**Sam Hamm**
Florida Electroacoustic Music Studio
The University of Florida School of Music
P.O. Box 117900
Gainesville, FL 32611-7900
http://emu.music.ufl.edu
http://samhamm.com
sam@samhamm.com

# Sensor Selection in Design of Alternative Interfaces for Music, Dance, and Other Art Forms

**ABSTRACT**

Technological advancement has always been a vehicle for musical innovation. New materials, new inventions, and new archetypes lead to new ideas for musical and artistic expression. Computers and microprocessor technology are allowing the development of an entire new breed of musical interfaces that transcend the traditional function and role of musical instruments. It is now possible to create interfaces that can be used not only in music performance and composition, but also to integrate performer or user control of music and sound in other artistic genres, such as dance or the visual arts. It is a field of remarkable opportunity, but the opportunity comes at the price of a steep learning curve. Designing a system for an alternative interface is a challenging and multi-faceted task with several layers and functions of technology to consider. Many interface systems include the following components: a host computer, communications between the host computer and the interface, a separate microprocessor dedicated to the operation of the interface, and a sensor array that gathers data from the performer or user of the interface. This paper focuses specifically upon the selection of sensors for alternative interfaces. The sensor is the point of contact between the interface and the user/performer, and can operate in a variety of ways fulfilling a myriad of roles. Since the sensor is the origin of the control data generated by the interface, the selection of sensors immediately and simultaneously opens options for and places limitations upon the ultimate use, function, and potential for expression of any alternative interface.

## Introduction

This article has emerged from research into the design and development of alternative interfaces. When confronted with a large and daunting challenge, a fundamental methodology of engineers is to break a large problem into a set of smaller problems that are more easily solved, and then assemble the tested parts into a working whole. Musicians do this as well: performers and conductors will rehearse passages of music as a means of perfecting the whole; furthermore, composers often utilize the formal design of a work as a means of organizing and controlling the compositional process. These observations represent guidelines rather than hard-and-fast rules, and they signal a similarity in approach that is useful in studying the integration of music and technology.

A helpful technique of investigating and breaking down this problem is to view existing musical instruments from the perspective of an engineer: what kinds of controls and materials would be needed to replicate the basic function of known instruments? The value of this exercise is not to gather data for use in an imitation of an instrument that already exists; rather, it aids in understanding and confronting some of the fundamental problems that concern the designer of a new interface. How many things can a performer do simultaneously, with what variety of output, to what degree of resolution, within what kind of time-response framework? These are all vital concerns for the designer of a new interface, and will be touched upon in this document. Each is a topic large enough for its own paper, but some of the basic ideas must be kept in mind for now.

This document focuses upon the considerations involved in sensor selection These considerations include what the varying types of sensors are that can be used, how they work, what sort of parallels exist between a given sensor and musical instruments, and how a sensor may be utilized in an alternative interface. It is assumed that the reader of this document has a working knowledge of computers and software, but not necessarily electronics. Therefore, it is the intent of this document to bridge the gap between composer and engineer, serving more to provide useful insight and perspective to interface design, the aim is to inspire future activity in the field, rather than to present a set of findings and conclusions.

## Prototype of an Interface Design

***Instrument vs. Interface***

The distinction between a *musical instrument* and a *musical interface* is not easily drawn. It is useful, however, to explore the meanings of these terms, so as to bring awareness to existing biases and ideologies, and to establish a method for quantifiable comparison.

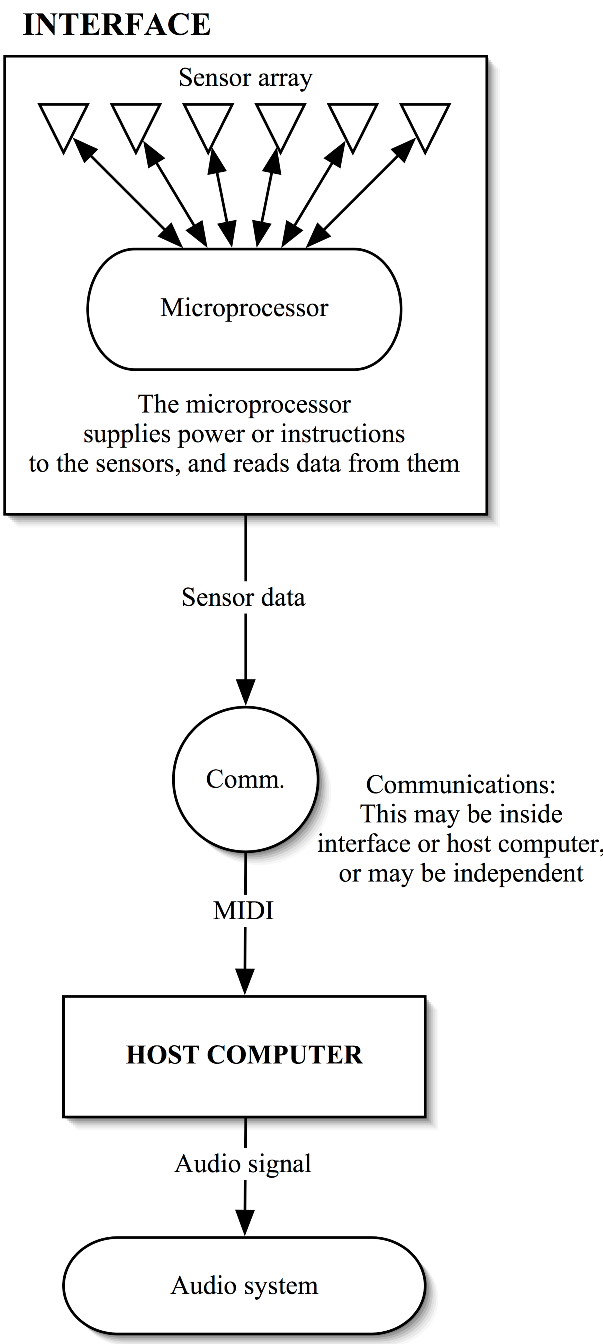
For the purposes of this article, the definition of *musical instrument* will maintain a historical bias. It is a device that facilitates musical performance through the creation and expressive control of sound. This definition encompasses all traditional acoustic instruments, as well as some more recent innovations such as synthesizers.

A *musical interface*, on the other hand, allows for a variety of possibilities, wherein the interface is capable of controlling different types of musical and auditory elements. These elements may consist of creating of sound, control of recording and playback of sound, processing of sound, as well as other kinds of interactions between a user or performer and the sound environment. This definition can include the traditional instrument approach, but when an interface extends the functionality of an existing instrument, it begins to cross the line into interface. A gray area exists, and it is not the intent of this article to resolve the question. Keeping the possibilities in mind along with existing paradigms, however, allows for more productive and cohesive discussion.

We will work from the definition of interface that allows the broadest possibilities of interface function.

***Overview and Terminology***

There are many possible kinds of interface designs; however, for our discussions we will start with a common prototype:




The output of an alternative interface begins with the sensor array. Sensors can operate in a variety of ways: some measure motion, some simply act as on/off triggers, and others may indicate some other specific type of activity. Sensors may be digital or analog. Digital sensors output a series of discrete values at a specified sampling interval, whereas analog sensors output a varying voltage that must then be sent through an analog-to-digital (A/D) converter in order to be used by a computer.

Numerous interfaces utilize a built-in (or, “on-board”) microprocessor that supplies power to and reads data from the sensor array. In these cases the processor also handles A/D duties. The selection of an on-board microprocessor depends upon the communications options, the macrodesign of the interface, and the types of sensors being used.

Many alternative interface systems also involve use of a host computer that must be programmed with response characteristics based upon the output of the actual interface. The means of communication between the host computer and the interface presents some important concerns, such as what communications protocol to use (MIDI is common, for instance), whether the main connection should be wired or wireless, and the possibilities and/or limitations of each choice and combinations thereof.

For the purposes of this document, I will assume this fundamental prototype of an alternative interface. Certainly, a variety of other designs and structures are possible that utilize a greater or lesser number of components than those outlined here. In those designs there is a tradeoff between flexibility and complexity. Greater flexibility often requires greater complexity, thus giving rise to more difficulty in implementation. A more detailed examination of each of the components of our prototype follows.

### *Sensor Array*

The sensor array is the part of the interface that gathers data and information from the performer, user, or environment. The considerations involved in selecting components for the sensor array is the primary focus of this article.

The sensor array can take on a variety of different forms. It can be a simple set of buttons or a complicated collection of various types of sensors such as pressure sensors, infrared sensors, accelerometers, and flex sensors. The complexity of the sensor array is dependent upon the type and amount of data desired by the designer; it must be remembered, as was stated in the prior section, that intricate designs lead to greater possibility of complications.

Generally, the limitations of the sensor array are determined more by the capabilities of the performer(s) or user(s) than it is by technological constraints. There are a number of technological issues to consider, such as the power demand of the sensors, the potential for electronic interference from environmental factors, the number of input/output ports on a microprocessor, or the limitations of the selected communications protocol. In addition, a composer or interface designer must devise a means of managing all of the data that is received.

In my conversations with various designers and users of alternative interfaces, such as Richard Boulanger of the Berklee School of Music and Jeff Stolet of the University of Oregon, one mantra keeps reappearing: “Keep It Simple, Stupid”, a.k.a. “The K.I.S.S. Rule”. The underlying axiom is that it is often challenging for a performer to learn to control and achieve meaningful expression with an alternative interface. Naturally, this concept will have a direct effect upon sensor selection.

### *On-board Microprocessor*

Though we may not be aware of it, many electronic devices in common use have microprocessors built into them. These processors commonly run at clock speeds of less than 10 megahertz (MHz), and thus seem woefully slow and limited when compared to the processors in personal computers, which have speeds usually measured in the gigahertz range. However, even a 2 MHz chip produces more than two million clock cycles per second, which is much faster than data rates used for audio and musical control data, including MIDI, which will be discussed in the following section..

In our prototypical alternative interface, an on-board microprocessor will perform several tasks. To do these tasks, the microprocessor must first be programmed. Each microprocessor has its own unique programming language that can be used, and many also have translators available that will allow a user to write code in a common language such a C, C++, Fortran, or Java, that can then be modified for use by a specific microprocessor. Once the microprocessor is programmed, it can perform its duties.

First, it supplies power or a reference signal to the sensor array, which is then modified based upon what is measured by the sensor. In the case of a digital sensor, the microprocessor simply receives the data sent by the sensor. In the case of analog sensors, the microprocessor receives the modified signal from the sensor and converts it into a value representing the measurement. This can be done a number of ways; one method that is common is A/D conversion, discussed previously. Another method involves using the value returned from the sensor as an index in a lookup table of predetermined responses. This data is then sent by the microprocessor to the host computer using whatever communications protocol has been established.

These microprocessors and their development kits are produced by several manufacturers, such as Motorola and Atmel, and are widely available from electronics retailers such as Jameco and Digikey. There are a tremendous number of resources available such as books, manuals, and datasheets; usergroups and clubs can also assist in learning how to utilize these devices.

### *Communications with Host Computer*

The most common communications protocol in electronic music is MIDI (Musical Instrument Digital Interface). It is safe to assume that many musicians simply think of MIDI as a means of generating bad realizations of acoustic scores, and unfortunately, there is a wealth of material to reinforce that notion. It is important to remember, however, that MIDI emerged as a standardized tool to allow communications between different kinds of musical instruments and equipment. To suggest that MIDI is somehow tainted because of the way it is commonly used is akin to suggesting that the telephone is flawed because people primarily use it for mindless, time-wasting conversation.

As a communications protocol, MIDI is well-suited for use in alternative interfaces. MIDI permits the transmission of up to sixteen channels of 8-bit data; in other words, a single MIDI channel can transmit numerous kinds of data, each at any value between zero and $2^8$-1 (255), usually scaled for presentation within the range ±127. MIDI operates at a rate of 31.25 kilobaud, meaning that up to 31,250 bits per second may be transmitted via MIDI. Thus, an alternative interface can provide output, via MIDI, of data from up to sixteen sensors with 255 grades of resolution at time intervals of a few milliseconds or less. For many musical purposes, this is adequate.

Another advantage of MIDI is that it is based on standard serial communications protocols. This allows standard industrial and project-based microprocessors to be programmed to convert sensor data into MIDI messages that can be output to musical devices. This will be discussed in more detail in the following section.

It is also possible for the computer to receive data from a source that outputs data in a format other than MIDI, and for the host computer to simultaneously run a specialized program that translates that information into MIDI. One such program is called Junxion, produced by the Steim Foundation based in Amsterdam. This program, which receives data from any USB device and translates it into MIDI, is very useful and effective, and is highly recommended.

### *Host Computer*

The host computer for an alternative interface is used to run software that processes or generates audio. The software is programmed with some set of characteristic responses based upon the data received from the interface. The nature of these responses is dependent solely upon the intent of the composer.

The choice of software is an important one. Some inventors and designers of alternative interfaces will write their own software for use on the host computer. More common, however, is the situation where the data received from the interface is in MIDI format, and this data is then utilized in a software package that can receive and respond to MIDI in real-time, for example, Max/MSP, Supercollider, Kyma, or Pro Tools. If a composer has background with a particular software package, there is an advantage in learning to harness and apply the data provided by the interface.

Now that the parts and basic functions of an alternative interface have been laid out, it is time to examine sensors in more detail.

## Mechanical Sensors

For clarity, I have divided sensors into two categories: mechanical and non-mechanical. The first category, mechanical, indicates sensors that measure some kind of motion or physical quantity through direct physical contact. These sensors typically require careful placement and mounting so as to be accessible to the performer or user, and for the sensor and its connections to physically withstand the processes or actions that are being measured. These sensors have the advantage of being less prone to environmental interference than the other category of sensors, non-mechanical.

### *Accelerometers*

Accelerometers can serve a variety of functions. In addition to measuring displacement and motion, simultaneously in up to three axes or directions, these versatile sensors can be used to measure tilt or vibration. And, since measurements can take place without need of a solid physical reference, the freedom afforded by accelerometers is appealing because the need for hardware support structures is diminished. For instance, some sensors such as pressure sensors and flex sensors must be carefully mounted on an object so as to measure and register meaningful and useful data.


Accelerometers do have drawbacks, however. They are considerably more expensive than other sensors, and the three-dimensional data they provide can be difficult to interpret and harness. Selecting the correct accelerometer for a given purpose can be challenging, as there are a variety of types differentiated by specifications that are not easy to measure or represent in normal life. For instance, how many G's are involved in raising your arm over your head, and how sensitively would you want to measure that motion?

***Flex Sensors***

Flex sensors are a common component of alternative interfaces such as gloves and dance interfaces. These sensors are supplied a reference voltage, and depending upon the amount the sensor is flexed or bent in either direction, returns a different voltage to the microprocessor. These sensors are very durable and dependable, and can be used in many different ways.

It can be challenging to mount flex sensors effectively, however. They require a combination of stable mounting so as to maintain proper positioning at the point of flexing, but also need a limited amount of mobility so as to accommodate the motion since these sensors are typically mounted directly upon the object being measured. In addition, care must be taken to leave appropriate amounts of slack in the attached wiring.

***Potentiometers***

Most musicians are familiar with potentiometers, for we deal with them on a regular basis. It would be a mistake to think of a potentiometer as a simple know, however. With additional materials and mechanical design, a potentiometer can be attached to any number of moving parts and used to measure some sort of relative spatial movement of the parts. For instance, a potentiometer mounted to measure the angle of movement of a hinge, such as would be used on an elbow or knee in a dance interface, connected by wire and a spring to measure the amount of pressure used to pull on an object.

There are tremendous varieties in the kinds of potentiometers, and different types and sizes of potentiometers are suited for different uses.

***Pressure sensors***

Pressure sensors are used to measure the amount of pressure applied to whatever object the sensor is mounted upon. Pressure sensors are available in numerous sizes and ranges of measurement.

***Pushbuttons***

Pushbuttons represent the most clear analogy to the controls or potential musical instruments: after all, most wind instruments can be viewed as being made up of three main control systems: a breath