The Manipuller II: Strings within a Force Sensing Ring

Adrián Barenca
Music Technologist
Cork City, Ireland
adrianbarenca@gmail.com

Milos Corak
Software Developer
Cork City, Ireland
corakm@gmail.com

ABSTRACT

This paper presents a new prototype of the *Manipuller*, a novel gestural controller for live music performance based on strings and force sensors.

Keywords

Strings, Gestures, Manipulation, Force Sensing, Musical Interface.

1. INTRODUCTION

The *Manipuller* was conceived as a musical interface which combined strings and force sensors [7] in order to achieve higher grades of feedback and expressivity during live music performance [3]. Force sensors register the dynamic effort of the performance, transmitting effectively its expressivity [11], and strings, besides being common items of our daily life since ancient times, have intrinsic expressive qualities.

Originally inspired by the singular performance expressivity transmitted by blues and rock guitar players when bending the strings, the *Manipuller* exploits the very low frequency range of strings manipulation to produce deterministic musical outputs.

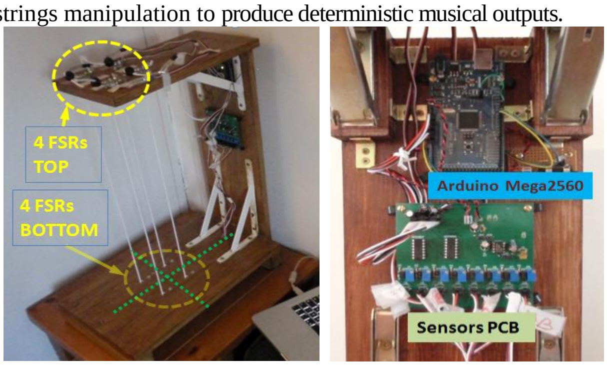

Figure 1. The Manipuller I, four parallel strings

The first prototype of the *Manipuller* (Fig. 1) disposed four parallel strings in square formation with force sensors on each string end. Hence the sensing hardware of the *Manipuller I* (MP1) consisted of four pairs of Force Sensing Resistors (FSRs) [6] placed at top and bottom of a semi-open wooden frame, a custom made sensors linearization board, and an Arduino Mega2560 development platform [1]. The sensed data is streamed in real-time to a computer running the gesture tracking algorithms and the different musical applications under MaxMSP [4].

One disadvantage in the MP1 was that the structure felt oversized for a comfortable hand manipulation, resulting too often in unnecessary gestures and excessive physical effort to operate over the full interface range. Also its body appearance was not aesthetically pleasing; perhaps rather sinister.

NIME'14, June 30-July 3, 2014, Goldsmith, University of London, UK.

The new *Manipuller II* (MP2) proposes important interface changes aimed to provide a more engaging user experience, and also the challenge of a further hardware and software integration towards a stand-alone system, able to run completely without the need of an external laptop computer.

2. RELATED INTERFACES

Some relevant examples of musical interfaces related to this research topic are *The Web* [8], *The Soundnet* [10], *The Pullka* [5], or *The Strimidilato*r [2]. Certainly both the Manipuller I and II have the strongest similarities with *The Web*, first presented by M. Waisvisz in 1989.

While the MP1 used isolated force sensing channels of independent strings, the MP2 uses a ring with a net, a configuration of interconnected strings similar to *The Web*. The difference is that in the MP2 just one hand is enough to actuate over the entire net, pulling its different string segments and registering simple or combined forces.

3. THE MANIPULLER II

The structure of the *Manipuller II* was originally inspired by the net configuration of ancient *dream catchers*. The first challenge was to adapt the technology of the existing prototype MP1 into a ring structure with an interlaced string. For the MP2 we modified the structure of a *bodhrán* (traditional Irish drum) to accommodate the strings net within the ring structure and four force sensors distributed equally around the ring.

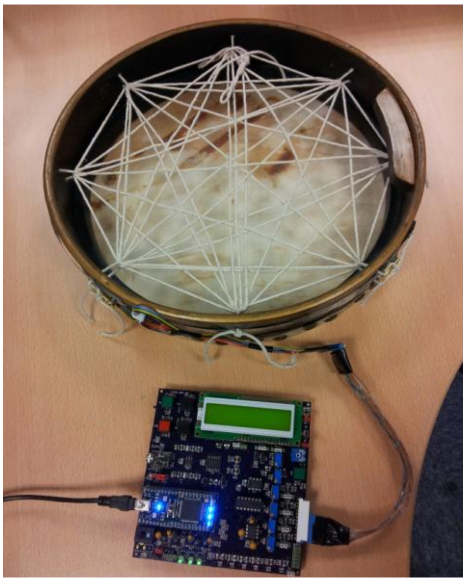

Figure 2. The Manipuller II, strings within a *bodhrán*

The complex net pattern showed in Figure 2 gives an idea of the multiple possibilities to configure the string segments within the ring. However, for better clarity, we will be using a simplified pattern to display the force sensing action lines and examples of how the user may interact over the interface.

3.1 The Ring Interface

In the MP2 the whole hand is involved in the manipulation and all fingers can play a role within the net, which improves user control during performance. In the MP1, the manipulation was more a grabbing gesture and the fingertips did not play a practical role. In essence we may say that the manipulation of single strings happen in 1D (only hand), while the action over a net occurs in 2D (hand and fingers).

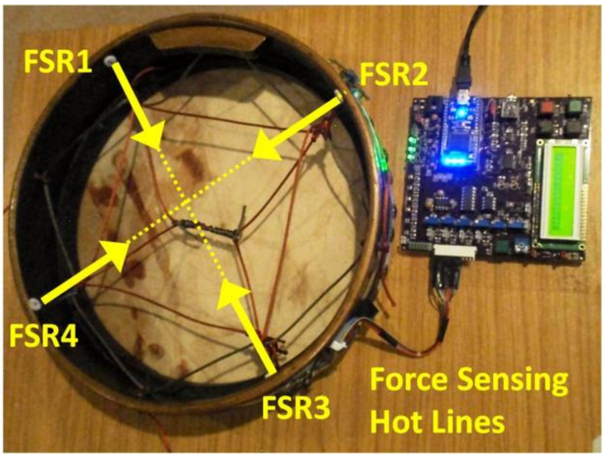

Figure 3. Force Sensing Directions

For this smaller frame we have reduced the number of FSRs sensors used from eight down to four. These sensors and their mechanical actuators are placed at the outside of the ring and connected to the strings via drilled holes. Figure 3 shows the resulting force sensing hot lines, which determine the direction of maximum pulling force sensitivity.

3.2 Hardware

The hardware has been redesigned and all electronic blocks have been integrated in one single printed circuit board, reducing drastically the wiring between sensor's and micro-controller boards. The MP2 uses a much more powerful processor than its predecessor, along with added peripherals such as an audio DAC, a µSD memory or a display.

The challenge is to make the *Manipuller II* independent of an external computer to run its own integrated musical applications. However the use of an external computer is still possible, since the system streams each sensor reading in real time via its USB-serial port. This still allows the interface customization with the use of third party software tools.

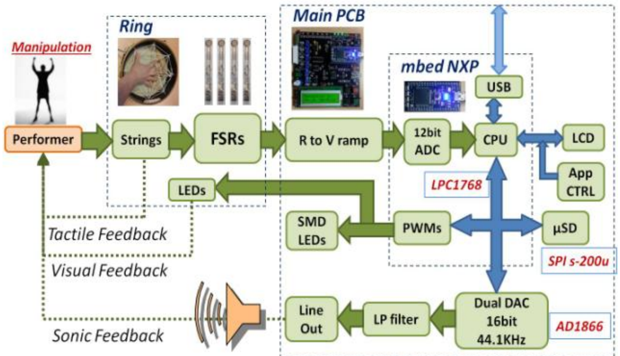

Figure 4. The Manipuller II Hardware Block Diagram

The sensors signal conditioning is carried out by the linearization circuit shown in Figure 7. This circuit configuration provides a ramp output between ground (0V) and the rail supply (3.3V), which accommodates the output voltage to the full ADC input range of the LPC1768. The gradient of the ramp (sensitivity) is adjusted with potentiometer R21 and the capacitor C6 sets the low pass cutoff frequency around 100Hz, which should be enough to register gesture variations of up to 50Hz.

There are also dedicated LEDs which provide visual feedback of the tension sensed at each sensor. For instance, a LED brightness that no longer increases with applied force indicates that the voltage output has saturated, whilst no brightness would indicate absence of external manipulation. This is very useful for adjusting the offset and sensitivity of each sensor.

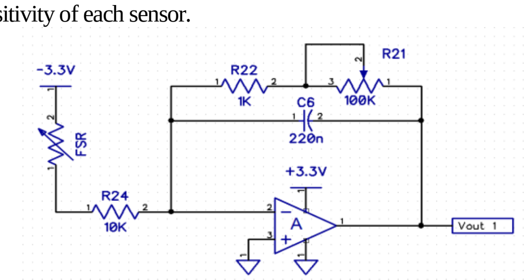

Figure 5. Single FSR linearization circuit

The force sensors chosen for MP II have a significantly smaller force range (1lbs vs. 25lbs), resulting in smaller resistance values at full scale. This allows the use of lower resistor values in the feedback loop of the linearization circuit, improving its noise immunity.

A micro-SD memory card (placed on the bottom layer) provides external memory to store the audio files required for applications of sound manipulation. Furthermore this storage media may also be used as a large audio buffer for extra computing memory to implement certain sound effects, such as reverb or delay. This memory card is accessed via high speed SPI bus permitting data transfer operations above 10MB/s.

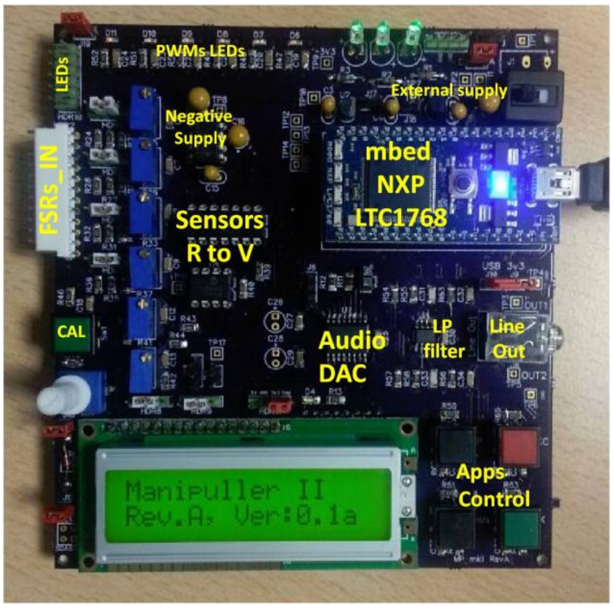

Figure 6. The MP II Board with NXP mbed LTC1768

3.3 Firmware

All the on-board hardware control is carried out by the *mbed* NXP platform [9]. It provides a powerful processor with all the necessary ADC inputs, PWM outputs, digital I/O pins and communication busses to drive all the on-board peripherals, such as the LCD display, the micro-SD memory card and the audio DAC.

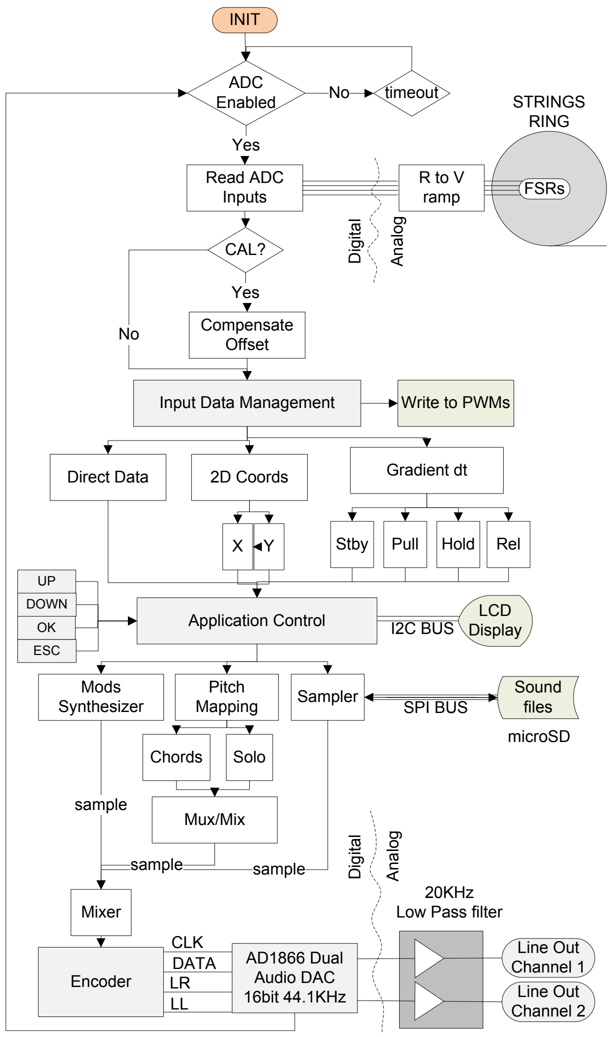

Figure 7. Main Firmware Flow Diagram

Figure 7 shows the flow diagram of the main firmware routine. After initialization, if the AD conversion is enabled, the four analog inputs corresponding to each force sensor are read and, if applicable, their standby offset is compensated. The standby offset is the residual voltage with the absence of external manipulation. The converted data is then processed to determine gesture tracking parameters such as string status or the associated spatial coordinates.

The gesture tracking parameters are fed into the application control and then routed to the corresponding musical mapping algorithm, which calculates and outputs one sample. This single sample is passed to the encoder, which produces the corresponding clock, data and latch signals that the audio DAC needs to turn the numerical sample into a voltage. To keep a minimum output sampling frequency of 44.1 KHz at the audio DAC outputs this main loop must be executed in no more than 22.68µs, although this specification is not critical at this stage and may be relaxed if necessary.

4. GESTURES TRACKING

Although the strings are interconnected it is possible to actuate over single force sensors without affecting the rest. In this section we analyze the different gestures tracking options implemented in the *Manipuller II*.

4.1 Pulling Status

It is possible to detect whether the force registered by a given sensor is pulling or releasing by calculating the gradient over time. Hence there are four different statuses for each sensor: gradient is positive, status is *pulling*; gradient is negative, status is *releasing*; gradient is zero and sensor reading is non zero, status is on *hold*; gradient is zero and the sensor reading is zero, status is *standby*.

Table 1. Comparison of MP1 vs. MP2

	MP1	MP2
Structure	Semi-Open frame	Wood Ring
Configuration	Parallel strings	Interlaced
FSRs	8	4
Dev. Board	Arduino Mega2560	mbed NXP LPC1768
CPU Speed	16MHz	96MHz
ADC Precision	10bit	12bit
Max ADC rate	10KHz	200KHz
Computer	Yes	Optional
Musical Apps	Third party only	Custom Integrated
Audio Hardware	Computer Soundcard	On-board Audio DAC

4.2 Twisting and Combined Pulling

By grabbing the central knot and twisting left or right we simultaneously actuate over opposite sensing lines. Similarly, by pulling over the square lines we may actuate over adjacent sensors. Single sensor actuation may be achieved by pulling over any of the four square corners nearest to the ring. There are many possibilities for shaping the net; the simplified pattern displayed in this paper felt effective and may be complemented with extra string segments.

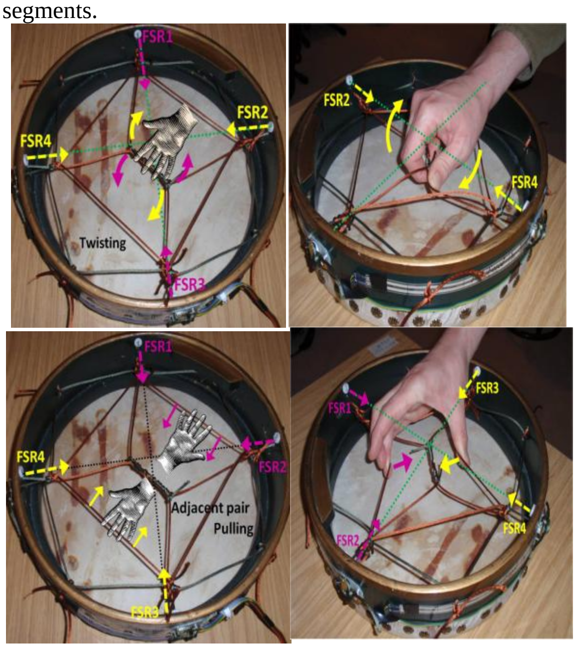

Figure 8. Twisting and Combined Pulling in the MP2

4.3 Vector Position in 2D

The original reason to implement spatial vector positioning was to carry out a gestural control of sound *spatialization*. Hence in the MP1 each of the four strings was assigned to a 3D base position vector. Coordinates (x, y) were scaled by the corresponding averaged top-bottom force sensing values to map the four quadrants of the horizontal plane, while the differential top-bottom determined the relative azimuth (z). In the MP2 the position estimation has been simplified and only the horizontal plane is taken into account.

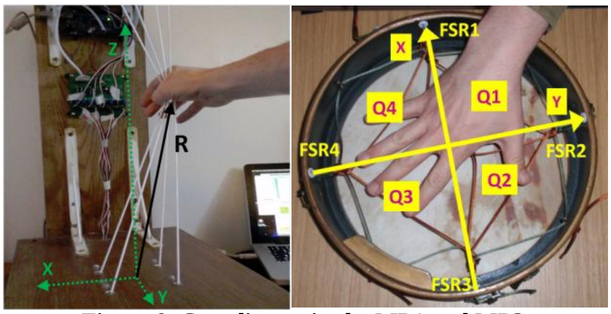

Figure 9. Coordinates in the MP1 and MP2

Table 2. Coordinates estimation in the MP2 and the MP1

MP2	(X, Y), Z=0	**MP1**	(X, Y, Z)
FSR_1	(1, 0)	String 1	(1, 1, z_1)
FSR_2	(0, 1)	String 2	(-1, 1, z_2)
FSR_3	(-1, 0)	String 3	(-1, -1, z_3)
FSR_4	(0, -1)	String 4	(1, -1, z_4)

5. MUSICAL APPLICATIONS

At this early stage it is still difficult to determine which musical applications or mapping strategies would work better in the *Manipuller II*. Hence it makes sense to use a reduced but versatile group of musical mapping strategies to evaluate this prototype: synthesis, sound manipulation, and pitch mapping.

5.1 Sound Synthesis: Modulations

To carry out a Frequency Modulation synthesis, the carrier frequency is given by FSR_1, the modulation index by (FSR_2 - FSR_4)*10, and the *harmonicity* ratio by FSR_1/FSR_3. To obtain meaningful sound FSR_1 and FSR_3 were rescaled from [0, 1] to [100, 3500] Hz. Similar type of combinations and procedures were used for other synthesis methods, such as Amplitude and Ring Modulation.

5.2 Sound Manipulation: Sampler

This application maps the string status to *sound file playback control*, where pulling the string plays the sound forward, and releasing it does it backwards. The average value registered by the force sensors control the ratio of the playback speed. On hold, the playback direction corresponds to the previous state. For instance, if the string is being released and then set on hold, the playback would continue backwards. In the practical implementation we used threshold values for the hold status condition and the minimum gradient variation. The *twisting* gesture is also used here to playback a single sound file. Similarly to a turntable, the direction and magnitude of the gesture determines the sound sample indexing sign and speed.

5.3 Polyphonic Pitch Mapping

The values of each force sensor can be mapped to portions of a given *pitch scale*. The first FSR is mapped to the lower pitches; the second to the low-mid pitches; the third to the mid-high; and the fourth to the higher pitches. Therefore, when the net is manipulated, it generates a chord of separated pitches. The strings status algorithm determines the trigger, the hold time and the release of each generated pitch.

6. DISCUSSION

So far the preliminary tests on both hardware and firmware are promising and the minimum technical specifications are being met. Perhaps there may be questions about whether the strategy of integrating on-board musical applications has true practical advantages versus a more simplified interface with USB-Serial port capabilities, which may be connected to a laptop computer running third party music software.

The three musical applications presented are simple and offer a general view of the wider range of mapping possibilities. Possibly the most successful application is sound manipulation, for the perceived sense of manipulation over an existing sound, as if we pulled the strings within its own waveform.

7. CONCLUSION

The configured strings within the ring set a qualitative improvement in the way the manipulation is transmitted in the *Manipuller II*. The result is a more responsive and meaningful gesture tracking and a more engaging user experience.

The development of the *Manipuller II* continues to date and further improvement considerations are continuously taken into account. Special emphasis is being put towards the improvement of the structure for a more reliable and simpler instrument setup.

The immediate challenge for MP2 is to successfully complete the transfer of the existing gestural control and musical applications from the laptop computer to the embedded platform. The task ahead is considerable and some applications are still in the early implementation stage. The ultimate goal is to leave behind the prototyping stages and move towards a more defined end product, which would offer a different approach to live music performance and may inspire other research works.

8. REFERENCES

[1] Arduino. Arduino Mega Board, 2007. Retrieved 1st Oct. 2013: http://arduino.cc/en/Main/ArduinoBoardMega2560.

[2] Baalman, M. A. J. The STRIMIDILATOR: a String Controlled MIDI-Instrument. In *Proceedings of the 2003 Conference on New Interfaces for Musical Expression,* (Montreal, Canada, 2003).

[3] Barenca, A. and Torre, G. The Manipuller: Strings Manipulation and Multi-Dimensional Force Sensing. In *Proceedings of the 2011 Conference on New Interfaces for Musical Expression,* May 30- June 1, (Oslo, Norway).

[4] Cycling'74. MaxMSP Graphic Programming Environment, 2009. Retrieved 10th August 2010: http://www.cycling74.com

[5] Fels, S. and Vogt, F. Tooka: Explorations of Two Person Instruments. In *Proceedings of the 2002 Conference on New Instruments for Musical Expression,* May 24-26, (Dublin, Ireland, 2002).

[6] Flexiforce. Flexiforce Sensors User Manual, 2001. Retrieved 16th October 2013: http://www.tekscan.com/pdfs/FlexiforceUserManual.pdf

[7] Hollinger, A. and Wanderley, M.M. Evaluation of Commercial Force-Sensing Resistors. In *Proceedings of the 2006 Conference on New Interfaces for Musical Expression,* (Paris, France, 2006).

[8] Krefeld, V. The Hand in the Web: An Interview with Michel Waisvisz. *Computer Music Journal,* 14 (2), 1990.

[9] Mbed. Mbed NXP LPC Handbook, 2013. Retrieved 1st December 2013: http://mbed.org/handbook/mbed-NXP-LPC1768

[10] Tanaka, A. Musical Performance Practice on Sensor-based Instruments. Faculty of Media Arts and Sciences, Chukyo University, Toyota-Shi, Japan, 2000.

[11] Wanderley, M. M., and Orio, N. Evaluation of Input Devices for Musical Expression: Borrowing Tools from HCI. *Computer Music Journal*, Vol. 26, No. 3, New Performance Interfaces (Autumn), pp. 62-76 Published by The MIT Press, 2002.