Pd Controlled Interface for Latin-America[1]

Jose Ignacio Lopez Ramirez-Gaston
University of California, San Diego
Department of Music
Center for Research on Computing and the Arts
nacho@discosinvisibles.org

ABSTRACT

This paper presents and interface developed with the hope of generating alternative gestural and performative techniques useful to the specific cultural environment of Lima, Peru. It also uses for its construction a post-colonial approaches in the sense that it hopes its relevance can be primarily defined by users in their own context. It is a work in progress that calls for a language that represents the unique needs of a culture-region and the realistic possibilities of technological application for that particular setting. The possibilities of human computer interaction depend on our definition and careful consideration of the 'human' participating in the process.

Keywords

Pure data, Interactivity, gesture

1.INTRODUCTION

The interface we use as an example on this work consists of a foldable circular structure that when opened surrounds the singer at waist level and contains three T-sensors in different positions so the performer can control predefined parameters with his/her hands. It also contains two piezoelectric-transducers used as on/off switches for different parameters, which can be defined and rearranged for each performance.

Two versions of the interface have already been tested. One made out of wood and a second one made out of cardboard. The cardboard version seems to be a better option for international traveling or local public transportation. In April of 2008 I traveled with the cardboard version from California to Peru which proved to be a relatively easy task. Figure 2 shows the setting up of the interface in Lima and Figure 3 shows the testing of the cardboard interface.

The setting up takes longer than for the wood version since the circuits have to be saved independently from the 'table' and are not attached to it and only placed on top of it, while in the wood version of the interface everything is already set up internally and ready for performance as soon as it is unfolded. Further testing will define the best option by comparing portability, weight, durability and resistance.

[1] Parts of this paper have been published in a Master Thesis at UCSD. See Reference.

Figure 1: Version of the interface made out of wood

Figure 2: Setting up the cardboard version of the interface

2.INTERFACE DETAILS

The interface is divided in three areas. Each one of these areas can include different sensors for the capturing of information. The version tested that we will discuss here uses three T-sensors/Theremins and two piezoelectric transducers for control. The information received by the sensors is routed to a Pure Data patch. The following diagram can give us a basic idea of how the information is routed to the computer.

The three T-sensors used are connected to a switcher that works together with a PIC controller to send MIDI information through a MIDI interface to a computer with Pure Data.

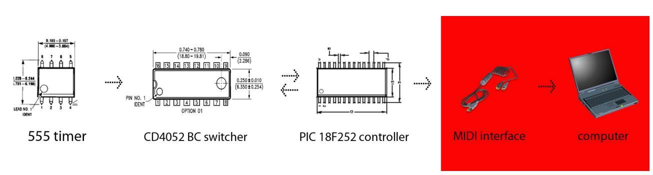

Figure 3: routing of information

2.1T-sensors

Movement information is received by Pure Data through standard Theremin or T-sensor circuits. The classic Theremin, invented by Russian scientist Leon Theremin in 1919, is a musical instrument based in the heterodyne principle to produce sound. The heterodyne principle is based on the simple product-to-sum trigonometric identity.

By mixing and multiplying two oscillating waveforms we can generate two new frequencies according to the properties of a sine function, one based on the sum of the frequencies used and the other one on the difference. While setting one of the resulting frequencies outside the audible range the other one can be controlled to produce pitch variations.

While using the heterodyne principle for a Theremin could be the most accurate and reliable way to gain control over the frequencies used for the instrument, it requires two oscillators per Theremin. A simpler Theremin can be constructed by using only one oscillator. This selection is also important for the simplicity of its fabrication and the accessibility of the materials.

In the case of this interface I use three Theremin/T-sensors connected to a switcher and a PIC that transfers the information by MIDI to a computer where it is received by Pure Data.

The Sensors are a variation of the one used by Terry Fritz on his Theremin Vision – II, an essential source of information[2]. While the Theremin Vision uses the heterodyne principle by relating two different circuits, one used as a sensor oscillator and the other one as a reference oscillator; in this particular interface I use each electric field sensor independently. An example of contemporary use of electric field sensors like the ones on this interface is the Multimodal Music Stand developed at the University of California, Santa Barbara[3].

2.2Oscillator/555 timer

The 555 timer IC (Integrated Circuit) was the first commercial timer IC and is still used for several basic sound projects by hobbyists of home built electronics because it is cheap, stable and very user-friendly.

The interface Theremin is implemented with a 555 timer in 'astable' mode as a square wave oscillator. The period of the oscillation is determined by the charging time of an external capacitor. As the capacitance increases the period of oscillation increases and therefore the frequency decreases. The T-sensor uses a fixed 22pF external capacitor to set a base oscillation frequency of around 100 kHz. The antenna and the body of the performer form the two plates of a second capacitor. The second capacitor is parallel with the 22pF capacitor and therefore the total capacitance is found by adding their respective values. By varying the distance between the performers hand and the antenna the capacitance of the second capacitor is changed. This results in varying the overall external capacitance of the timer circuit which in turn produces a change in the frequency of the output signal.

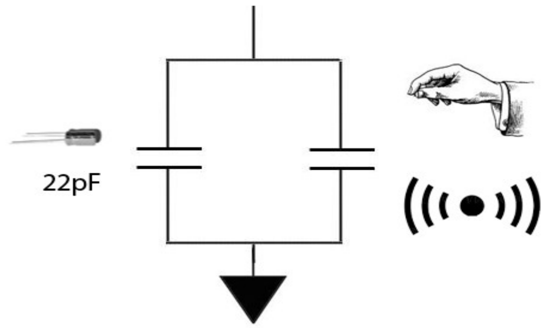

Figure 4: Capacitance diagram

Any number of T-sensors can be used to control as many parameters as wanted. In the case of this interface we use three, with the possibility of adding a fourth one according to the specifications of the switcher we explain next.

The sensor antennas, visible on the cardboard version, are made out of loops of flexible galvanized wire of 5 x 5 inches each. This set up provides a range of about 6 inches, measured perpendicular to the surface of the antenna. The sides are less receptive which helps avoid interference from the body of the performer. The sensing range has been calibrated according to the distance between sensors and the physical comfort of a performer-singer. This range can be lowered, if necessary, by adjusting a 100k trimpot on the sensor.

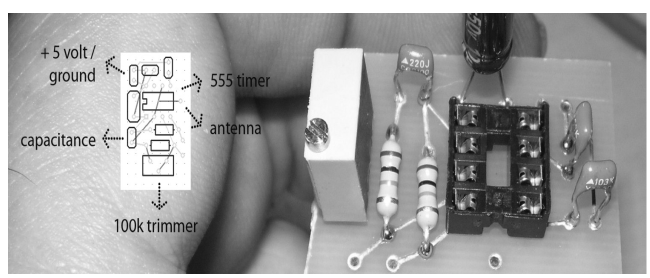

Figure 5: Circuit for the T-Sensor

2.3Switcher/CD4052 BC Multiplexer

The output of each T-sensor is a variable frequency square wave. In order to be transmitted to the host computer this signal must be converted into a numeric MIDI value. The PIC microprocessor is responsible for this conversion and the MIDI communication with the host. The PIC uses an internal counter circuit to count the number of on-off square wave transitions during 2 millisecond time periods to determine the frequency of a T-sensor square wave. It then transmits the counter value as MIDI pitch-bend values to the host computer.

The PIC has only one internal counter capable of performing the frequency to MIDI conversion. To accommodate three T-sensors a three-to-one routing switch (CD4052BC Dual 4-channel Analog

2 http://thereminvision.com/

3 http://www.create.ucsb.edu/~dano/MMMS/

Multiplexer/Demultiplexer) is placed between the T-sensors and the PIC. In this arrangement the output of each T-sensor is connected to the input channel of the switch. The output of the switch is then connected to the counter input pin on the PIC. The PIC is also connected to the switch control pins allowing it to sequentially select one of the three input channels every 2 milliseconds.

2.4Counter and MIDI transmission/PIC 18F252 Controller

The interface uses a 18F252 Controller to communicate the information from the T-sensors to the computer as MIDI data. The C++ code used to program the controller was written by Kevin Larke for this interface. As we mentioned before the PIC sends the information to switch channels every 2ms to the switcher using pins 23 and 24 sending all possible combinations of a ground/voltage pair (in this case three for the three sensors), each combination corresponding to a sensor input as explain in the following chart, were A and B represent the input pins on the switcher and their values 0 and 1, ground and 5 volts respectively. The instructions for channel switching can be found on lines 445 to 455 of the code.

A	B	Channel
0	0	sensor 0
0	1	sensor 1
1	0	sensor 2

The PIC program is designed to simultaneously perform the frequency to numeric conversion on one sensor while transmitting the previously converted sensor value. This design eliminates the time gaps between sensor readings which would occur if the conversion and transmission were done sequentially.

The steps taken by the controller are:

1.start a 2ms timer
2. read and store counter value
3. select next sensor
4. reset/start the counter
5. low pass filter and transmit the stored counter value
6. wait for the 2ms timer to elapse and start again

The information to be transmitted by MIDI goes through a low-pass smoothing filtering to eliminate noise. The smoothing filter works by taking the average of the current and previous three values (see lines 460 to 469 of the code). The smoothed values are transmitted as MIDI pitch bend messages. The PIC MIDI implementation is included as a C header (midiapp.h). The PIC serial port is connected to the host computer through a commercial MIDI interface. Pure Data reads directly from the MIDI port the information from the sensors as incoming pitch bend values with the [bendin] object according to the MIDI channel assigned. Once the signal is received, an initial calibration needs to be made to define the minimum and maximum distance points for the sensors and their equivalent numbers. Some basic math is necessary to convert the pitch value received into a range that starts with a zero to maintain the effect lines deactivated until the hand crosses the range of control.

2.5Piezo Transducer

To take advantage of the regular stereo audio input on a computer, one side of the stereo signal is assigned for the performer's microphone and the other to a piezoelectric transducer (from now piezo) used to turn on and off the singer's input.

The piezo is used as a contact microphone and set through the audio input for Pure Data and to a db/gain threshold. By tapping on the piezo a mute option for the microphone is activated. I do not consider this to be the more effective way to generate an on/off sensor but is definitely the most accessible in comparison with commercial touch or pressure sensors.

2.6Pure Data

Besides the common practice of using cracked versions of software for musical production, there is the possibility of free or open source software. The use of pirate copies of commercial software always open the possibility of the program failing or being penalized for copyright and intellectual property infringement. The social ethics of hacking culture as developed in the Massachusetts Institute of Technology (MIT) during the 1950s and 1960s fit perfectly the needs of Latin America. The basic principles present in Steven Levy's seminal work "Hackers: heroes of the Computer Revolution"[4] are not only conceptually important but essential for artistic development and musical production in Peru. The notions of sharing, openness, free access to computers, world improvement, the use of computers of artistic purposes, complete freedom of information, and the use of computers for personal transformation and improvement; seem not only useful universally but a momentary solution to current problems of accessing information and tools.

This is why the software platform used by the interface is Pure Data, a graphical programming language for the creation of interactive computer music and multimedia works, created by Miller Puckette at the University of California in San Diego[5]. Pure Data has opened the door to digital artists in Latin America to a world of musical creation that relies on the aesthetics and political implications of the free Open Source software culture. While at this point users of Pure Data in Peru are almost exclusively members of the academia or musically trained performers (whether they participate in popular electronic music or not), the figure is changing as popular performers start to get access to information and the culture of the sound arts.

The search for a ´legitimate´ working software platform is not merely a utilitarian quest, since "(s)oftware is not just a device with which the user interacts; it is also the generator of a space in which the user lives"[6] and it becomes obvious that the case of

4 Steven Levy, *Hackers: Heroes of the Computer Revolution* (New York: Anchor Press/Double Day), 1984.

5 Software available at http://puredata.info/

6 Terry Winograd Ed., "Introduction," in *Bringing Design to Software* (New York: ACM Press, 1996), xviii.

Peru calls for the support of free or affordable options. Only a basic knowledge of Pure Data is necessary to make the patches run. Alternative variations could and will include more sophisticated patches according to specific requirements by different authors or performers. A very basic patch could include: and [adc~] object using both regular channels of a regular computer for any configuration of microphones or piezo sensors (explained later) for a voice input to be transformed or a signal to be used as control; a collection of arrays with sound files for playback written on them with [soundfiler]; and a set of effects to be selected (ex: reverberation, delay, etc). and with their parameters controlled by the user through [bendin] objects receiving MIDI information from T-sensors.

The patches used so far for the interface include different options for live performance. The patch show here has the option of playing a composed piece from beginning to end, make variation or process parts of the composed piece during playback, play randomly samples of parts of the piece for live processing, or controlling both the playback of samples and the processing with the T-sensors. The level of live participation can be decided at the moment of the performance. This allows the interface to serve as a bridge between the largely common practice by electroacoustic and tape music performers to present their pieces with almost no participation and the obsession by popular electronic musicians to show their skill at processing live in order not to 'cheat' the audience. Traditional roles can be switched and alternative intermediate resolutions achieved. Pre-composed musical pieces can loose the stiffness of regular playback or be reconfigured completely at every performance, while a musician accustom to live improvisation can perform semi-organized variations of a mostly intuitive and random performance.

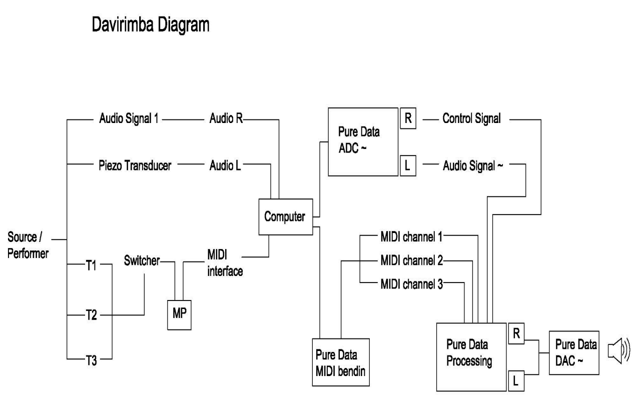

Figure 6: Audio routing for the interface

As an example, in one of the version that I have used for live performances (as shown here), the three inputs from the t-sensors are received in MIDI channels 1, 2 and 3 by Pure Data and controlling delay time and feedback gain, and reverberation feed back gain. An electronic piece is played and effects controlled during the performance by moving the hands over the sensors antennas.

3.CONCLUTIONS

This interface is not a solution but an example. It hopes to add to the conversation for the generation of alternative 'messages' useful to a specific cultural environment: Lima, Peru. It also uses for its construction a post-Colonial approach in the sense that its relevance can be primarily defined by the users in their context. This thesis is a work in progress that calls for a language that represents the unique needs of a culture-region and the realistic possibilities of technological application for that particular setting. The possibilities of human computer interaction depend on our definition and careful consideration of the ‘human’ participating in the process. The technological choices are not attached in any particular way to Peruvian culture but are used as an attempt to look at the process of adopting foreign technology for local use. This paper is intended to lead to further research into the possibilities raised when exchanging technological ideas for music making across cultural divides.

4.REFERENCES

[1] Levy, Steven, Hackers: Heroes of the Computer Revolution. New York: Anchor Press/Double Day.1984.

[2] Lopez Ramirez-Gaston, Jose Ignacio. Constructing Musical Spaces Beyond Technological Eden: A Participative Initiative for Musical Interface Development Based in the Peruvian Context. Masters Thesis, University of California, San Diego, 2008.

[3] Lopez Ramirez-Gaston, Jose Ignacio. "Cuando Canto Bajan los Cerros: An Initiative for Interface Development Informed by a Latin-american Context", Proceedings of the 2008 International Computer Music Conference (Belfast, 2008).

[4] Lopez Ramirez-Gaston, Jose Ignacio. “Bricherismo Musical: Rabietas Existenciales Sobre la Busqueda de un Lenguaje Propio”. Acido/Coma Magazine 4(0) (Mexico: Editorial Paranoia, 2008).

[5] Lopez Ramirez-Gaston, Jose Ignacio, Gomez, R., Caballero, D., and Wilder Gonzales Agreda. “Estrategias y Caracteristicas de la Musica Electrónica Limena”. Conference presented at the Sonoteca of the Centro Fundación Telefónica, Lima, Peru, April 24, 2008.

[6] Puckette, Miller. *The Theory and Technique of Electronic Music.* Singapore: World Scientific Publishing, 2007.

[7] Winograd, T. (Ed.). “Introduction,” in *Bringing Design to Software*. New York: ACM Press, 1996.