THE PIPE: Explorations with Breath Control

Gary P. Scavone
Center for Computer Research in Music and Acoustics
Stanford University
Stanford, CA 94305
gary@ccrma.stanford.edu

ABSTRACT

The Pipe is an experimental, general purpose music input device designed and built in the form of a compact MIDI wind controller. The development of this device was motivated in part by an interest in exploring breath pressure as a control input. *The Pipe* provides a variety of common sensor types, including force sensing resistors, momentary switches, accelerometers, potentiometers, and an air pressure transducer, which allow maximum flexibility in the design of a sensor mapping scheme. *The Pipe* uses a programmable BASIC Stamp 2sx microprocessor which outputs control messages via a standard MIDI jack.

Keywords

MIDI Controller, Wind Controller, Breath Control, Human Computer Interaction.

1. INTRODUCTION

The Pipe, shown in Fig. 1, is an experimental, general purpose music input device designed and built in the form of a compact MIDI wind controller. While acoustic wind instruments, as well as most existing commercial wind controllers, make use of dynamic air flow for activation, *The Pipe* is based on a "flow-free" breath pressure paradigm. The development of this interface was in part motivated by an interest in exploring the use and effectiveness of static-flow breath pressure as a control input. In addition, *The Pipe* provides a variety of common sensor types, including force sensing resistors (FSRs), momentary switches, accelerometers, and potentiometers, which allow maximum flexibility in the design of a sensor mapping scheme. The device uses a BASIC Stamp 2sx microprocessor which can be programmed via a serial interface to a computer and outputs control messages via a standard MIDI jack.

Figure 1: *The Pipe*: View from above (top) and below (bottom).

A nearly complete version of *The Pipe* was constructed more than two years ago but then left unfinished when size and space complications arose during final assembly. The impetus to continue work on the device, which ultimately led to a complete rebuild, occurred in the course of music composition experiments with real-time physical modeling algorithms.

2. INTERFACE CONCEPT AND DESIGN

The Pipe is intended to function either as an independent MIDI controller or as an interface to computer-based synthesis algorithms. During its design, several goals were identified, including:

- the ability to use and experiment with static-flow breath pressure as a control input
- to provide a wind-like control surface for use with existing woodwind tonehole synthesis models
- to provide a more hygienic breath pressure interface for device sharing
- to allow precise variation of controller values
- to make use of as many different sensor types as could reasonably be located on or within its structure
- to be housed within a durable, compact shell

In particular, *The Pipe* is meant to operate in place of several previous experimental digital interfaces built by this author [4, 5], allow generic control of most Synthesis ToolKit in C++ (STK) instruments [2], provide an expressive interface for wind instrument performance control, as well as offer flexibility for future synthesis model control and input sensor developments.

The Pipe was conceived as a standalone, battery powered device which would provide a set of finger "keys" in a configuration akin to a musical recorder, as well as contain all necessary circuitry to process the sensor data and output MIDI-formatted control messages. A housing was fabricated from 1.5 inch (3.8 centimeter) inner diameter Acrylonitrile Butadiene Styrene (ABS) piping, cut to a length of approximately 14 inches (35.5 centimeters). A small circuit board was cut so that it could be slid in and out of *The Pipe* along a set of grooves at its downstream end. This layout is diagrammed in Fig. 2. All necessary wiring and electronics are contained within the ABS body to maintain maximum durability.

The finger keys or switches on all commercial wind controllers provide only limited, binary state information. When

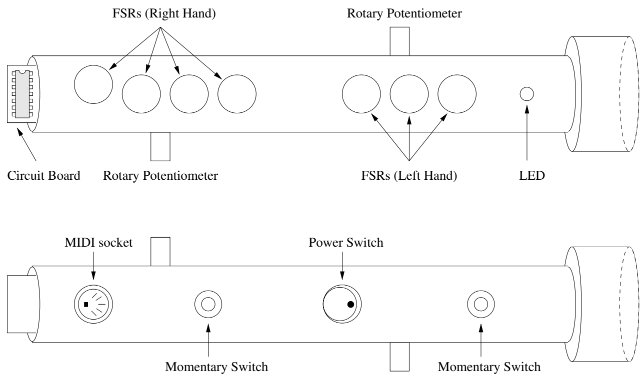

Figure 2: *The Pipe* layout diagram: View from above (top) and below (bottom).

a key is depressed or touched, a new MIDI "Note On" message is produced which corresponds to some predefined fingering/note number mapping. A previous controller by this author [4] used FSRs in conjunction with finger keys to enable a 7-bit range of finger position data. *The Pipe* was designed so that the fingers rest directly on or above FSRs. A set of seven finger depressions were drilled along the length of *The Pipe* in a traditional two-hand arrangement. Circular force sensing resistors of 3/4 inch diameter were positioned in the depressions and slots were chiseled for the FSR leads to facilitate an internal wiring system. Initially, the FSRs were covered with only a thin layer of tape and direct finger pressure was applied to them. Later, compressible, foam-like pads were added on top of the FSRs to provide some haptic feedback in the fingering mechanism.

Traditional wind instruments are driven by dynamic air flow through an acoustic air column. Most commercial digital wind instrument controllers have made use of similar breath control schemes (Yamaha WX5, Akai EWI). In an electronic wind controller, this air flow is completely unnecessary and even undesirable given potential complications involved with humid air flow through or near electronic components. In addition, the limited capacity of the human lungs requires that outgoing air flow, and the resulting pressure, be periodically stopped. Pressure sensing in *The Pipe* was therefore based on a static air flow paradigm within an air tight enclosure on the upstream end of the instrument. A removable, contoured mouth cap was designed to be positioned against the performer's face but beyond the mouth and lips to minimize hygienic concerns that might arise when sharing the device. In this scheme, an air-tight seal is formed between the player's face and the cap which allows pressure to be maintained and controlled indefinitely inside the cap while breathing normally through the nose. The mouth cap was fabricated from a short, ABS coupling section.

A variety of additional sensors are included in *The Pipe*. An earlier controller by this author [5] provided tilt sensing in two dimensions using a dual-axis accelerometer. Experience with that device indicated that tilt offers a convenient control parameter that is especially easy to use concurrently with other input sensors. In addition, accelerometers are naturally suited for sensing gestures appropriate when controlling shaker synthesis algorithms [1]. Two rotary potentiometers allow for small or subtle control value variations, as well as a means for setting values which are intended to remain fixed until further modified. These controls were added to address limitations with the use of FSRs, which are difficult to use for precise control and which require continuous action on the part of the player to maintain a non-zero value. Two momentary switches were provided for use as triggers or in conjunction with other sensors to create more complex control schemes.

Given the variety of possible applications, a programmable microprocessor interface was required. *The Pipe* was built using a Basic Stamp 2sx microprocessor by Parallax, Inc.

2.1 Hardware Details

The seven FSRs are arranged to be used by the first three fingers of the upper left hand and the four fingers of the lower right hand. A momentary switch and rotary potentiometer are located in proximity to each of the finger "banks". The system was balanced in such a way that it could be easily played with either hand independently.

The electronic components were mounted on a circuit board which was cut so that it could be slid into the downstream end of the ABS housing. The FSRs (Interlink Electronics), switches, potentiometers, MIDI socket, and LED were mounted on the ABS housing (see Fig. 2) and connected to the circuit board via ribbon cables. The BSIIsx is powered by a 9-volt battery, which is housed inside *The Pipe* just below the detachable mouthpiece.

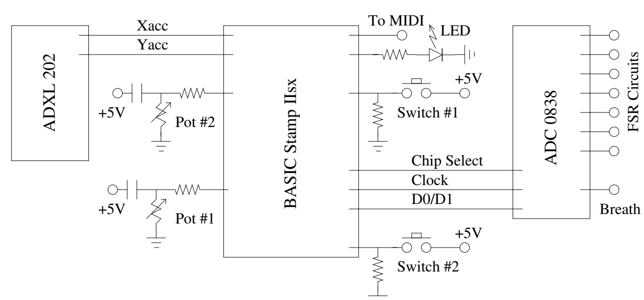

Figure 3: *The Pipe* circuit schematic.

The circuit board schematic is shown in Fig. 3. The dual-axis accelerometer, an Analog Devices ADXL202, provides digital output and interfaces to the BSIIsx via two of its sixteen input/output pins. The potentiometers and momentary switches are connected via one input pin each. The MIDI interface jack uses one output pin. The seven FSRs and the air pressure sensor are read through an 8-channel, 8-bit serial I/O multiplexing analog-to-digital converter (National Semiconductor ADC0838), which connects to the microprocessor using three pins. Currently, a Motorola MPXV5010 air pressure sensor is being used with a range of 0 to 1.45 PSI. In the future, a more sensitive device may be substituted. Finally, an LED was included via another pin as a visual feedback mechanism to distinguish program features. In its current form, five input/output pins remain for future modifications or upgrades.

A single MIDI cable is necessary to connect with an external synthesizer or computer. A removable, four-pin connector provides a serial interface to a computer for microprocessor programming. The BASIC Stamp makes use of a simplified and customized form of the BASIC programming language called PBASIC. With the BSIIsx processor, up to eight programs can be stored in 2 Kbytes of memory each. The scaling of sensor data for 7-bit MIDI message output is left completely to the programmer.

2.2 Sensor Mapping

As configured in *The Pipe*, the FSRs and breath pressure sensor produce a non-zero output only when activated by finger or breath pressure. This makes them appropriate in situations where the sensor is used to produce a control gesture which always begins and ends at the same equilibrium, non-active value. This behavior mimics the spring-loaded keys of wind instruments. However, sensors of this type are less well suited for situations where one wishes to make modifications, perhaps precise, above and below a mid-range value. The response of the FSRs in particular is highly non-linear and in the current configuration has much greater variation at low pressure values. The momentary switches function in a similar manner to the FSRs but produce only two possible states.

The rotary potentiometers, on the other hand, provide a good means for making small control value changes about a non-zero mean, as well as settings which maintain their state. The accelerometers, when measuring tilt, can be used in a similar way though the usable range in hand-held situations such as this is limited to approximately 16 or less distinct, controllable positions (approximately 4-bits of resolution).

While each sensor type presents particular constraints, it is possible to use combinations of sensors to achieve greater flexibility and augment the raw capabilities of the device. For example, if one wishes to use *The Pipe* in a situation that calls for six "slider-like" controls, the FSRs can be used as "switches" to activate particular control parameters for modification by a nearby rotary pot. Another useful combination involves the use of a momentary switch to activate modifications made using FSRs or the breath pressure sensor. In this way, a control value can be positioned using pressure and then held at a given value by releasing the momentary switch *before* releasing the pressure. Combinations using the dual-axis accelerometer are possible as well. Finally, the control interface provided by *The Pipe* has limits, particularly in situations where a large number of control values need to be modified simultaneously. When using the device to drive a real-time computer synthesis engine, such manipulations can be made into preprogrammed functions on the computer which are simply triggered by the controller.

Maximum flexibility with *The Pipe* is achieved by offering a variety of conveniently located sensors which can be controlled using one or two hands and which can be configured and programmed as desired for a given situation.

Another aspect of parameter mapping concerns the way control values are mapped to synthesis parameters [3]. A few example mapping schemes of this sort are considered in the following section.

3. MUSICAL APPLICATIONS

The Pipe was designed for a variety of specific uses, as well as made generic enough to function in as-yet unknown situations. One goal was to have an instrument which could function in place of two previous experimental controllers. Another goal was to create an instrument for use as a real-time performance controller. This section documents these applications.

3.1 A Tonehole Controller

Physical modeling research reported in 1998 led to an efficient, real-time model of woodwind instrument toneholes [6]. In conjunction with this research, a MIDI wind controller was designed and constructed to provide a playable, intuitive interface for the model [4]. Of particular interest was a finger sensor which could provide a range of position data for mapping to a range of tonehole states between open and closed extremes. The seven FSRs of *The Pipe* can be configured to provide this functionality.

As a tonehole controller, *The Pipe* provides MIDI formatted messages to a computer-based synthesis engine running a real-time implementation of the tonehole model. Custom MIDI control values for finger pressure were designated for each of the toneholes. Additional controls, including breath pressure, breath noise, and register hole state, are configured from the available sensor inputs. The one-to-one mapping of finger pressure to tonehole state provides an intuitive interface to the model. In some instances, however, it becomes difficult to consistently maintain the pressure necessary on all FSRs to prevent inadvertent upstream hole "leakage". This problem points to an inadequacy with the *Pipe's* finger "key" mechanism which will be investigated in the future. In response to this difficulty, an alternative binary position scheme was programmed such that light finger pressure triggers tonehole closure, while a potentiometer is used to vary the rate of finger hole closure.

3.2 A MIDI Sequencer

Devices making use of programmable microprocessors can easily be configured as simple MIDI sequencers for use with external MIDI synthesizers. In general, a recurring pattern of notes, or ostinato, is programmed as a loop and subdivisions within that interval are denoted for possible sensor-controlled events. Applications of this type were previously explored using the *Phoney Controller* [5] and are easily repeated and extended with *The Pipe*.

Flexible, improvisatory control mappings within a constrained musical "scene", as created by the sequence, can provide hours of entertainment for musicians and non-musicians alike. Force sensing resistors function well as volume controls for distinct voices and auxiliary rhythm section instruments. Potentiometers are convenient for controlling tempo or selecting voices. Within a fixed modal harmonic scheme, a scale and/or octave changes can be mapped to tilt sensors to provide an easily mastered, "no fail" improvisatory control surface.

3.3 A Synthesis ToolKit Interface

The Synthesis ToolKit in C++ (STK) is a flexible set of open-source audio signal processing and algorithmic synthesis classes written in the C++ computer programming language [2]. STK provides a standardized control interface for most of its synthesis algorithms. Configuration of *The Pipe* for generic control of STK algorithms involved a variety of sensor mapping issues.

In most STK algorithms, a control parameter is provided for energy input to the instrument. For wind instrument algorithms, this parameter corresponds to breath pressure, for shaker instruments it corresponds to shake energy, and in other cases it is a volume control. Given the array of sensors on *The Pipe*, outputs from both the air pressure sensor and the accelerometer were mapped to the energy parameter. While providing intuitive control mechanisms, this mapping also allows one to "blow" shaker instruments or "shake" wind instruments.

Most of the remaining STK algorithm parameters are designed to be modified both above and below default values. A mapping strategy to allow this type of control was developed as previously discussed, such that FSRs are used as "switches" to activate particular control parameters for modification by a rotary pot. In this way, the potentiometer decrements or increments a parameter value selected by a corresponding FSR and this value is saved for subsequent adjustment. The other rotary potentiometer is used to make program changes. In addition, the two momentary switches are used to trigger "Note On" and "Note Off" events.

The primary difficulty observed while using *The Pipe* with STK algorithms has been the lack of a visual mechanism for determining existing parameter settings. Given that a single rotary potentiometer, in conjunction with distinct FSRs, is used to modify a number of different parameters, it becomes difficult to remember the last value setting for a particular parameter. Also, this scheme allows only a single parameter to be adjusted at a time. In a performance situation, a more responsive mapping would likely be necessary.

3.4 An Expressive Performance Controller

The Pipe was designed in part to allow expressive and subtle control of real-time physical models in a performance setting. Explorations in this context are ongoing though several mapping strategies are evident. Breath pressure control offers an intuitive means for "energizing" systems continuously driven by air or bows, such as wind instruments or bowed strings, bars, or bowls. While struck or plucked systems can be triggered using momentary switches, a more natural mapping can make use of velocity-based physical gestures which are calculated from the accelerometer inputs. Static tilt can be appropriately mapped to parameters which are typically varied above or below a default value for relatively short periods of time.

Subtle performance control with real-time physical models demands the ability to make minute parameter adjustments, sometimes simultaneously across several different parameters and/or instruments. When interfacing to a computer-based synthesis environment, some of these demands can be transferred to the synthesis system and simply triggered by the controller to achieve results beyond the capabilities of the device or performer.

4. OBSERVATIONS AND FUTURE WORK

Musical activities with *The Pipe* are continuing and possible improvements are actively being explored. Changes to the current breath pressure mechanism are being considered to provide a more air-tight fitting between the mouth cap and player's face, as well as more sensitivity and resolution from the sensor. In addition, finger position sensors with greater sensitivity and better tactile characteristics are desirable. In their current form, there is no way of knowing (without extensive practice) what amount of finger pressure represents full "deflection" of the finger sensors.

In its existing configuration, it is difficult to make adjustments to the rotary potentiometers when both hands are positioned on the finger "keys". It is unlikely that an appropriate modification can be made to the existing structure, though a future improvement could instead make use of a thumb operated "roller" mechanism.

The addition of a small microphone in the mouth cap to record vocalizations has been contemplated, though this would require an extra audio connector. At this point, a lip pressure sensor has not been considered, though input/output pins are still available on the microprocessor for future sensor extensions.

5. ACKNOWLEDGMENTS

This author would like to thank Perry Cook for early music controller inspirations and suggestions. A number of individuals on the *Wind Controller Mailing List* provided helpful information about existing commercial wind controllers as well as feedback on wind controller design issues. And discussions with Bill Verplank at CCRMA helped motivate a return to this project after more than a year of inactivity.

6. REFERENCES

[1] P. R. Cook. Physically informed sonic modeling (phism): Synthesis of percussive sounds. *Computer Music Journal*, 21(3):38–49, 1997.

[2] P. R. Cook and G. P. Scavone. The Synthesis ToolKit (STK). In *Proceedings of the 1999 International Computer Music Conference*, pages 164–166, Beijing, China, 1999. Computer Music Association.

[3] A. Hunt, M. Wanderley, and M. Paradis. The importance of parameter mapping in electronic instrument design. In *Proceedings of the NIME-02 Conference on New Interfaces for Musical Expression*, pages 149–154, 2002.

[4] G. P. Scavone. The Holey Controller, 1998. http://www-ccrma.stanford.edu/~gary/.

[5] G. P. Scavone. The Phoney Controller, 1999. http://www-ccrma.stanford.edu/~gary/.

[6] G. P. Scavone and P. R. Cook. Real-time computer modeling of woodwind instruments. In *Proceedings of the Internation Symposium on Musical Acoustics (ISMA-98), Leavenworth, WA*, pages 197–202, June 1998.