 23...24 mm wide white keys. Table 3 lists some key widths for comparison.

Table 3. Some key widths.

Keyboard	Width	Source
Apple MacBook laptop computers	19.0 mm	measured
Yamaha keyboard instruments	21 mm, 22 mm	Yamaha Corporation, 2018
CME MIDI controllers	23.3 mm	measured
Original Jankó	20.2 mm	Daskin, Inc., 2018
Daskin Jankó	21.6 mm	Daskin, Inc., 2018
Chromatic Music Lab Jankó	23 mm, 20 mm	measured, Chromatic Music Lab, 2018
C-Thru Music hexagonal keyboard	24.7 mm	measured

4.2 Key primitives

Illustration 11 shows the starting point for designing a simple key. The key tops can be located outside the frame structure, inside the frame or close to the center. The keys can also be stacked.

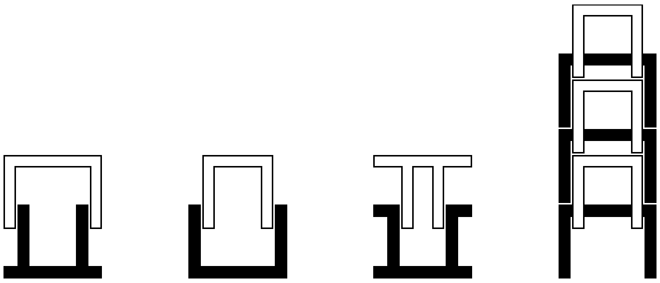

Illustration 11. Key archetypes. From left to right: The key top outside the frame, inside the frame, close to the center, and stacked. Key caps are white, the frame is black.

The first structure with outer key cap offers protection from dust and hides the frame from sight. If the key structure is tall, like it is likely to be here, it may not be rigid, because the mechanical support is only available at the bottom. The frame needs to be attached firmly to the underlying layer, which plays a major role in the support. The electronics may not have enough space at the edges, and some additional inner structure may be required.

In the second structure the key cap recesses into the frame. The frame is very rigid, especially it if is honeycomb-shaped, but because the gap between the keys needs to be small, the frame may be difficult to attach to the underlying layer. Again, the electronics of adjacent keys may not have enough space.

The third structure is a compromise of the previous two, moving the mechanics closer to the center, which provides more space between adjacent keys to facilitate placing of their mechanics and electronics. This structure may pose more strict tolerances for the mechanics and the backlighting.

The last structure shows the idea of stacking the key caps into layers to gain more space. A disadvantage in a multi-layer key structure is that it must be assembled a whole layer at a time, which means that a single key cannot be assembled or replaced without involving other keys as well. Another problem is that it may be flimsy, and the support structures or bars of the key cap have only little space at the edges. It is undesirable for the support bars of adjacent keys to share the same hole in the frame. Considering a hexagonal keyboard, this means that the bars should preferably be placed away from the corners. Three bars may be used at the edge of every second side of the hexagon if all keys are oriented the same way, as depicted in Illustration 12.

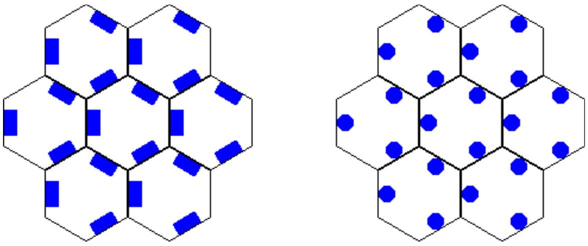

Illustration 12. Support structure (blue) of a hexagonal stacked key at the edges with rectangular or circular bars.

4.3 The new key structure

The mechanical implementation of the new key is shown in illustrations 13-15. Illustrations 16 and 17 show the electrical layout, which is explained in more detail later. A practical size for a single (Jankó) module is 8*6=48 keys, but the illustrations only show half for simplicity. The key width is approximately 2.2 cm, and the key dip is 5 mm plus 3mm maximum for the aftertouch. Most structures are designed to utilize 5 mm thick plastic sheets. Injection molding and 3D-printing may be utilized if available. The design is explained from the bottom up.

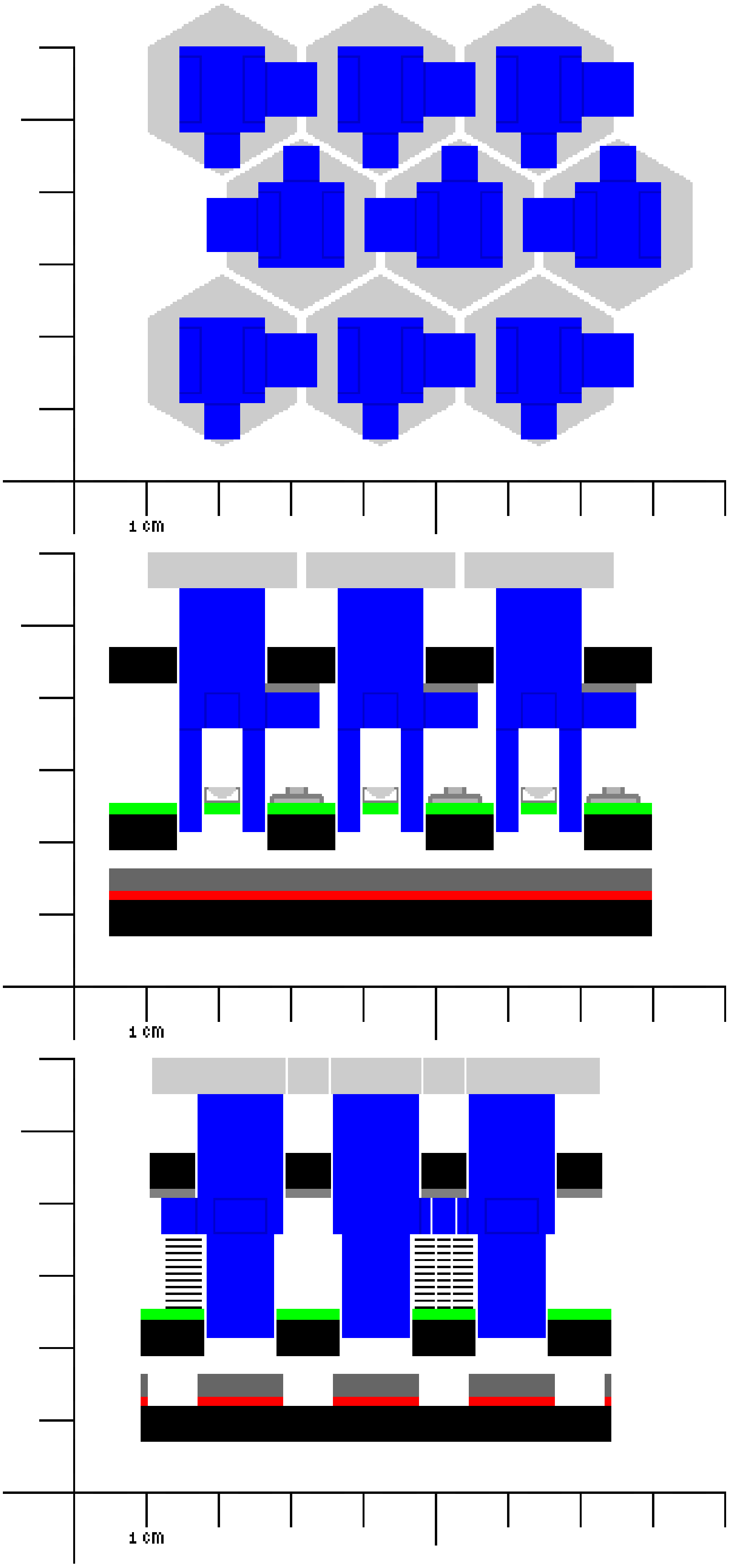

Illustration 13. The key structure from top (top picture), front (middle picture), and right (bottom picture). Depicted is a section from the front right edge.

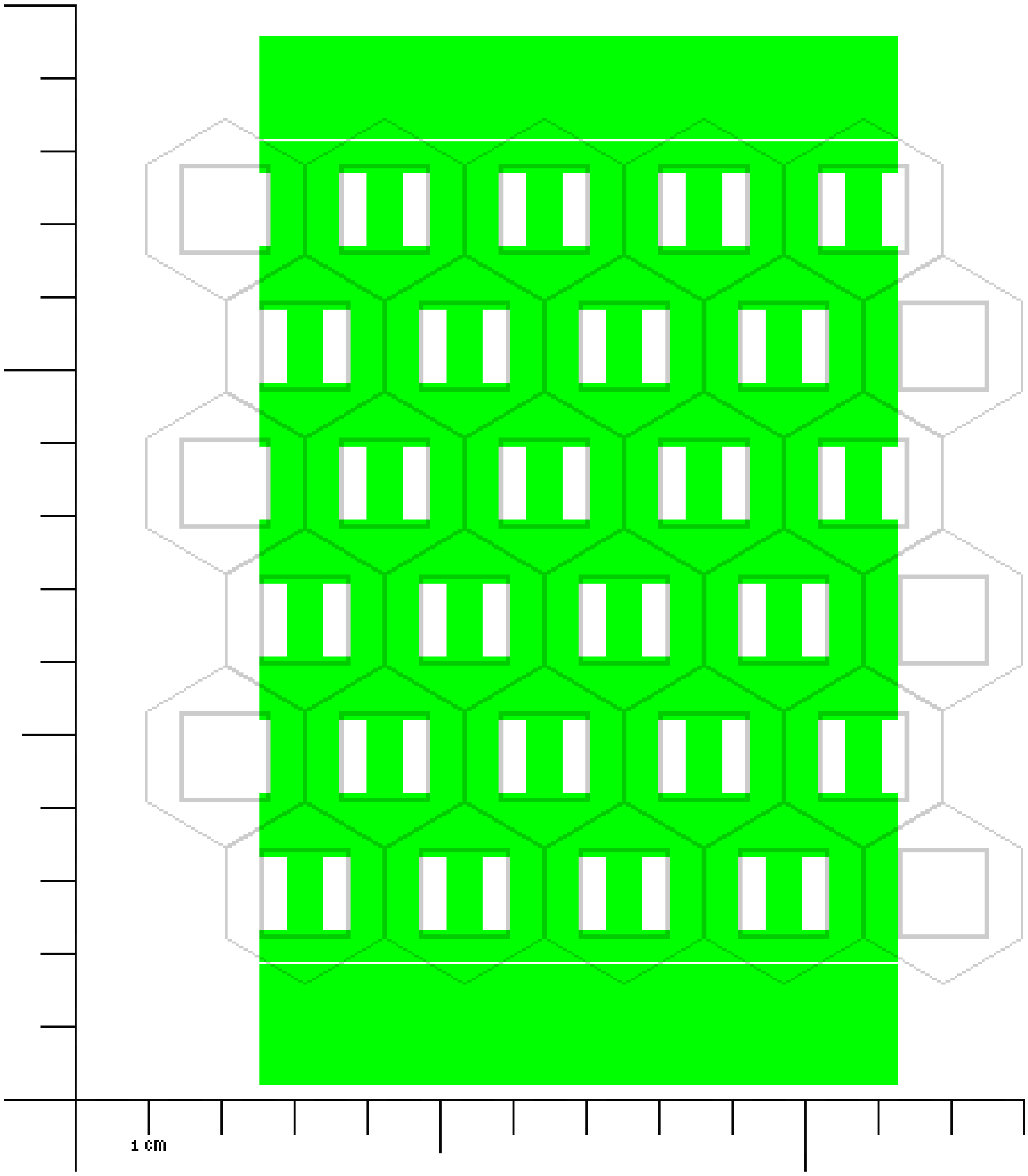

Illustration 14. Mechanical layout of the circuit board: Key places from the top. Keys are outlined with gray against the green circuit board that can be cut from the white lines in order to tile several boards for more key rows.

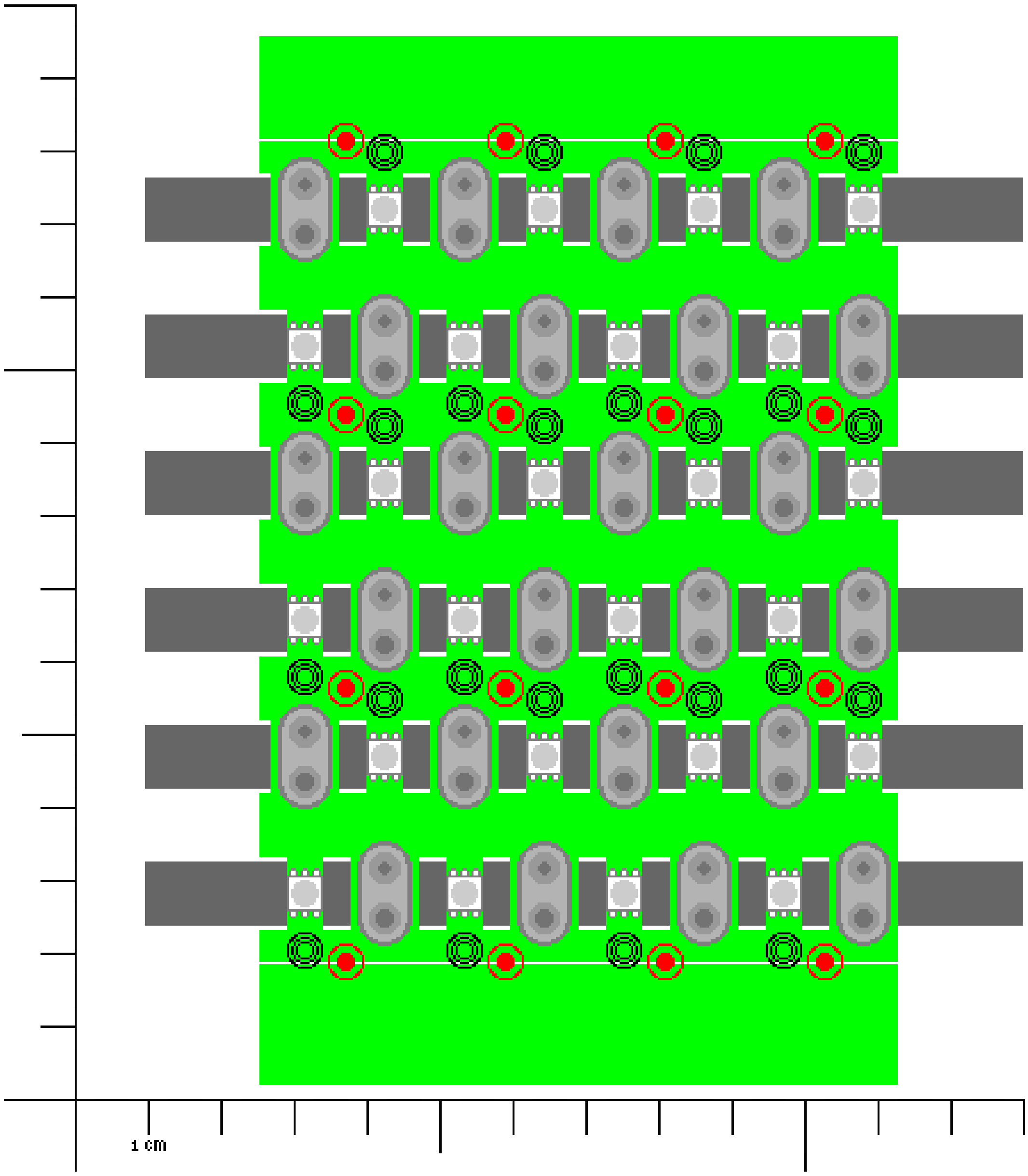

Illustration 15. Mechanical layout of the circuit board: Components. Surface mounted LED ICs in the middle of the keys and the rubber dome switches at the sides. Aftertouch strips are marked with dark grey. The black circles are key return springs and the red circles are support pillars.

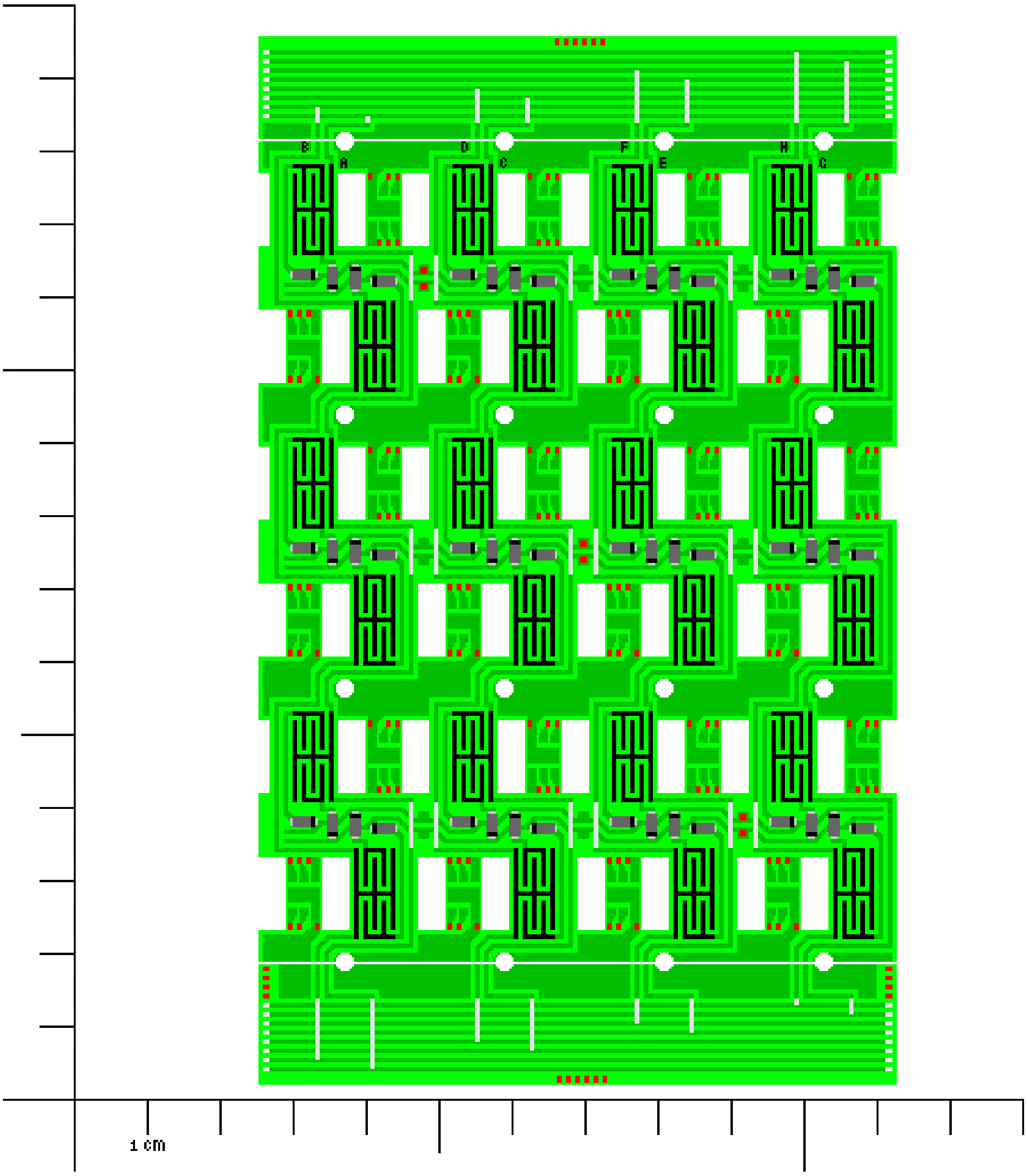

Illustration 16. Electrical layout of the circuit board: Top side. The red dots are through-holes for the tracks (dark green). Gray rectangles are the switch matrix diodes and white lines jumper wires.

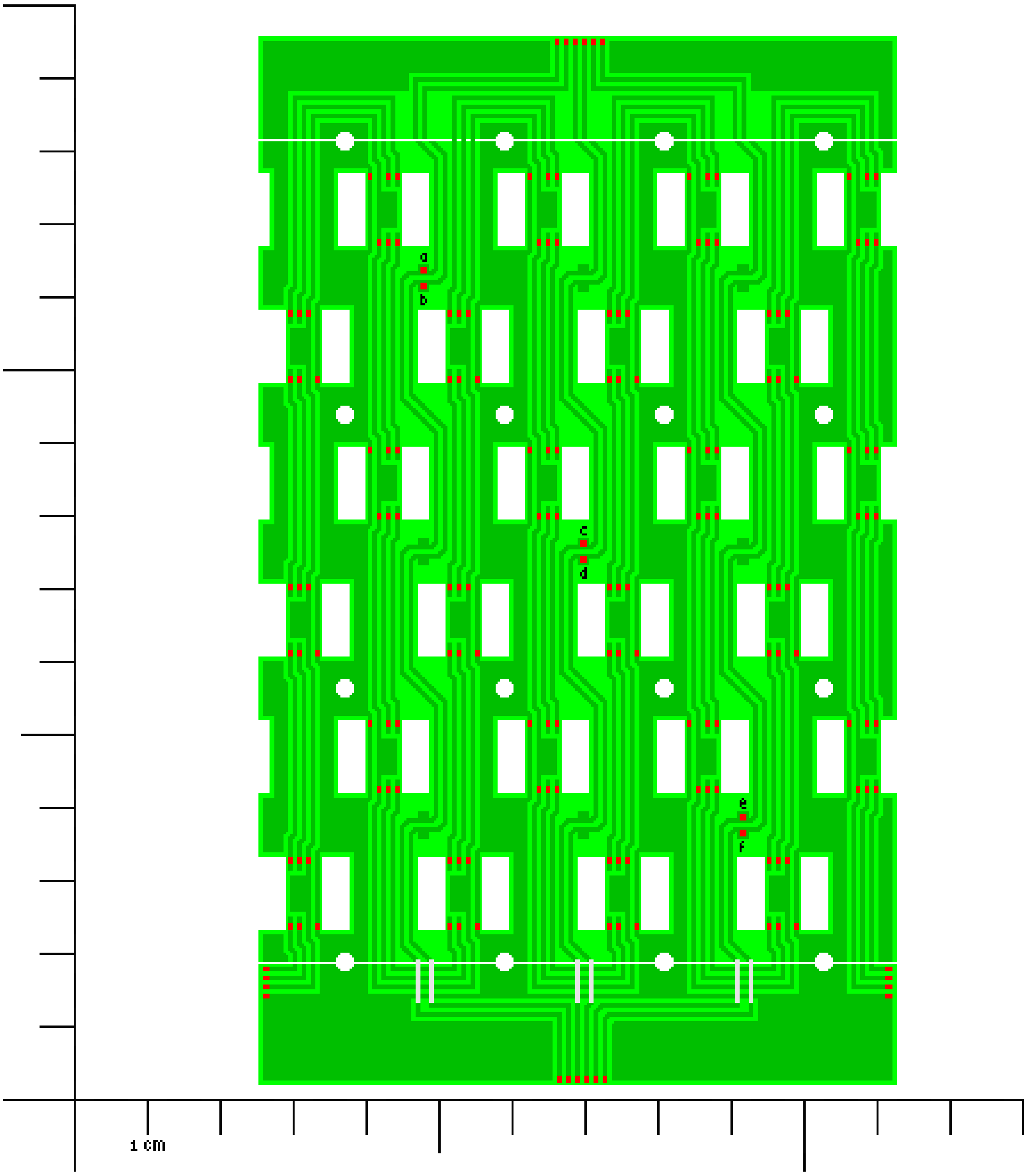

Illustration 17. Electrical layout of the circuit board: Bottom side seen from top (mirrored). The red dots are through-holes for the tracks.

4.3.1 Bottom layer

The layer at the very bottom (marked with black) is the base of the body of the module or the whole instrument. It could be made of metal to increase cooling through bottom.

4.3.2 Aftertouch

The force-sensitive resistor strips of the aftertouch (marked with red) are attached to the base with double-sided adhesive tape that also protects the underside of the strip. The strips are covered with 3 mm thick plastic foam tape (marked with dark grey) that may be further coated with textile tape to distribute the force more evenly and to prevent permanent sharp dents forming in the foam. Aftertouch strips run in full length under every row of keys. The strips can be connected in both parallel and series, due to the nature of them being effectively just two conductors with practically infinite resistance between them, and each pressed point forming parallel resistance. This facilitates extending of sub-keyboard-length strips by allowing the strips to face each other with their solder points at the opposite sides of the keybed. It also helps in connecting all strips into a single comprehensive aftertouch. Between the aftertouch strips, there is support structures for the upper layers and heat conducting foam or equivalent material for cooling through the bottom (not shown in the illustrations). All layers are spaced and attached by support pillars (marked with red circles on the circuit board) that consist of hollow cylinders as spacers and screws that go through the spacers and also all layers. The pillars partake in cooling also. The aftertouch must be on the bottom, because it cannot be penetrated by the support pillars, and it occupies a lot of surface area on its layer. Other electronics can easily be placed on top of it on upper layers. The reason for choosing the monophonic aftertouch is that it is easier to implement, and the polyphonic aftertouch may not result in significant improvement. The aftertouch tends to be sensitive but inaccurate, and fingers on the same hand rarely move completely independently. A separate aftertouch for left and right hands would make more sense, but is next to impossible to implement with anything other than two keyboards.

4.3.3 Support layers

The next (black) layer is the lower one of the two that hold the keys straight. The holes are rectangular, but unlike in the upper one, they are not exactly square to save space on the circuit board. The support layer can be shaped a little to make room for components on the circuit board. The material choice may be of low friction. Bushings incorporating for example silicone or Teflon™ (PTFE) may be added to the holes for better performance. The choice of lubricant may be affected by the close vicinity of the electronics. The upper support layer may be transparent for more even lighting, but the bleeding of the light to the adjacent keys should be taken care of in other ways.

4.3.4 Circuit board

The circuit board (green) is perforated to allow the forked key stem to pass through to the aftertouch layer. This structure enables both symmetrical load forces of the key and symmetrical backlighting. For smaller key widths and to save space on the circuit board, it may be sufficient to use only one side of the fork and move it with the backlight as close to the center as possible. At the worst it could lead to creaky keys when pressed and irregular aftertouch. The edges of the circuit board are shaped so that they can be tiled to form a larger keyboard surface. The back and front edges are wider for practical reasons, but they can be trimmed down to the white lines in order to expand the number of rows. In that case the circuit board tracks need to be reconnected between the modules. A double-sided (two-layered) circuit board is enough, although more layers may help, especially with the electrical shielding.

4.3.5 Backlight

The backlight consists of series of RGB LEDs with integrated control circuitry (such as the APA102) in SMD 5050 package (5mm*5mm). The LED chain begins from left front corner and continues backwards before proceeding to the right in order to simplify the configuring process. This way the light controller software does not need to know the exact width of the keyboard, but only the starting octave, and the graphics it plots are just cropped. In contrast, the number of rows has to be known, but in case of a Jankó keyboard it is likely to know how many keyboards or *manuals* are to be built into the instrument. Illustration 17 shows an example of chaining the LEDs back and forth. The LED control tracks can also return from back to front before starting the next column, which simplifies the software, but requires jumper wires. A major concern with the backlight is the heat it produces. It is possible to melt plastic or burn the LEDs with too bright settings. Here the heat is transferred through wide ground areas and other circuit board tracks to the sides, from where it is conducted to the bottom layer as mentioned previously. The open space above the circuit board allows for using a temperature-controlled fan as a backup also. The backlight requires a capable power supply. Power may be fed from multiple points on the circuit board for more even brightness and color in case the resistance of the circuit board tracks is too great.

4.3.6 Switches

The switches shown in the pictures are double silicone dome switches, but they can be individual switches too. The small size of the domes, the long travel, and the weight of the keys are likely to prevent them from being used as springs to return the keys back up. Therefore external springs are used, which also loosens the otherwise strict tolerances for the domes and allows varying implementations. For example a dome can have a long soft shaft, or it may be almost flat and actuated by a delicate spring or even a piece of soft plastic foam attached to the moving part of the key. In that case the velocity timing can be easily adjusted by varying the foam thickness. There is not much room for holes in the circuit board for attaching the domes. They can be kept in place by the support pillars if they are connected as a larger mat. Individual domes may also be attached with silicone glue or double-sided adhesive tape with some reservation. An alternative option to using rubber domes is to use simple wire switches and suitably bevelled key stem. Another is to use metal surface at the underside of soft plastic foam that is attached to the key to form an open structure push-button-style switch. The various switch designs do not require modifications on the circuit board.

4.3.7 Springs

The springs for returning the keys back up (marked with black) are attached to the circuit board at the side of the keys by soldering, and the material should be chosen accordingly. The springs are easy to make also by hand. The springs can also be located at the top between the upper support layer and the key cap. Even though it would help in the assembly and component placing, the spring would need additional space resulting in additional height, especially above the support. Longer key stem between the support layers is more stable. The springs may be cone-shaped to save vertical space while under tension.

4.3.8 Key body

The key stems (marked with blue) are transparent for the backlighting and made of for example polycarbonate for its strength. They are rectangular for easier manufacturing even with modest tools. The rectangular shape is also better than hexagonal in preventing the keys from rotating. The key stem can be made out of a single cuboid by cutting a notch at the other end to form the fork, or it can be made from three rectangular sheets attached together. The top part of the stem is solid to make the key more rigid. Small blocks are attached to the sides for the velocity switches and the return spring. The blocks can be topped with textile layer (dark grey) to act as a return dampener. Every second row of keys is rotated by half a circle to gather the springs closer together and to save space for the electronics on the circuit board.

The key caps are made of semi-transparent white plastic that diffuses light. They can be attached to the stem with acryl- or polycarbonate glue for best light transmission. They may also be transparent and painted with translucent layer of paint. The key caps may be shaped in a way that helps to target the fingers into the center of the keys, or they may be topped with stickers with embossed shapes.

4.4 Alternative key structure with reduced height

Illustration 18 shows an alternative key structure that utilizes mechanically more challenging and possibly machined parts. Some of these ideas can be utilized with the previously presented structure as well. The default material thickness is reduced from 5mm to 3mm. The key travel is kept the same, but could be further reduced. The components are divided between two circuit boards; the LED and the return spring are on the top, and the switches are on the bottom. The aftertouch is brought right under the lower circuit board, and the receiving dampener foam is in smaller pieces in the circuit board holes. The lower part or the floor of the key stem is attached to the upper part with tiny screws. The upper part of the key stem could be parallel rectangular sheets or merely support rods as in the previously mentioned *stacked* key primitive. The frame (black) is in two or three pieces and can be screwed together, and it is responsible for holding the key and guiding it smoothly. This structure offers more shallow keys, more effective backlighting and more effective cooling through the open top. The downsides are that it needs a lot of screws in accurately placed holes in the thin material, accurate placing of the dampener foam pieces, and cooling from the top may be uncomfortable for playing. Also, the structure is lightweight but may be too flexible and fragile for an instrument.

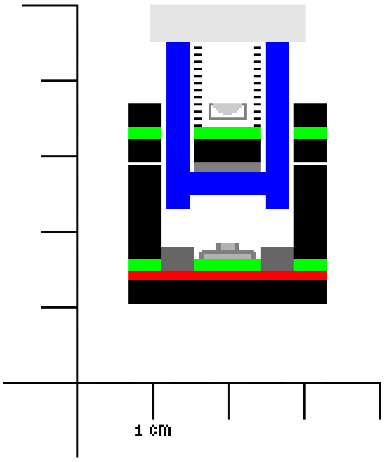

Illustration 18. A key structure with reduced height. The LED and the return spring are in the upper section and the switches and the dampener foam in the lower.

If 3D-printing is considered as a simple manufacturing method, and if the printed result is of sufficiently high quality, it is possible to use more complex key structure. Illustration 19 shows a basic principle of this type of fully machined key. The spring is moved to the top and outside of the key stem to make room on the circuit board, although there is very little space at the top. The outer edge of the key stem is cylindrical, and the two protruding blocks for the switches are located at opposite sides and are included in the key piece. The upper support layer is notched so that the key can be inserted and then locked into its place by rotating. Taking into account the shape of the key top, the key can be rotated by 60° (by one hexagon side) while holding the key below adjacent key tops. The fork and the lower support layer prevent the key from rotating back, and suitable grooves may also be used. The lower support layer forms a tight channel in which the key slides. The switches may also locate at the back or the front of the key stem, or both, rather than on the side. 3D-printing also enables fairly accurate and dustproof implementation of simple wire switches with two grooves of different length on the key. It is also possible to reserve grooves for possible O-ring bushings. The height of a stacked multi-layered key cannot be reduced further without reducing the key travel.

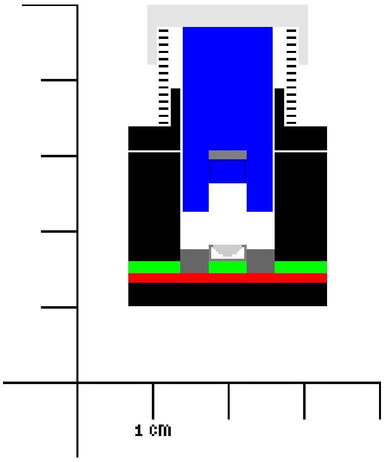

Illustration 20. A schematic of a fully machined key. The key stem is cylindrical and has protruding blocks in the middle. The spring is at the top. The black support structure guides the key stem and is shaped so that the components and moving parts fit.

5 NEW ELECTRICAL IMPLEMENTATION

5.1 Backlight

It is very useful to have the keyboard illuminated in dark environment, like on stage. The backlight can also be used to mark octaves, for example by having the black and white keys from the piano in different colors.

The backlight can be used to display current settings. Current values from other modules such as *Control Change* messages are easy to display, but changing and teaching new control knob numbers and other internal settings of the modules are more challenging. The backlight-display may not be completely independent after all, or some of the modules may need their own displays. In some cases it may be possible to re-wire the module display directly to the backlight controller logic board.

The backlight can be used in showy effects, like when the pressed key or column lights up or changes color. This effect is easy to implement by cloning the MIDI output to the light controller logic board input. MIDI keyboards with motorized sliders and knobs that are capable of moving by themselves in order to reflect the current settings work by having the corresponding *Control Change* messages that they output fed back to them as input. The same principle may be used for the backlight, which also makes it possible to use standard MIDI files or other MIDI equipment to output MIDI to the light controller board and to use the backlight for training purposes. Further on, the backlight can be used for example in interactive games. A drawback with lighting the keys with plain MIDI on/off messages is that all instances of the same note on the same column will light up.

Another great use for an idle (or not so idle) keyboard is animated effects and promoting. The backlight display suits extremely well for small and simple GIF animations that are also easy to make. Even if every instrument in a band is covered with LEDs, the keyboard wouldn't be left unnoticed, whether it was promoting the band or just the instrument.

The backlight consists of series of daisy-chained RGB LEDs with integrated control circuitry. A good choice is the APA102 LED IC chip that works with 5V and is fast and controllable with SPI (Serial Peripheral Interface). The chip utilizes a master-output-slave-input data wire and a clock wire between two units, and the data transfer takes place at the clock speed. The APA102 control data consists of a 32 bit header, n*32 bits for the color values of the n LEDs, and at least n/2 bit footer. A Jankó keyboard with full MIDI range (128 notes or about 10 octaves) has 3*128=384 keys. To refresh the lights with reasonably low latency like 20ms, or in other words have refresh rate of 50Hz, the data/clock speed needs to be at least

$$\left(32\text{bits} + 3\cdot 128\cdot 32\text{bits} + \frac{3\cdot 128}{2}\text{bits}\right)\cdot 50\text{Hz} = 625\,600 \text{ bits/s} \;\Rightarrow\; 630\text{kHz} \quad . \qquad (1)$$

For example an Arduino Uno with 16MHz clock is capable of SPI speed up to 8MHz, which is more than enough. A Raspberry Pi may also be used, especially if long or otherwise demanding animations are to be displayed. The maximum speed of the APA102 chip is unclear, but 4MHz has been reported to work, and some reports suggest that speeds of few tens of MHz would work.

5.2 Basic switch matrix

Illustration 21 shows a typical keyboard switch matrix. The diodes prevent false readings when multiple switches are closed (key ghosting), and the polarity may be reversed if needed. The paper-level vertical wires or columns represent 8 *pseudo-notes* (A-H) and the 16 paper-level horizontal wires or rows represent 8 *pseudo-octaves* (1-8) each consisting of two rows (a and b) for the velocity switches. These notations should not be confused with the traditional musical notations, especially when using the numeral *octave* is more justified here. Scanning takes place by sequentially activating each row at a time, and the columns are read as parallel 8 bits. Idle rows and columns may need pull-up or pull-down resistors.

The 8 columns and 16 rows result in a typical 64-key matrix that is used in most keyboards with 61 keys or about 5 octaves. It should be noted that the number of keys on piano keyboards follow historical convention and is not logically reasoned. More rows may be added to the matrix in order to increase the available notes. The maximum number of notes in a maximum sized MIDI keyboard is 128, which results in 32 rows and exactly twice the height of the matrix that is shown.

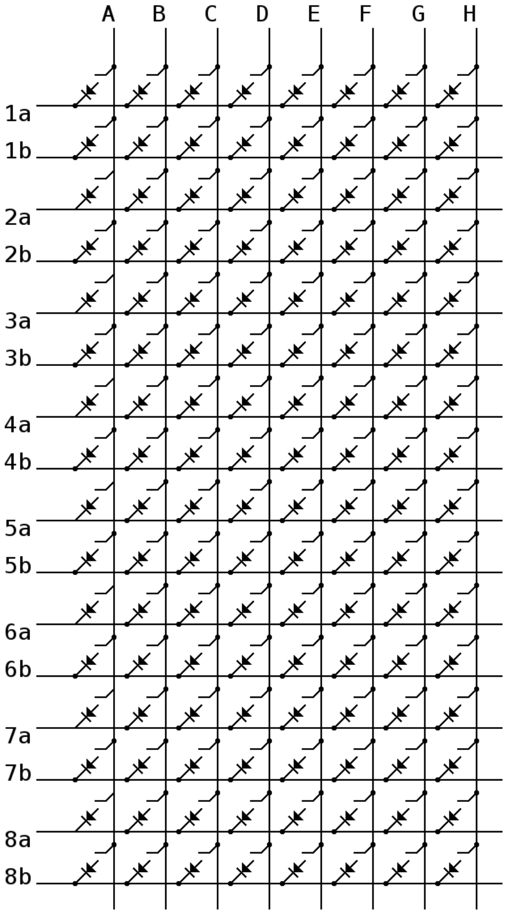

Illustration 21. A typical key switch matrix for the typical 64 (or 61) keys.

The columns reserve 8 pins on the controller board. The maximum of 32 rows are addressable with a 5-bit binary decoder, which reserves only 5 pins on the controller board. Thus the full 128 keys with velocity reserve only 5+8=13 pins.

Let us examine the scanning speed. Roughly estimated, a very slow but still musical key press takes about a quarter of a second. The velocity switches are difficult to adjust to work at the very extremes of the key travel, so a margin of one fourth of the total travel is spared for both ends, leaving half of the total travel and half of the total travel time between the switches. To make all 127 velocity values available, the velocity scanning interval T should be less than

$$T = \frac{1}{4}\mathrm{s} \cdot \frac{1}{2} \cdot \frac{1}{127} = 0.984\mathrm{ms} \quad . \tag{2}$$

The scanning speed or refresh rate f has to be multiplied by the number of rows, which is 32 at maximum, giving

$$f = \frac{1}{T} \cdot 32 = 32.5\mathrm{kHz} \quad . \tag{3}$$

For example an Arduino with a processor running at 16MHz clock speed would have almost 500 clock cycles for every scan. In reality this is less. The scanning speed should be greater for better accuracy in fast key presses and for smaller overall latency. Greater resolution is also beneficial when using velocity curves and scaled values. It should also be noted that instructions written in C often compile to multiple processor instructions that also may take multiple clock cycles to complete. Nevertheless, it is likely that an Arduino will suffice to scan the keyboard. With optimized code and simple interrupt- or timer based multitasking the same single controller board is also likely to be able to send MIDI data through serial port (UART).

5.3 Jankó keyboard switch matrix

A Jankó keyboard matrix is large and may require faster scanning speeds, more I/O pins and more processing power than the typical 64 keys. The easiest way to wire a Jankó-keyboard is to connect the three identical double rows of keys together electrically and using the switch matrix presented previously (illustration 21). The circuit board presented previously (illustrations 18 and 19) has an option for through-holes that connect the rows, or they can also be connected in the connector. This wiring is compatible with most MIDI keyboard controller electronics, taking into account the diode polarity.

A more sophisticated method is to use three separate matrixes and scanners for the three double rows of keys of a Jankó keyboard. The row wires of the three matrixes are shared, but the 8-bit readouts are connected to different input pins. This way the scan speeds may be kept the same, but pin count is added by 16 compared to the single key row case, resulting in 5+8+16=29 pins. The separate matrixes are shown in illustration 22. To achieve this connection matrix on the previously presented circuit board, some modifications are needed.

The middle key row has to be separated by cutting the 8 column tracks A-H between the springs, and the tracks in question, which are also visible as jumper wires, need to be bridged to adjacent pseudo-octaves for example with a flat cable. The back and front rows already have tracks at the edges for the connections. The row wires of the three Jankó rows are connected together in each pseudo-octave like in the previous example.

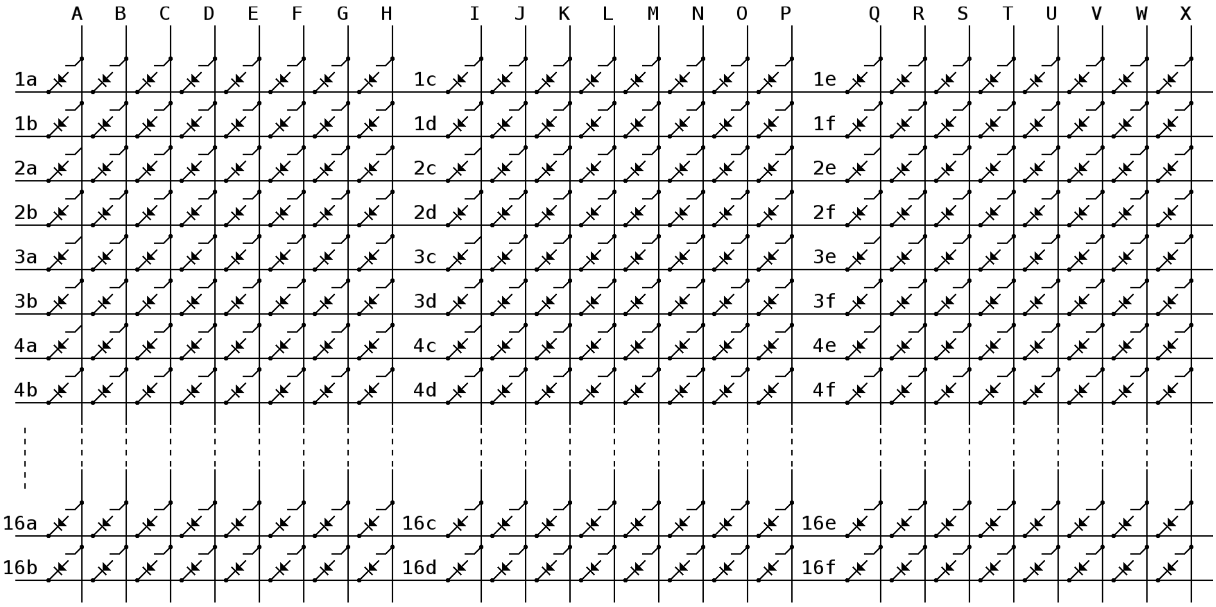

Illustration 22. Three switch matrixes for the three double rows in a Jankó keyboard. The numbers denote pseudo-octaves, capital letters pseudo-notes, and small letters the velocity switches.

It is even possible to use completely separate MIDI controller logic boards for the three Jankó double rows, if the double messages are interpreted at some other stage. When two *note on* messages are received, the sound may just continue until two *note off* messages are received, like the lever arm implementation would do. The second *note on* may also re-trigger the sound, effectively sending a *note off* before the second *note on*. The note may be re-triggered also when releasing one note while still holding the other. Sometimes it would be useful if releasing the most recently pressed key re-triggers the sound, but releasing the previous or older key does not. This function is unachievable with mere MIDI messages that do not distinguish the keys of the same note.

A very functional keyboard wiring is represented in illustration 23. The whole 128-note (3*128 key) keyboard is divided into four sections from left to right. Each section contains 32 notes or 4 units of 8-note pseudo-octaves and 3 Jankó double key rows. The previously presented circuit board illustrations (16-19) equal to a single pseudo-octave in a 4*6=24 key grid, so each section consists of 4 of these. If the keyboard module size is 16 notes, each section consists of two modules, and the full-length 128-key MIDI keyboard equals to 8 modules. All pseudo-octaves and double rows of keys in a section share the 8 column wires that go to the controller board as the output from that section, which also helps in chaining the modules. A total of 4*8=32 column wires go to the controller board. Each logical switch row (of the total 6) and pseudo-octave in a section has its own row wire (totaling 24), but the row wires are shared between the sections. With a binary decoder, a total of 5+32=37 pins are required. In practice the 4 sections mean that the keyboard is read from 4 locations simultaneously. This helps a little with the I/O handling overhead and also the frequency related interference in the I/O lines, and if further needed, the row activating order may be changed as well. Shorter keyboards utilize fewer pins on the controller board, like for example the 64 notes that only require 16 column wires, but the software needs not to be changed, and the scanning speed may remain the same. Depending on the implementation and the number of available I/O pins on the controller board, even more than 128 notes may be used, and also the 128 notes may be divided into two 64-note *manuals* to create a space saving seamless double keyboard. The type of key switch matrix presented here also simplifies the circuit board design by allowing the tracks to form an even grid in a single layer.

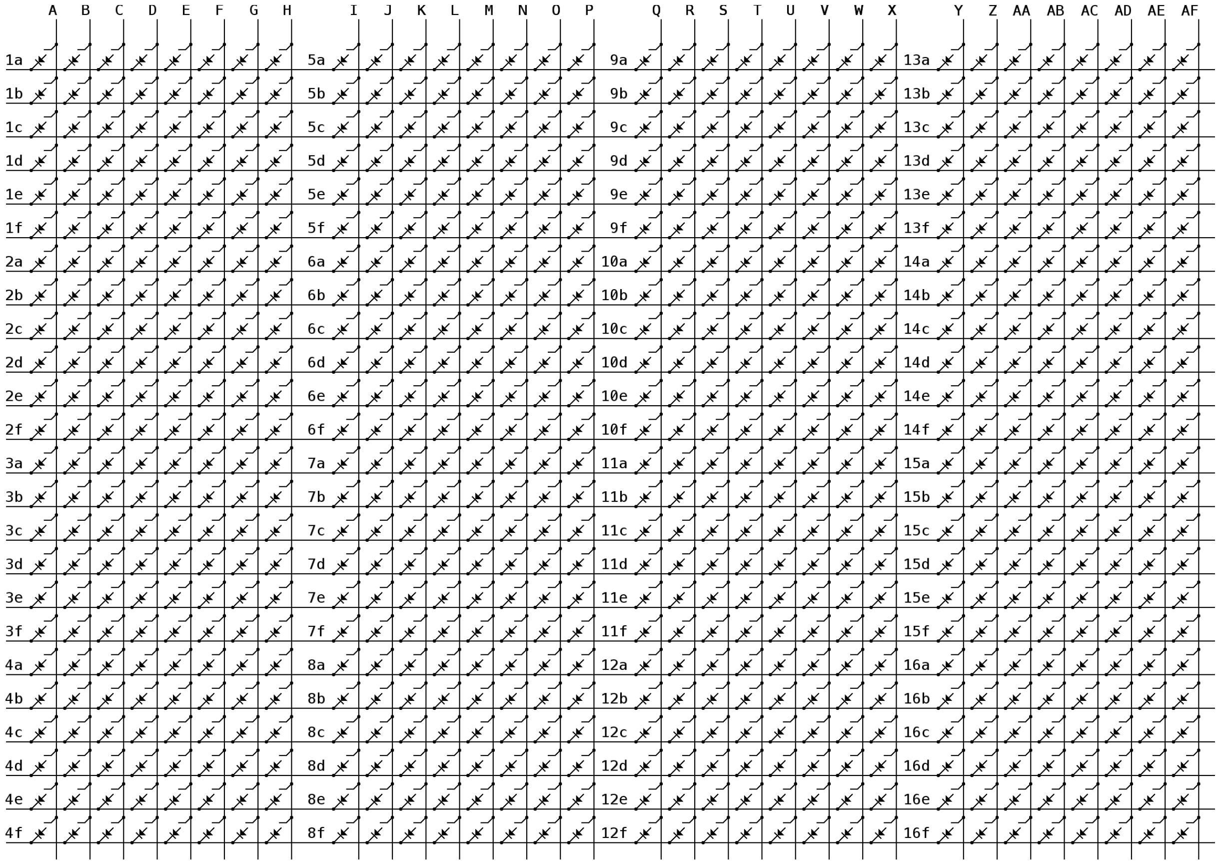

Illustration 23. The full key switch matrix utilizing 5+32 pins. The numbers denote pseudo-octaves, capital letters pseudo-notes, and small letters the velocity switches. The full 128-note keyboard is divided into 4 sections.

5.4 Wiring of MIDI controls

Usual MIDI controls are the *aftertouch*, *pitch bend* and *modulation* wheels, *expression* and *sustain* pedals, *breath controller*, and different knobs and sliders. In practice they are potentiometers that divide the used voltage, and they may be connected to the analog inputs of the controller logic board.

A special attention should be paid to the connectors for the external controls, because usually the same type of connectors are used for audio. The audio and external control connectors should be located far apart to avoid mistakes, and a failsafe check for example at power-up is recommended. Using different connectors and adapters may be considered.

Let us view an example of using the popular Yamaha BC3 or compatible breath controller. It uses the same 3.5" connector as ordinary earphones, but uses 9V DC operating voltage, which is more than enough to cause damage to both ears and earphones.

The sound pressure level is usually given as the relation of the RMS pressure level p_{RMS} to the smallest audible RMS pressure level $p_{0,\,\mathrm{RMS}}$ in logarithmic bel scale and noted as dB (SPL).

$$x \text{ dB(SPL)} = 10 \cdot x \text{ B(SPL)} = 10 \cdot \log \frac{p_{\mathrm{RMS}}}{p_{0,\mathrm{RMS}}} \tag{4}$$

The frequency dependency of ear sensitivity can be taken into account by multiplying with an approximated weight function A, and the weighted sound pressure level is noted with dB(A). At frequency of 1kHz, the weight function A has value of 1, and dB(SPL)=dB(A) @1kHz.

The technical datasheet of a pair of in-ear earphones (Shure SE535) mentions sensitivity of 119dB(SPL)/mW and impedance of 36Ω at 1kHz, which means that the pain threshold of 120dB(A) is reached with power of about 1mW. In volts this corresponds to

$$\begin{gathered} P = UI = \frac{U^2}{R} \quad \Rightarrow \\ U_{\mathrm{RMS}} = \sqrt{PR} = \sqrt{0.001\mathrm{W} \cdot 36\Omega} \approx 0.19\mathrm{V\ AC} \end{gathered} \quad . \tag{5}$$

In consumer audio equipment, line-level signal is defined as -10dBV (decibel-volts) with reference of 0dBV=1V$_{\mathrm{RMS}}$. The misleading notation means reduction to -10dB=0.1 in power, not voltage. The consumer audio line-level voltage is

$$\begin{gathered} U = \sqrt{PR} \Rightarrow \\ U_{\mathrm{RMS}}(-10\mathrm{dBV}) = U_{\mathrm{RMS}}(0\mathrm{dBV}) \cdot \sqrt{-10\mathrm{dB}} = 1\mathrm{V}_{\mathrm{RMS}} \cdot \sqrt{0.1} \approx 0.316\mathrm{V}_{\mathrm{RMS}} \end{gathered} \quad . \tag{6}$$

In professional audio equipment, line-level signal is defined as +4dBu (decibel-milliwatts unloaded). The reference level of 0dBm (decibel-milliwatts) was defined as the level to dissipate 1mW of power in a 600Ω load in impedance-matched transfer lines, but such practice is no longer used in audio. The professional audio line-level voltage is

$$\begin{gathered} U = \sqrt{PR} \Rightarrow \\ U_{\mathrm{RMS}}(0\mathrm{dBu}) = \sqrt{0.001\mathrm{W} \cdot 600\Omega} \approx 0.775\mathrm{V}_{\mathrm{RMS}} \\ U_{\mathrm{RMS}}(+4\mathrm{dBu}) = U_{\mathrm{RMS}}(0\mathrm{dBu}) \cdot \sqrt{+4\mathrm{dB}} = 0.775\mathrm{V}_{\mathrm{RMS}} \cdot \sqrt{10^{0.4}} \approx 1.23\mathrm{V}_{\mathrm{RMS}} \end{gathered} \quad . \quad (7)$$

Even line-level signals may be too powerful for earphones. A simple de-coupling capacitor is not enough for protection.

5.5 Controller logic board

Different MIDI devices can be multiplexed or merged to seem as a single device outwards. The multiplexing can be done by a separate MIDI multiplexer device or circuit, or by the main logic board. The USB can be used as well. Multiple controller boards may be used for specific tasks if needed.

An Arduino Uno runs at 16MHz clock. It has 14 digital pins of which 2 can be used for one UART serial port and 4 for SPI. It has 6 analog pins, which can be used as digital I/O if more digital pins are needed. Thus the Arduino Uno has not enough pins for the sectioned key scan with separate rows, but it is able to scan the full-sized 128-note keyboard in the sequential way (using 2 digital pins as UART serial port for MIDI output, 1 analog pin as digital, and 1 USB). Alternatively, it is suitable for controlling only the backlight (using 4 digital pins for SPI, 2 digital pins for UART serial port for MIDI input, and 1 USB), and possibly up to 8*6=48 control knobs without a binary decoder at the same time (using the 8 free digital pins and 6 analog pins).

An Arduino Mega has more pins than the Uno.

An Arduino Due runs at 84MHz clock. It has 54 digital I/O pins of which 8 can be used for 4 UART serial ports and 3 for SPI (more specifically for the 3 slave-select lines while the rest are located at a separate *ICSP header*). It has 12 analog input pins, which can be used as digital I/O if needed. It also has one additional USB port (2 in total) for the use by the programs. Thus the Arduino Due is capable of reading the full keyboard (using 37 digital pins), sending to 1 MIDI port (using 2 digital pins as 1 UART serial port), receiving from 3 MIDI ports (using 6 digital pins as 3 UARTs), controlling the backlight (using 3 digital pins as SPI), reading up to 12*6=72 controller knobs without a binary decoder (using the 12 analog pins and the 6 free digital pins), outputting to USB (using 1 USB port) and receiving from USB (using 1 USB). I2C communication is possible through dedicated pins or some of the general-purpose I/O pins.

Arduino Due utilizes 3.3V logic instead of the older 5V logic, which should be taken into account. The voltage is relatively easy to lift with external converters to match the 5V of the LEDs to avoid possible compatibility problems and damage during accidental liquid spills. The 5V and 3.3V difference is still likely to cause less damage than when using 12V LEDs.

A Raspberry Pi is problematic in this application area despite the low cost, high processor speed and highly versatile programming possibilities. The number of I/O pins needs to be expanded through the SPI or I2C ports. It is likely that a real-time operating system needs to be used instead of the popular Raspberry Linux versions due to the multitasking nature of the latter and the resulting variation in the I/O pin timing. For example the keyboard scanning loop must be carefully synced to obtain consistent velocity values, which means that other processes, even of lower level, should not be allowed to interrupt it to gain processor time in irregular manner.

There are different keyboard scanner chips available, and some of them are even aimed for velocity-sensitive MIDI keyboards. They are very affordable, but the reason for favoring an Arduino is the customizability, easy availability, compatibility and long term support.

5.6 Software overview

The most time-critical part is the scanning of the velocity switches. This should be implemented with timed interrupt, like in timer1 overflow interrupt service routine. The two switches in each key, called *A* and *B*, are first scanned and stored in corresponding boolean matrixes (or vectors) *A* and *B*. A boolean matrix *History* holds the information whether the key is going down or up, and integer matrix *Velocity* holds the current velocity values. Every key or the corresponding matrix elements are then processed as shown in table 4.

Table 4. Velocity implementation. Possible variable values of a single key [x,y] and the resulting action. A and B denote the velocity switches.

A [x,y]	**B [x,y]**	**History [x,y]**	**Description**
0	0	0	Key not pressed. No action.
1	0	0	Key moving down. *Velocity++*.
1	1	0	Key down. Output note ON command with *Velocity* to the send buffer. *Velocity* = 0. *History* = 1.
1	1	1	Key still at the bottom. No action.
1	0	1	Key moving up. *Velocity++*.
0	0	1	Key up. Output note OFF command with *Velocity* to the send buffer. *Velocity* = 0. *History* = 0.

At the output, the velocity values are scaled suitably, and different velocity curves may be applied. Also, proper note values are chosen for the keys (from a similar matrix), and the rules for the keys with the same note may be applied (a few example rules were discussed previously in the *Jankó keyboard switch matrix* section). The output may be easiest to handle by a subroutine, which is also in charge of the MIDI command shortening.

The controls are not very time-critical, but their values should be scanned at reasonable intervals. The control scanning may be synced to the keyboard scanning so that the controls are scanned after the keyboard, but only in every nth cycle. Similarly, the pitch bend and modulation wheel may be scanned more often than the other controls. The note on/off commands are crucial, and they are not likely to overload the MIDI DIN port (UART), but *Control Change* messages need to be limited. The control scanning speed may be higher if USB is used. This may be changed manually by the user or automatically at the startup initialization.

The backlight refresh is not very time-critical, but because the LEDs work like shift registers, the image should be sent quickly through the SPI and stay visible for a while to avoid continuous updating and distorted image. The backlight control will likely work fine in the main program loop. The amount of RAM and processing power may be limited for the most demanding animations, and a Raspberry Pi may be used for the backlight altogether. In that case it is enough to send only the pressed key numbers to the Raspberry through the SPI (or UART or I2C). This may also be done in the output subroutine. Both backlight methods may be implemented for compatibility and possible upgrade later. It is important that the backlight does not degrade playability and causes minimum to zero delay.

The multiplexing of MIDI input ports to MIDI output port is not time-critical and can be done in the main program loop, because the UARTs have a 64-byte serial buffer. A concurrent write to the output buffer with the scanning routine must be avoided by disabling interrupts briefly when writing a packet. This also gives priority to the main keyboard, which is desired.

Sending and receiving MIDI signals through the one *native* USB port on the Arduino Due is pretty straightforward using the MIDIUSB library. However, communicating with peripheral devices (USB slaves) connected to an Arduino's USB port that is serving as a host (USB master) is more complex. The other USB port on the Due is used for the *programming*. The serial firmware may be flashed with modified version to allow more native functionality of the port, although the Arduino development environment will not be available until it is flashed back. Another solution is to add a USB shield (which utilizes an SPI port and a few other pins). At the time of writing, by default with some exceptions, only one USB host shield is supported, and only one connected USB device is supported, excluding USB hubs. Therefore the used shield and libraries should be chosen carefully, and additional in-depth work is likely required. While in theory it is possible to connect the modules to an Arduino through USB ports, it may not be at the top of the list of features to implement yet. On the other hand, it is quite easy to use separate Arduinos to convert USB input into MIDI, faster serial (UART), or SPI output that is received by the main controller. Not to mention that a Raspberry Pi is much more capable of handling the USB, and also manipulating the traffic.

The instrument settings may be modified through the Arduino subsystem. It is also relatively easy to transfer the settings as a text file between the instrument and a computer by the means of *MIDI System Exclusive* messages. The user interface contributes to the overall experience and should be paid attention to as well. The settings menu may be triggered by a special reserved key or a key combination. The different settings may be browsed for example with the modulation and pitch bend wheels, or called directly from the keys. The backlight can be used as a text display, in which the text should be scrolling to support various keyboard widths. Confirmation of settings and possible navigational keys can be shown with color and animation. The first level or item in the menu may be a *quick menu* that offers color-coded shortcut keys to commands such as *Program Change*, *MIDI panic* and *Reset*. These functions may also be offered in a button module. If the backlight is not used, the readouts may be output to a connected computer. They can be sent through MIDI as *System Exclusive* messages that show in a MIDI monitor application or through the virtual serial port of the Arduino development environment into the debug console.

6 ROADMAP TO KILLING THE PIANO

6.1 The biggest obstacles and failures

- There has been several patents involving the Jankó layout keyboards. Some have expired, and a few are still active. None of them has helped in popularizing the layout in over a hundred years. They may even discourage small companies or individuals from entering the scene. It is challenging to make profit on the specific field, just as it is on the heavily competed area of piano layout keyboards. The solution is openness in software and hardware to lure more actors and enthusiasts to join forces.

- Exotic or experimental instruments usually depend on a single manufacturer, and the development and support may end at any time. Spare parts or equivalent products are not available. Openness in hardware and software helps in this too. The instrument proposed here bases on mature and easily available technology and is simple enough to be built from the very beginning by a hobbyist in a garage.

- The novelty keyboard instruments usually cost a lot. They are packed with features to rival a whole range of traditional keyboards and to appear as instruments to be taken seriously. The proposed instrument is modular, which enables entry with the bare minimum hardware at minimum cost, and the instrument can be expanded later as needed. The main attraction of the keyboard is the keyboard itself, although it can be merged with a synthesizer of choice.

- People do not want to switch from piano layout, because they do not want to learn a new instrument. This claim is not exactly accurate, because the Jankó layout is not completely new, but shares similarities with the piano layout. The Jankó layout is also faster to master than the piano, along with other benefits.

- People do not want to learn an exotic instrument, because it exists nowhere else, and piano keyboards are available everywhere. While this claim holds true, it actually has no practical meaning to majority of keyboardists, even if thought so. Nowadays keyboards are usually personal instruments with settings and sounds customized by the user. Most of the playing takes place with a player's own instrument and musical gear. A Jankó keyboard is also narrower and thus more mobile than the piano keyboard of the same pitch range.

- The piano keyboard has a lot of teachers and learning material easily available, while the Jankó has not. This claim is misleading. The two layouts are similar, and many teachings can be easily translated to the Jankó layout, especially if the keys are colored black and white correspondingly. The piano lessons usually contain music theory in addition to the repetitive practicing of the fingering, and in such cases the Jankó layout is even more logical and easier to comprehend. While the Jankó layout certainly needs practice in fingering, the additional possibilities for the finger positions offer more personal choices and better ergonomics. After learning the basic fingering the Jankó layout does not even need as much practice as the piano layout.

More about the different layouts can be found from: Peltokoski, 2014.

6.2 Future development

In the first phase the instrument can be built by hobbyists and enthusiasts. It may also be a prototype for further refining. Especially the velocity switches may need adjusting to better match the chosen controller electronics.

The second phase is to build keyboard modules. The keys are the crucial part of the keyboard, and they cannot be obtained from anywhere. The keyboard modules may be retrofitted to existing keyboards, and they can be wired differently and utilized in other exotic keyboards too. Therefore there is a small market for selling these modules to hobbyists, even without the backlight installed. A retrofitting service to existing synthesizer or MIDI controller keyboards may be offered, and also do-it-yourself retrofitting kits. The plastic parts of the modules are simple, of only a few types, yet numerous, so they are suitable for injection molding. Having these injection molded by a proper subcontractor will cut down the production costs but will not prevent individual persons from making spare parts at home.

The third phase is to program an Arduino-based controller logic board. It may be offered as an expansion module, and it may be used in different MIDI keyboards also. Arduino community already has several working open-source projects and libraries involving MIDI keyboard scanning and MIDI communication. The controller may start from the very basics, because it is easily upgraded. Features like velocity, aftertouch, pitch bend and pedals will follow. The backlight control is more device-specific and especially the animations may be implemented on a dedicated logic board if needed.

The fourth phase is to build a complete keyboard that has a frame that can be expanded with modules. The frame may be offered in a few different sizes, and there should be space for expansions like knobs and sliders. An independently working module with motorized knobs and sliders is a great addition to the product line. An option to power the keyboard via USB may be added. An independent synthesizer module or synthesizer software may be considered with some reservation.

The fifth phase is to offer extra contents for the keyboard, like different light effects and animations, and tablatures as MIDI files to feed to the keyboard (the backlight) for practicing. Other learning material and integration to musical games will also help in popularizing the keyboard.

7 RESULTS AND CONCLUSIONS

The keyboard described in this work features hexagonal grid of keys in a Jankó layout with velocity sensitivity, monophonic aftertouch and individually backlighted keys. It consists of modules for easy customization, expansion and upgrading. The keyboard can be built with rudimentary tools in home environment. It is therefore easily repairable independently from the manufacturer or spare part availability. Using more sophisticated manufacturing methods, such as plastic injection molding and machined rubber dome switches, will help but is not necessary.

The keyboard utilizes the same sensors as nearly all of the current keyboards; two switches for measuring the key press velocity from the time difference, and a pressure-sensitive resistor for measuring the aftertouch while the key is held down. The keys have layered structure with the aftertouch on the bottom layer and the switches and the lights on the upper. The key stem is forked and traverses partly through the upper layer.

The keybed is small enough to be inserted in the place of a piano keybed in most electronic keyboards, not taking into account the height and possible frame support structures. The duplicate rows can be wired together on the circuit board to facilitate retrofitting to existing controller electronics. Some adjustment in the velocity switch timing may be needed, which may be done at the simplest by adding soft plastic foam stickers to the key stems.

The keyboard can work as a stand-alone MIDI device by using an Arduino microcontroller as the main logic board. The Arduino Due has a fast processor and lots of I/O pins, allowing a maximum sized (128 notes) keyboard matrix to be scanned and interpreted quickly and from 4 locations simultaneously. The scanning is synced with timed interrupts. The same software is applicable for all keyboard widths and needs not to be changed when adding modules. USB may be used for MIDI output needing no drivers at the other end. However, using USB for MIDI input from the other modules proves to be challenging using only Arduinos.

The backlighting is based on daisy-chainable LED IC chips. The heat produced by the LEDs is transferred to the bottom for cooling. Active cooling and limiting of the maximum brightness may be considered. The backlighting can be handled by the main controller logic board, an additional Arduino, or even a Raspberry Pi in case of more demanding effects. The different controller board modules may communicate through UART, SPI or I2C.

The keys are not specifically weighted and do not feature hammer action. Adding weight to the keys is theoretically possible, but it heavily contributes to the total weight of the instrument. A hammer action mechanism may be tried to be added under the keys, but it will increase the height and weight of the keys. Also, making the keys unequal by imitating a limitation of an acoustic instrument may not make much sense here. For sharper key movement it may be possible to utilize small and powerful magnets at the bottom, for example.

The presented keyboard can enter the market gradually unlike most of the other exotic or conventional instruments. It is open-source hardware and software, and independent from the manufacturer. The product may start off as a keyboard module for a Jankó or other exotic keyboard instrument, or even as the mechanical keys only. It may be a retrofit replacement for piano keyboards at first and followed by a stand-alone controller module. A complete keyboard frame may follow, along with a range of independent expansion modules. The entry price with a minimal system is low, and the instrument may be expanded later.

One of the greatest obstacles is unreasoned prejudice towards the new keyboard layout. Compared to the piano layout, the Jankó is more logical, easier to comprehend, more ergonomic, and it offers new ways to play. It is also a little bit narrower and thus more mobile. Sharing knowledge is important, but it is also crucial to have an easily and affordably approachable keyboard. Openness makes the keyboard more attractive and directs the futile competition between marginal instruments towards the piano layout itself. This keyboard has no competitors, only cooperators.

Illustration 24 shows a 3D-printed prototype of the key that is capable of being fully handmade. For playable feel the gap or the clearance between the key and the support should total less than 0.5mm per axis. The downside with the rectangular key stem is that the edges and especially the corners of the support can be felt from the keys when playing. To counteract this, the keys can be made tighter with smaller gap and the corners can be rounded a little. Also, with some reservation, the lower support layer may have two holes per key, and the fork spikes may be round. The key return springs may need a little more vertical space. The lower support layer may be modified to accommodate better heat conduction. The tips of the fork in the key stem may need sharpening or thinning and rounding to better match the properties the receiving dampener foam on the aftertouch. In a maximum length keyboard (10 octaves or 384 keys) the keys would weight less than 5kg allowing great mobility. The design is resistant to dust and water spills, but may need cleaning. Full waterproofness in a mechanical keyboard is only achievable by coating the keyboard, which greatly affects the playability. The electronics do not operate specifically on audio frequencies, which helps in keeping the disturbance to other musical gear at minimum.

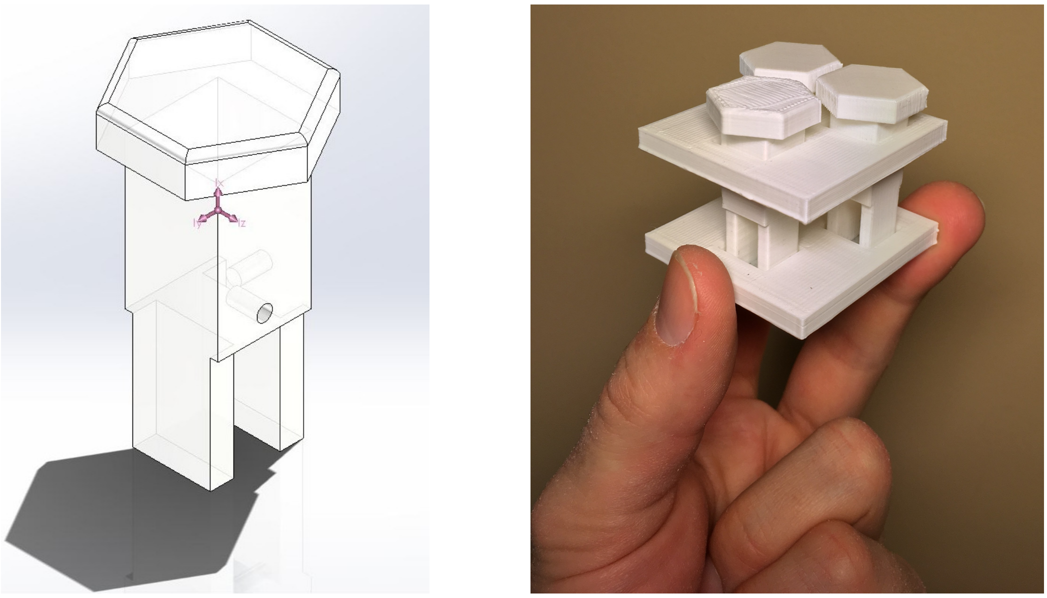

Illustration 24. 3D-modeled design of the key (left) and printed prototype with some keys and support layers (right).

If more advanced manufacturing methods such as 3D-printing are considered as an everyman's tool, it is possible to use more complicated key structure resulting in similar but even more sophisticated implementation. Among other improvements, 3D-printing enables reduced height, simpler assembly, press-and-turn attaching of the keys, improved cooling, using of O-ring bushings, and possibly even simpler switches.

8 DISCUSSION

A choice for an instrument bases largely on a personal opinion. Every single keyboard has its weaknesses and strengths, and detailed comparison is left for the reader. Fighting over superiority among the alternative keyboards is counter-productive. However, a few personal observations should be mentioned.

During the writing process, I received a Chromatone keyboard from Chromatic Music Lab; a beginner-level Jankó keyboard that bases on the lever arm implementation. It has the basic features such as pitch bend, velocity sensing, sound sample library, speakers, MIDI output and even battery compartment. Its greatest shortcoming is the typewriter-like key caps that are not suited for glissando (note slide) and need to be glued to stay in place. Also, the key press feels a bit mushy or spongy, which is expected from an instrument aimed for beginners. Otherwise the playability is surprisingly good, especially considering such light all-plastic parts. It is definitely a great instrument to familiarize the new layout with. When I started playing with a real instrument, it quickly became apparent that the familiar marking of the black and white keys helps and is practically needed to navigate on the keybed, at least at first. Another interesting yet somewhat expected attribute is that the alternating rows and positioning fingers on them requires a little more attention than on the piano keyboard. At first the playing felt a little more alien than I had anticipated, and the transition from the piano layout required a little more work, but after a few days of practicing the chords and melody progressions felt more natural and relaxed than they ever did before. This layout seems to be my personal favorite, and I would recommend it to others as well.

I have also tried the Axis 64 keyboard from C-Thru Music; a surface of keys in a hexagonal grid with *harmonic table* layout (practically 12 notes per octave divided into 4 rows). Although the layout is good with chords, it did not feel natural to me. But the keys in it are absolutely of professional grade and the playability is near perfection, despite the missing aftertouch. The high quality rubber dome switches contribute to the excellent playability without doubt, but the proposed implementation does not rely on them so much, which gives hope in achieving the same with mere plastic and springs. The key width in the Axis is slightly larger than the proposed, and the key travel slightly less. While the Axis keys are something to look up to, the proposed measurements base on the piano keys and a differently aligned layout. They may still be adjusted later.

There is no real season for manufacturers not to offer keyboards with the Jankó layout. The manufacturing costs of the alternative keys are not significantly higher than of the piano keys when machined industrially. The big manufacturers are sleeping in this regard while they concentrate on developing the synthesizer side only. Some manufacturers offer electronic instruments and even keyboards with button accordion layout, but not Jankó, even when the structure is practically the same.

What the big manufacturers may not want is modularity and expandability, because they will cut down the profit margin. Usually the whole keyboard family is manufactured on the same product line to save in manufacturing costs, and the cheaper end models are artificially limited in features and sold cheaper, even when the hardware is technically as capable as in the more expensive end models. Upgrading and repairing a keyboard necessitates buying a completely new and significantly more expensive unit, which is profitable for the manufacturer. On the other hand, modularity reduces waste and minimizes the impact on the environment. It also extends the product lifespan and thus raises its status.

During this work I learned that designing a keyboard requires a surprising amount of work. The mechanical implementation was the most demanding and the most important part at this point, but the electronics and the software needed careful planning and studying as well. The structure is a result of logical reasoning with lots of trial and error with various schematics. There are not many viable variations for a keyboard with mechanical keys and the same functionality. I am confident that if it is to be redesigned anew, the result will be very similar. Therefore I find it strange that this has not been tried before. Likewise, a distinct or comprehensive Arduino keyboard software has not been made, maybe due to the difficulty in obtaining the keys. Especially the 3D-printed keys can result in a truly professional grade keyboard. There is still work to be done to complete the instrument, but the groundwork is comprehensive and it is easy to proceed. I warmly welcome everyone to join this project.

REFERENCES

Peltokoski, J. 2014. Background for Designing an Improved Digital Musical Keyboard Instrument. Lappeenranta University of Technology. Bachelor's Thesis. 37 p.

The MIDI Association. 2018. Specs: The MIDI 1.0 Specification: The Complete MIDI 1.0 Detailed Specification. Referenced 20.09.2018. Available: https://www.midi.org .

The MIDI Association. 2018. Specs: MIDI Reference Tables: Summary of MIDI Messages. Referenced 20.09.2018. Available: https://www.midi.org .

The MIDI Association. 2018. Specs: MIDI Reference Tables: Control Change Messages (Data Bytes). Referenced 20.09.2018. Available: https://www.midi.org .

Sparkfun electronics. 2015. Search: Tutorials: MIDI Tutorial. Referenced 20.09.2018. Available: https://www.sparkfun.com .

MusicScienceGuy. 2009. The Axis-49 Revealed. Referenced 20.09.2018. Available: http://www.musicscienceguy.com/2009/03/the-axis-49-revealed.html .

Yamaha Corporation. 2015. FAQs: Search Result: Clavinova Ensembles: Key Width and Octave Distance of Keyboards with Full Size Keys. Referenced 20.09.2018. Available: http://faq.yamaha.com/us/en/article/musical-instruments/keyboards/digitalpianos/cvp_series/198/8717/ .

DASKIN Manufacturing, Inc. 2014. The Janko Layout: Frequently Asked Questions. Referenced 20.09.2018. Available: http://daskin.com .

Chromatic Music Lab. 2018. What's Chromatone: Spec. Referenced 20.09.2018. Available: https://chromatone.jp .