Snyderphonics Manta Controller, a Novel USB Touch-Controller

Jeff Snyder
Princeton University
310 Woolworth Center
Princeton, NJ 08544
jeff@scattershot.org

ABSTRACT
The Snyderphonics Manta controller is a USB touch controller for music and video. It features 48 capacitive touch sensors, arranged in a hexagonal grid, with bi-color LEDs that are programmable from the computer. The sensors send continuous data proportional to surface area touched, and a velocity-detection algorithm has been implemented to estimate attack velocity based on this touch data. In addition to these hexagonal sensors, the Manta has two high-dimension touch sliders (giving 12-bit values), and four assignable function buttons. In this paper, I outline the features of the controller, the available methods for communicating between the device and a computer, and some current uses for the controller.

Keywords
Snyderphonics**,** Manta, controller, USB, capacitive, touch, sensor, decoupled LED, hexagon, grid, touch slider, HID, portable, wood, live music, live video

1. INTRODUCTION
In early 2008, I invented the Manta to solve my own performance dilemmas, and to give myself a more expressive interface with my computer audio software. I soon found, however, that other composers and performers were interested in the capabilities of the device, so I redesigned the instrument to be possible to manufacture in small quantities and released it as a commercial product in May 2009. There are, as of my current writing, around 130 Manta users worldwide, and many of them use the controller for purposes I never originally imagined or intended.

2. DESIGN FEATURES
2.1 Sensor Layout
The most salient feature of the Manta is the hexagonal sensor lattice. There are six rows of eight sensors each, totaling 48 hexagonal sensors. Each sensor is capable of sending information about how much surface area is covered by the performer's finger, independent of the other sensors. If you use the sensors as each triggering separate "notes", then this can be seen as an implementation of polyphonic aftertouch. The data sent by each sensor is slightly less than 8 bits, since digital conversion is performed at 8-bit resolution, which is reduced by a built-in headroom[1]. In addition to the hexagonal sensors, there are two touch sliders that send centroid data at 12-bit resolution, and four assignable function buttons, which are the same technology as the hexagons but are visually distinct from them on the layout to simplify their use as user-defined special-purpose buttons. The sensor layout is fixed, but the data the sensors send is general and could be used for any purpose the user wishes. One could argue a significant disadvantage to a fixed sensor layout in an age when the interface trend is toward infinitely flexible touch-screen interfaces, but there are also several advantages to a fixed sensor layout in a musical instrument. For one, an unchanging physical distance between the sensors encourages the development of muscle memory on the instrument. Also, it allows for the use of subtle tactile feedback, which would be harder to implement on an interface like a touchscreen.

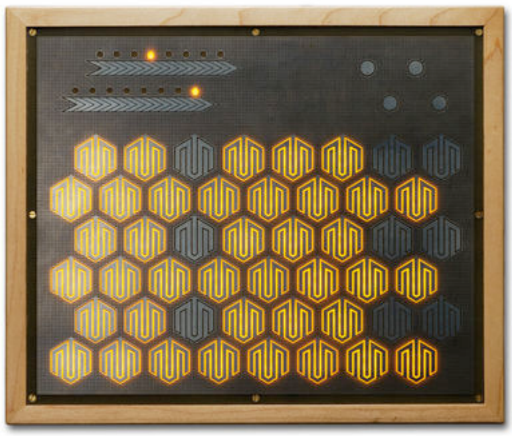

Figure 1: The Snyderphonics Manta controller

2.2 The Hexagonal Lattice
The main inspiration for the use of the hexagonal lattice pattern was an interest in the theoretical work of Ervin Wilson, whose microtonal keyboard designs are in turn inspired by the regularized keyboard designs of the 19th century, like those of Paul von Jankó or Robert Bosanquet[1][3], However, the limited number of available "keys" on the Manta, when compared to a design such as Bosanquet's, reduces the possibilities for redundant unison notes in a pitch layout, removing some of the advantages of Bosanquet's layout, such as unbroken identical scale "shapes" regardless of key center. Similar developments that focus on button instruments like the

1 This headroom is necessary to compensate for sensor drift (due to temperature and EM noise), which is monitored and compensated for by the firmware.

accordion have adapted the Jankó/Bosanquet idea, but adapted it to be more useful with a limited number of buttons, by removing the vertical unisons and replacing them with octaves. For instance, the Wicki-Hayden system[2] implements this change, while also rearranging the pitch assignments between rows to put fourths and fifths nearby, as opposed to the Jankó design, which places semitones in this position. Therefore, if one intends to use the Manta as a keyboard interface with pitches assigned to the hexagons, accordion layouts are more appropriate to the limited number of sensors on a Manta, and the 48-sensor arrangement allows for a simple implementation of the layout shown in Wicki's patent[3], with the omission of three repeated pitches[4]. Nevertheless, any Wicki-Hayden keyboard cannot achieve the advantage of truly transposable scale shapes, avoiding so called edge-effects, without at least 100 sensors[5], so one of the primary advantages of regularized keyboards is compromised for the sake of size and portability. Since my intentions for the instrument were not necessarily to always use the sensors as "keys" assigned to particular pitches, this compromise was a design choice and involved a tradeoff between sensor size, cost of sensing components, and overall device footprint.

Considered as a more general concept, the hexagonal grid affords the user three degrees of close relationships between directly adjacent sensors, which are, in the case of the Manta, horizontal and the two diagonals. These adjacencies can be inspiring for avoiding more standard rectangular grid control mappings. There are, of course, several similar applications of the hexagonal lattice to music controllers, such as those designed by C-Thru Music[6], Thumtronics[7], Starr Labs[8], Cortex Design[9], and Opal[10], and all of these are building upon either the Jankó layout or some variant of the Euler/Reimann Tonnetz[11]. However, to my knowledge, the Manta is the only commercial touch controller that combines this type of layout with the benefits of capacitive touch sensing.

2.3 LED feedback

Each hexagon and function button can be backlit in either red or amber, and this functionality can be computer-controlled. By default, the amber LED behind a sensor turns on when the sensor is touched, but this direct coupling can be deactivated with a command from the computer, after which the LEDs are completely under computer control. This functionality opens up several possibilities for more complex or context-specific visual feedback, and its primary inspiration was the Monome controller, designed by Brian Crabtree and Kelli Cain[12].

2.4 Capacitive Touch Sensing

Capacitive touch sensing is the basic sensing apparatus for the Manta. I was particularly inspired by the 100-series touch controllers designed by Donald Buchla[13], which I had the opportunity to use while studying for my doctorate at the Columbia University Computer Music Center. While these touch controllers were not polyphonic in the same way the Manta is, I found the control they afforded to the user to be very satisfying, and I decided that a surface-area-based capacitive sensing design would give me the expressive capabilities I wanted as a performer. Capacitive sensing in electronic musical instruments goes back at least to the Theremin, and has been used in musical instruments throughout the 20th Century. Examples of effective musical instruments using this technology include the Trautonium by Friedrich Trautwein, the left-hand controls of the Electronic Sackbut by Hugh Le Caine, the Multiply-Touch-Sensitive keyboard by Bob Moog and Thomas Rhea, the Wasp by EDP, the Synthi-AKS by EMS, and the Sal-Mar Construction by Sal Martirano, and the EVI and EWI by Nyle Steiner.[14] The technology has seen a recent resurgence in other markets due to the current trend of capacitive touch screens and buttons implemented by portable devices like the iPod and iPhone[15]. I find it to be a very useful sensing method for musical purposes, although I think the typical approach for modern devices of placing a glass or plastic sheet above the sensors reduces the tactile feedback to the point where the usability of the approach is significantly diminished. I designed the Manta to give the users direct contact with the metal traces of the sensors, which are etched onto a circuitboard laminate so that there is some amount of tactile feedback. This also makes sliding, glissando gestures more satisfying to the user, since the friction of the surface is less than that of glass or most plastics, such as that used on the iPad.

2.5 Velocity Detection

Early in the development of the Manta, I found that while the continuous sensor data the Manta outputs was very inspiring and suggested many expressive uses, I was also often interested in getting standard note-on and note-off data, with velocity. This is the information usually conveyed in the keyboard controller paradigm. On a standard electronic music keyboard, when you press a key, the keyboard reports which key you pressed, and how fast that key went down. This value is called velocity, and is usually mapped to amplitude of the resulting sound. When you release the key, the information about which key was released is sent. Note-on and note-off are trivial to implement in a capacitive touch-sensing system by simply determining a threshold of capacitance measurement and reporting when that threshold is crossed, possibly with some hysteresis and de-bouncing. However, "velocity" data is much harder to determine on an interface with no moving parts. It is impossible to measure the time it takes for a key to go down if the key doesn't move.

After much experimentation and collaboration with Angie Hugeback, a statistician and postdoctoral researcher at the University of Washington, I found a technique that produces a reasonably reliable velocity data based on information gathered from two successive samples above the "on" threshold. The technique involved training on example data, and using machine learning to generate an algorithm that could be applied

[2] The Wicki-Hayden system is so named because it was originally discovered by Kaspar Wicki and patented in 1896, and later independently discovered by Brian Hayden and patented in 1986

[3] Swiss patent Nr. 13329

[4] The Wicki patent diagram is for a 51-note bandoneon.

[5] This limitation is discussed in a paper on the Wicki-Hayden keyboard by Robert Gaskins at http://www.concertina.com/gaskins/wicki/index.htm

[6] http://www.c-thru-music.com/

[7] http://www.thummer.com/

[8] http://www.starrlabs.com/

[9] http://www.cortex-design.com/projects_terp1.htm

[10] http://www.theshapeofmusic.com/

[11] http://en.wikipedia.org/wiki/Tonnetz

[12] http://monome.org/

[13] http://www.buchla.com/

[14] A good overview of many of these instruments is available at http://web.media.mit.edu/~joep/SpectrumWeb/SpectrumX.html

[15] http://www.apple.com/

to the continuous data stream. Because the Manta was designed to be a commercial product, and cost is therefore a factor, the design I chose limits me to a minimum scan rate of around 6-8ms for each sensor.[16] This means that by the time two successive samples above the threshold have been collected, the elapsed time is just past the threshold of human latency perception. This meant that two samples is all I can use without the velocity algorithm feeling too slow, and the 6-8ms scan rate also guarantees that most of the sensor information happening within the very fast action of the initial key touch has been lost. Nevertheless, we were able to produce a surprisingly successful algorithm, which we implement in the host computer software rather than the Manta hardware to avoid the additional computation time it would add to the MCU loop. The addition of this velocity-detection algorithm allows the host computer to output both traditional note-on/note-off with velocity and the more unusual polyphonic continuous data simultaneously; the user can choose which data to use and which data to ignore, or combine the two data streams for a note-on with polyphonic aftertouch effect.

3. MANTA COMMUNICATION

There are currently three ways to communicate with the Manta on a computer – the Manta Max object, MantaCocoa, and libManta. I am in the process of developing the MantaMate, a dedicated hardware device that will communicate with the Manta without the use of a multimedia computer.

3.1 The Manta Max Object

The Manta is a USB controller, with a built in mini-B female connector. Since it is mostly HID class compliant, the built-in HID drivers on the Windows and Mac OS X operating systems correctly identify it[17]. However, because it is a vendor-specific HID device, another layer of software is needed to present the data to other programs that may want to use it, such as Max/MSP or Abelton Live. There are currently three ways to access the data from the Manta, and to send data to the Manta. The first way is via the [manta] Max/MSP object development by Brad Garton and myself. The [hi] object was not usable for this purpose since it does not include the ability to send output reports (which are needed to control the LEDs), so we developed a custom object, written in C, that pulls the data from the HID driver and presents it to Max/MSP, and also allows for Max/MSP users to send data to the Manta. Additionally, it makes it possible to route the data from the Manta to other applications using the Max/MSP free runtime environment and a patch that sends the Manta data over MIDI or OSC. The Manta Max object is available from the Snyderphonics website[18]. Damon Holzborn has added to this work by creating a Max For Live patch that utilizes the [manta] object and easily interfaces the Manta with Abelton Live.

3.2 MantaCocoa, The Manta OSC router

Jan Trützschler von Falkenstein has written a standalone program in Cocoa for Mac OS X that presents the data from the Manta as OSC messages, and receives OSC messages to control the LEDs and various operation modes of the Manta. This program makes the use of the Manta easier for Mac users who don't wish to use Max/MSP. MantaCocoa is especially popular among SuperCollider users who perform with the Manta. MantaCocoa is available from Jan Trützschler's website[19].

3.3 libManta

Spencer Russell has released a beta version of libManta, a C++ library to present a simple, consistent, cross-platform API to those programming applications for the Manta. It is based on libUSB, and takes care of the asynchronous polling of the USB driver and the formatting of the bit packets that the Manta understands.

libManta is available at http://gitorious.org/libmanta and is released under the GNU Public License. Spencer has also been working on a Flext object for the manta, which works on PD as well as Max/MSP, and includes Linux support, as well as an open-source cross-platform OSC router for the Manta. Christopher Jacoby has recently joined the development team and is nearing a beta release of a standalone Manta MIDI router for Mac and Windows built on the libManta library.

3.4 The MantaMate

I am currently working on the hardware for a new device that will allow the Manta to more easily interface with voltage-controlled analog synthesizers. It is basically an embedded USB host, with four 16-bit DACs, and eight 12-bit DACs. I call the device the MantaMate, and it is currently in the prototyping stages. It is conceived with the goals of enabling 4-note polyphony for a wide range of voltage controlled synthesizers, communicate with the Manta without the use of a computer, have sufficient accuracy and resolution for the implementation of unusual tuning systems on analog synthesizers, and support both OSC over Ethernet and MIDI.

I intend to release the MantaMate as a commercial product once the prototype has been fully developed and tested. I conceive of the MantaMate as not just an interface for the Manta, but also as a general-purpose format converter for musical communication – allowing the conversion between OSC, USB-HID, USB-MIDI, MIDI, and CV standards. It is not, however, an embedded computer, as it is not designed to be able to run an operating system[20].

4. USES FOR THE MANTA

Manta users have found several ways to put the capabilities of the controller to use. I'll outline a few of them here.

4.1 Microtonal Keyboard

In my own music, I have primarily used the Manta as a microtonal keyboard. I have developed what I consider the

[16] I wanted to avoid any solution that uses the audio interface for the computer to handle the data stream from the device, and I also wanted to avoid bogging down the host computer with 54 dimensions of fast data (the 48 hexagons, the 4 function buttons, and the 2 sliders). This rules out approaches like those implemented in David Wessell's 2-D touch design [5], or Madrona Labs device [2]. However, I believe in the final design by Madrona Labs, they have integrated the DSP into the hardware. When designing the Manta, I chose not to use a dedicated DSP chip, so my scan possibilities are more limited. The 6-8ms latency includes processor overhead and USB transfer rate.

[17] When programming the original Manta firmware, I used a 16-byte USB output report. This is actually outside the HID spec (4-byte maximum is specified), but I didn't notice the problem because both Mac OS X and Windows ignored the issue. Further testing shows that Linux complains, but the issue is easily sidestepped by using libUSB instead of libHID, accessing the raw USB reports. Spencer Russell implemented this workaround.

[18] http://www.snyderphonics.com

[19] http://falkenst.com/

[20] It is based on at Atmel AVR32 series of MCUs, not an ARM architecture. Therefore, it's not possible to, for instance, run PD patches or Chuck programs on it.

"concert version" of the instrument, which consists of two Mantas side-by-side on top of a custom-built wooden resonator. My standard software patch considers each hexagonal sensor a separate note, and maps the continuous data from that sensor as the amplitude of that note. This allows the performer to fade in each note of a chord independently, and control the relative volumes of the pitches with careful precision. It also allows for a very expressive tremolo effect. Putting the two Mantas side-by-side achieves a 96-note playing surface, without sacrificing the portability much, since they stack up to around 0.7" thick and roughly the length and width of a 15-inch laptop during transport. The wooden resonator has an electromagnet attached to a spruce "top", which is driven by an amplified signal from a computer running a Max/MSP patch. This gives the "concert Manta" a characteristic sound by acoustically filtering its digitally synthesized voice. Usually, I write music to be played on the Manta by other people, combined with an ensemble of other instruments I have designed. I see it as the keyboard family in my invented orchestra. The hexagonal lattice helps to avoid the equal-tempered expectations performers have about standard piano keyboards. I describe the microtonal system I use on the Manta in detail in my doctoral dissertation, Exploration of an Adaptable Just Intonation System[4].

Other users, such as composer Stephen James Taylor[21], also find the Manta appropriate for this purpose. It's also, of course usable as a controller for standard 12-tone equal temperament, in which case the unusual keypad layout can serve to avoid stereotyped keyboard habits.

4.2 Interface for Live Processing

Sam Pluta[22], an NYC-based composer and electronics performer, was an early adopter of the Manta, and was extremely helpful in early beta development of the hardware. He performs live, improvised electronic music on the Manta. He wrote his own custom software in SuperCollider, which allows him to record, manipulate, and process audio coming into his computer from a microphone. He usually performs in combination with acoustic players, such as the trumpet player Peter Evans, grabbing, stretching, distorting, and otherwise transforming their performances into strange and otherworldly textures[23]. He often treats each sensor as a control for a particular function, sometime using the continuous data, sometimes the velocity data, and sometimes just the simple on/off data. He finds the continuous values from the sensors to be extremely useful to, in his words, "really get your fingers on the data in your computer"[24]. As a gigging musician in NYC, where one generally needs to get to a gig via subway, he finds the compactness and portability of the manta to be especially suited to his needs.

Other users have also applied the Manta to a similar live-processing purpose, including Christopher Jon, the keyboardist and synthesist for the band Android Lust[25], who uses the Manta live to process the voice of the lead singer. He uses the centroid-detection mode built into the [manta] Max/MSP object[26] to control DSP effects applied to the singer's microphone input by sliding his hand around the hexagon grid.

4.3 Unusual Uses for the Manta

I have a band with the composer Victor Adan[27], in which we each use a Manta to control old pen-plotters we have bought on E-bay. The band is called the Draftmasters[28], and we perform music by sending commands to the plotters and amplifying the motors that move the pen with electromagnetic pickups. Each piece by the Draftmasters uses the hexagonal layout differently – in some cases many of the sensors are X/Y coordinates for the pen to move to, in others they send different pen speed commands to change the frequencies the motors generate. We collect the Manta data in Pd, send it to a python script, and then output it to the plotters as serial commands in HPGL.

Dan Iglesia[29], a composer and video artist living and working in NYC, uses the Manta to control live 3D video, generated with OpenGL in realtime. He wrote the custom software he uses in Jitter and simultaneously controls the audio and the video from the Manta controller live.

5. Future Development

In the future, I hope to finish the MantaMate hardware to make interaction with both newer and older analog synthesizers more simple, as well as the control of MIDI hardware and networked devices. Also, I believe that Spencer Russell's libManta, when fully released, will make development of host computer software for the Manta much easier. I consider the Manta hardware itself to be stable and unlikely to undergo significant changes, or at least I aim to make any future changes backward-compatible. Further development is mostly focused on creating software that makes the Manta easier to use and more robust for software applications.

6. ACKNOWLEDGMENTS

My thanks to Brad Garton, Sam Pluta, Jan Trützschler von Falkenstein, Spencer Russell, Angie Hugeback, Christopher Jacoby, and Damon Holzborn.

7. REFERENCES

[1] Bosanquet, R.H. *An Elementary Treatise on Musical Intervals and Temperament.* Diapason Press, London, 1876.

[2] Jones, R., Driessen, P., Schloss, A., Tzanetakis, G., A Force-Sensitive Surface for Intimate Control. In *Proceedings of New Interfaces for Musical Expression* (NIME). 2009.

[3] Keislar, D. History and Principles of Microtonal Keyboards. *Computer Music Journal* 11, 1 (Spring 1987), 18-28.

[4] Snyder, J. *Exploration of an Adaptable Just Intonation System*. D.M.A. Thesis, Columbia University, New York City, NY, 2011.

[5] Wessel, D., Avizienis, R., Freed, A. and Wright, M., A Force Sensitive Multi-touch Array Supporting Multiple 2-D Musical Control Structures. *Proceedings of New Interfaces for Musical Expression* (NIME), 2007, 41-45.

21 http://www.stephenjamestaylor.com/

22 http://www.sampluta.com/

23 You can hear some of these improvisations on the 2011 recording "Sum and Difference", from Carrier Records. http://carrierrecords.com/

24 From personal correspondence with Sam Pluta

25 http://www.androidlust.com/

26 The centroid detection mode was developed by R. Luke Dubois, and finds a centroid when a large area of the hexagonal grid is covered by the hand, such as when the user applies two or three fingers in close proximity. The discrete layout of the sensors makes this sensing method much less accurate than a similar function on a touch-screen, but it is still musically useable as a low-resolution x-y controller.

27 http://www.victoradan.net/

28 http://vimeo.com/4611451

29 http://music.columbia.edu/~daniglesia/