Light Pipes: A Light Controlled MIDI Instrument

SookYoung Won
CCRMA, Department of Music
Stanford University
Stanford, California USA
sywon@ccrma.stanford.edu

Humane Chan
CCRMA, Department of Music
Stanford University
Stanford, California USA
humane@ccrma.stanford.edu

Jeremy Liu
CCRMA, Department of Music
Stanford University
Stanford, California USA
jliu@ccrma.stanford.edu

ABSTRACT
In this paper, we describe a new MIDI controller, the Light Pipes. The Light Pipes are a series of pipes that respond to incident light. The paper will discuss the design of the instrument, and the prototype we built. A piece was composed for the instrument using algorithms designed in Pure Data.

Keywords
Controllers, MIDI, light sensors, Pure Data.

1. INTRODUCTION
The Light Pipes MIDI controller (Fig.1) was developed as a project for the Human Computer Interaction Theory and Practice course at CCRMA. Light is used as the trigger for the instrument. Other light based instruments in the past have included Don Buchla's infrared sensor based Lightning MIDI controller [1], [2] and the LightHarp MIDI controller [3], [4]. For the Light Pipes, sensors contained in PVC pipes were combined with an Atmel microprocessor to create a MIDI controller unit. MIDI signals were generated by the instrument and sent to a Pure Data patch that received the MIDI messages and generated sound in response.

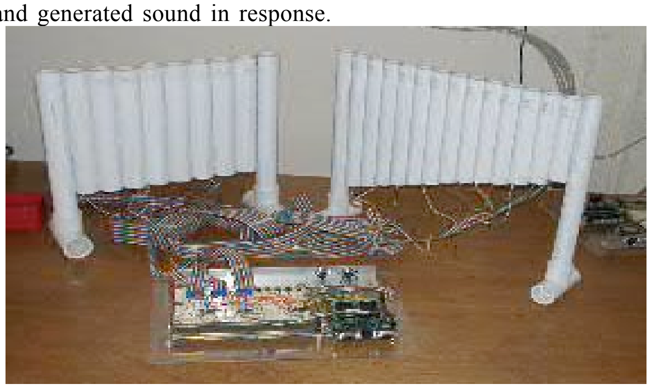

Figure 1. Light Pipes MIDI Controller

2. CONCEPT AND DESIGN
2.1 Initial Concept
The Light Pipes MIDI controller is an array of pipes that respond to light shining on the pipe ends to produce MIDI messages. Aspects of the pan pipes and marimba contributed to the visual appearance of the controller and its mode of operation. On the other hand, the instrument from its conception was intended to be a homophonic instrument, unlike either of those two instruments.

2.2 Final Design
The final design of the Light Pipes controller incorporated two arrays of pipes. One array contained 13 pipes representing a chromatic scale. This array was designated as 'key' pipes, with each pipe triggering a MIDI note message. The other array contained 8 pipes and was designated the 'subfunction' array. The purpose of these pipes is to perform other functions, such as modifying the messages sent by the 'key' pipe array and sending a variety of other MIDI messages.

2.3 Signal Flow Overview
The signal flow for the instrument is shown in Fig. 2. The light input activates the pipe sensors, sending voltages to the Atmel microprocessor. The microprocessor converts the voltages to digital values and processes those values, sending out appropriate MIDI messages in response to the incoming data. The MIDI messages are sent to a PC running Pure Data, which receives the messages and generates sound accordingly.

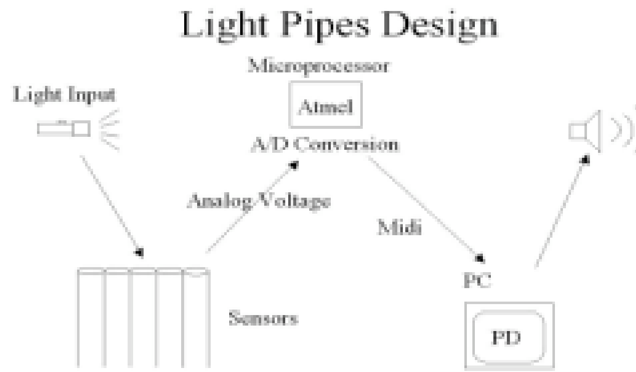

Figure 2. Light Pipes Signal Flow

3. HARDWARE AND SOFTWARE
3.1 Sensors/Structural Elements
The sensor arrays consist of photoresistors mounted inside PVC pipes. When the user shines light on the end of a pipe, it falls on the photoresistor. The resistance of the photoresistor changes in response to the incident light. The photoresistor is wired as one of the resistors in a voltage divider, so that the output voltage of the circuit changes in proportion to the incident light. The amount of light can be adjusted by using an external light source or by covering the end of the pipe.

3.2 MIDI Messages
The MIDI messages we chose were directly affected by our conceptual design. The key pipes only send note on and note off MIDI messages. The velocity is set to a default value of 64 for all these messages and the pipes have note values 60-72 in the default register. Two of the dedicated subfunction pipes, 'high' and 'low', change the register of the key pipes. The 'high' pipe shifts the key pipe values to 72-84. The 'low' pipe shifts the key pipe values to 48-60. The 'off' subfunction pipe only sends note off messages. It checks to see which notes are currently are in the 'on' state (see Section 3.3) and sends note off messages for all applicable notes. The remaining five subfunction pipes also send the same note on and note off messages as the key pipes, but they additionally send continuous values in the form of aftertouch messages on MIDI channels 1-5. The value 64 corresponds to neutral or ambient lighting for the aftertouch messages.

3.3 Microprocessor Signal Processing

The ATMEL microprocessor handles logic and signal processing tasks, in addition to A/D conversion and generating MIDI messages. Each note the key pipes represent (37 total) has a state that can be on or off. The microprocessor remembers the state of each note and changes the states in response to light input.

To make triggering reliable, we needed to calibrate the sensors. When the button on the protoboard is pressed, the ATMEL records the current values of each sensor and retains the values as neutral light levels. Potentiometers allow adjustment of thresholds, with values ranging from –512 to +512 displayed on an LCD. Each value represents a threshold equal to the neutral value of a particular sensor plus the value of the potentiometer. There are four potentiometers, representing high and low thresholds for each of the subfunction and key arrays.

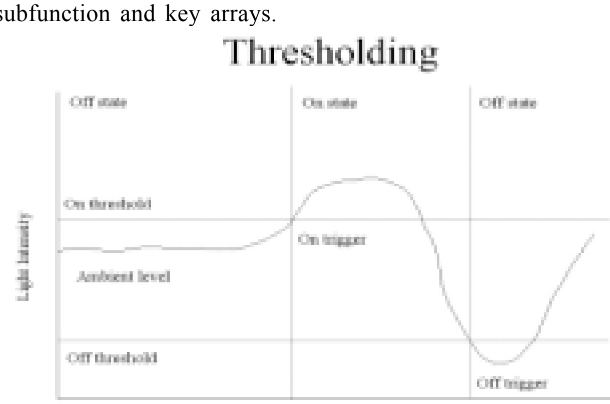

Figure 3. Thresholding

The thresholding operation is shown in Fig. 3. We begin in the off state. The ambient light level has been recorded by the microprocessor. As light intensity increases, the 'on' threshold is reached. This threshold is the value of the 'on' potentiometer at the time of calibration plus the saved ambient light level. At this point, the on state is triggered and a MIDI note on message is sent. In the on state, another note on message cannot be sent. First, the light intensity must drop below the 'off' threshold, which is the value of the 'off' potentiometer at the time of calibration plus the saved ambient light level. As we see in the figure, when the 'off' threshold is reached, we switch back to the off state and a note off message is sent. Now another note off message cannot be sent until we return to the on state.

Since we can adjust the thresholds relative to the neutral light levels, we have some flexibility in the use of our instrument. The simplest choice we have is the sensitivity of our pipes. However, since the potentiometer values are both positive and negative, we can set up our instrument in different modes. In Fig. 3, what we call the 'sustain' mode is shown. The 'on' threshold lies above neutral level, requiring additional light to turn a note on and the 'off' threshold lies below the neutral level, requiring the player to intentionally cover the pipe to shut the note off. This can be seen as a form of hysteresis, as the note is in a different state despite returning to the same light level. Another possible mode is the 'auto-off' mode. If the off threshold has a positive value and lies above the neutral light level, removing the light impetus after turning a note on will turn the note back off. A third mode that could be used is an 'inverted' mode, where darkness turns a note on and light turns a note off.

4. MUSICAL APPLICATION

Sook-Young Won composed a piece for the Light Pipes MIDI controller entitled "C-Improvisation". Patches in Pure Data were written as performance algorithms for sound generation. Some of the patches are described below. A video of the performance of this piece can be found at:
www-ccrma.stanford.edu/~sywon/courses/250a/lightpipes.avi

4.1 Travel

Travel is a sound generator based on the trill ornament. While normally two notes are played repeatedly in a trill, Travel can loop over a maximum of eight notes, following the order in which the notes are triggered. Travel remembers the last eight notes sent to it and loops when it is activated. The tempo is variable as well.

4.2 Loop 5th

The Loop 5th patch receives a MIDI note from the key pipes and plays the note a fifth up simultaneously with the base key note received. The two notes are played together repeatedly until the subfunction pipe is turned off. Tempo is again a continuous variable.

4.3 Half-tone Scale

The Half-tone scale patch plays a chromatic scale beginning from the MIDI note it receives. The scale stops when the MIDI note reaches a value greater than 100.

4.4 Random 3rd

The Random 3rd patch receives MIDI note messages and generates a random sequence of notes increasing in pitch starting from the base note. The sequence ends when it reaches a note with MIDI value greater than 110.

5. FUTURE WORK

We would like to add feedback lighting to the pipes that would indicate the state of a particular pipe. This could be done with LEDs. Another idea is outer shell for the pipe that would make the entire pipe light up. This would be visually appealing, as well as useful.

6. ACKNOWLEDGMENTS

We would like to thank CCRMA, Bill Verplank, Max Mathews, Rodrigo Segnini and Michael Gurevich for their help and contributions to this project.

7. REFERENCES

[1] Buchla, D. Lightning II MIDI controller. http://www.buchla.com, April 2004.

[2] Rich, R. Buchla Lightning MIDI controller. *Electronic Musician* 7(10), 102-108, 1991.

[3] Favilla, S. LightHarp MIDI controller. http://home.mira.net/~favilla/instruments.htm, April 2004.

[4] Favilla, S. Real-time Control of Synthesis Parameters for LightHarp MIDI Controller. In *Proceedings of the 1997 ACMA Conference,* Auckland, New Zealand, 1997.