

A New Technology for Musical Sound Synthesis and Control

RICHARD HODGES

*Gibson USA
2560 9th St.
Berkeley CA 94710*

Abstract

In using electronic music synthesis for performance, an important issue is providing the musician with a familiar and powerful way of controlling the instrument. Keyboards have been the most common control device, but the keyboard is limited in expressive possibilities compared to other instruments such as the guitar. We are developing an electronic instrument which uses the electric guitar as a performer interface which controls a wide palette of sounds and special effects.

We use a special guitar that is fitted with a fret scanning system and with separate magnetic pickups on each string. These inputs are processed digitally to analyze the guitarist's musical gestures. Several parameters are calculated for each string in real time and passed to a synthesis engine which plays sampled sounds and applies controllable modifications to the samples.

In another mode of operation, the actual signals from the guitar strings are modified by combinations of basic algorithms such as non-linear distortion, pitch shifting, and tuneable filtering.

This instrument incorporates a new digital protocol for communicating music performance data, called ZIPI. This protocol has features lacking in MIDI, including real-time updating of information for individual notes.

Electronic music instruments: promise unfulfilled

Electronic musical instruments have become very important in live musical performance and studio production. They have several advantages over conventional acoustic and electro-acoustic instruments: wide palette of sounds; flexibility and ease of control; and reproducibility of sound, rhythm, and intonation. But they have several disadvantages also. Principally, these stem from the keyboard model on which most electronic instruments are based. The keyboard has severe limitations as a musical interface and control device. Control of musical events is limited basically to timing of attack and release, and velocity of attack (of course some keyboards also support after-touch). Electronic instruments differ greatly from traditional instruments in which the musician is directly in touch with the sound producing vibrations. In wind instruments, he can make subtle and rapid changes in timbre and pitch by changing the pressure of his lips, tongue, and diaphragm. In plucked string instruments such as the guitar, he can change the timbre by where and how he plucks the strings and by touching the strings before and during the note, and he can change the pitch by "bending" the strings. In bowed string instruments, the possibilities of control are even greater.

Additional limitations on electronic instruments are incurred by the nearly universal electronic protocol known as "MIDI" which is used for controlling, recording, and communicating performances of electronic musical instruments. MIDI has two chief problems which relate to its origin as a way of communicating keyboard data: speed of communication, which can result in noticeable delays in musical events; and lack of a flexible and powerful way to specify modifications to ongoing notes.

As a result of all these factors, musicians have been frustrated by the inability to combine the advantages of electronic and conventional instruments. The benefits of electronic sound generation and manipulation have not been extended to the widest possible range of different kinds of music.

A new kind of electronic instrument

We are attempting to address the limitations of electronic music with a new family of instruments that provide much more intimate real-time control to the musician, and a wider palette of sounds and sound-modification algorithms. The first instrument in this family is a second-generation instrument based on the electric guitar. The basic aim is to enable the use of the electric guitar as a controller in such a way that the skilled guitarist can employ the large repertoire of instrumental techniques he has laboriously developed to control the synthesis and processing of musical sounds. Other instruments, including violin, woodwinds, voice, and drums, can also be fitted with electronic pickups and used as inputs to this new device, giving a similar range of expressive power to skilled performers on these instruments.

This work builds on experience gained with the guitar- and violin-based synthesizers developed and marketed by ZETA Music, which is now a division of Gibson USA. The principal predecessor instrument, the ZETA Mirror-6 MIDI guitar, used a combination of fret scanning, independent electronic pickups on each string, and real-time pitch analysis to produce MIDI signals that can be used to control a conventional electronic synthesizer.

The new instrument goes far beyond the earlier ZETA instruments in several directions. The continuing evolution of digital audio electronics allows much more complex processing of the guitar signals in real time within the practical constraints of a commercially viable package and makes possible many new capabilities in electronic music instruments. Features of the new instrument include:

- Built-in 16-bit sample playback synthesis engine, with high-quality sounds in internal ROM
- PCMCIA card interface for libraries of additional sounds, patch storage, and firmware upgrades
- Stereo output, with controllable pan for each voice
- Fast and accurate tracking of guitar strings vibration parameters
 - Pitch (fret pitch and instantaneous pitch)
 - Amplitude
 - Spectral envelop
 - Spectral Timbre analysis
- Complex synthesis engine
 - Multiple-component sound wave files (odd harmonics; even harmonics; non-pitched component)
 - Post-synthesis filter with real-time controllable parameters
 - Real-time matrix mapping of analysis parameters and external inputs to control synthesis
- ZIPI (and MIDI) protocol for fast flexible communication of control parameters
 - Can control other synthesizers
 - Can respond to external controllers
- Hexaphonic waveform mode processes string signals individually
 - Six channels of 16-bit low-noise digital-to-analog conversion
 - Independent processing of each string signal
 - Variable pitch-shifting and harmonizing
 - Other nonlinear algorithms
 - Pre- and post-filtering, controllable in real time

Implementation

The new instrument employs several high-speed digital processors, including CISC, RISC, and DSP technology. These processors perform all functions for analyzing and synthesizing waveforms, user interface, and communications, allowing functions of the instrument to be upgraded and customized in the field by loading new firmware.

In order to obtain accurate string pitch information, we perform a time-domain analysis of the waveform of each individual string. Based on our experience with the ZETA Mirror-6 MIDI guitar controller, the time periods are measured between successive inflection points of the waveforms and a variety of heuristics are

applied to these raw measurements. These heuristics are aided by knowledge of the position at which the string is fretted; allowance has to be made however for special techniques such as harmonics and deep whammy-bar pitch bends. A very accurate value for string vibration frequency can be determined in a little more than one complete period of the waveform. Accurate control signals for synthesis are produced with a time delay that is musically minimal.

We also use a digital processor to perform a spectral analysis. This enables us to measure parameters reflecting the timbre of the string signal. Spectral tilt is determined as a weighted ratio between high- and low-frequency components. We also measure the proportion of three different components of the string signal: odd harmonics; even harmonics; and non-harmonic vibrations (i.e. components of the sound not resulting from the vibration of the string). These kinds of information are computed in real time and combined according to the user-specified program or "patch." The results are used as control information for the synthesis engine.

A digital processor is used for synthesis of output waveforms. Sample files representing different instrument voices are accessed from built-in ROM or from a PCMCIA card; as in conventional sampling synthesizers, these instrument samples are interpolated and resampled to shift them to the pitch determined by the user patch. Often, the synthesized pitch would be the actual instantaneous pitch of the guitar string; however one major advantage of our instrument is that the pitch can be transposed by a fixed tonal interval or otherwise modified for musical effects. Each instrument voice is actually represented by three different files which contain the odd harmonic information, the even harmonics, and the non-harmonic information for the voice. These files are obtained by complex off-line processing of sound recorded from actual instruments under studio conditions.

Digital filtering is applied to the voices synthesized for the six strings. The resulting signals are mixed and panned between left and right output channels. These functions are also controlled by analysis parameters as specified in the user patch. Finally the signals are sent to two 16-bit digital-to-analog converters for output to external sound amplification equipment.

In the hexaphonic waveform mode, the processors perform a variety of algorithms which act directly on the waveforms generated by the strings and modify them; it is these modified waveforms that are sent to the output, rather than synthesized waveforms. This is an important extension of the technique of applying deliberate electronic distortion to the sound of the electric guitar. This technique became enormously important in commercial music due to the success of the genre of Heavy Metal. In our instrument, the modifying algorithms can be applied to the waveform of each string individually, or to combinations of the string signals. The advantage is that whereas traditional distortion devices inevitably result in cross-modulation products of the signals of the simultaneously played strings, in our instruments, this does not necessarily occur. Of course cross-modulation is sometimes musically desirable, but with our instrument it can be controlled. Also, importantly, our method allows the natural dynamics of the guitar to be reflected in the output dynamics if desired, as opposed to conventional nonlinear processing which results in dramatic compression of the dynamics.

The power of digital processing allows much more complex algorithms than the simple nonlinear mapping of waveform that was characteristic of previous distortion devices. We have implemented algorithms for pitch shifting and harmonizing of the guitar notes, preserving all timbral and dynamic qualities of the original signal. Other algorithms are under development which generate transformations of string timbre.

In the hexaphonic mode, individual filters for each string are also implemented which act both before and after nonlinear processing. These filters can be controlled in real time based on measured properties of the string signals.

A new paradigm for electronic music instrument communication

Contemporary practice in electronic music is to use modular units connected by MIDI cables. A typical setup for live performance or studio recording consists of several units including controllers (keyboards, drum pads, breath controllers, guitar and violin controllers), sound generators (samplers are the most common type today, although FM and other technologies are widely used also), and integrated units combining a keyboard and sound generation capability. In addition, personal computers are often fitted with MIDI interfaces which allows them to run software to record, play back, and modify MIDI data during performances. The development of the MIDI standard began about a decade ago in response to the need for such networks of music performance devices.

However, MIDI is saddled with numerous limitations. First, there is speed. In order to utilize the serial communication IC's used in early personal computers, the communication speed was limited to 31.25 kbaud.

Since the basic performance model of MIDI was the keyboard controller, which can only generate a limited number of note-on and note-off events at one time, this was marginally adequate. For example, the attack information for a ten-note chord can be transmitted in about 7 milliseconds, which is on the borderline of perceptibility. However, serious speed problems appear when several controllers are sharing the same MIDI network, a common practice, when musicians try to use techniques that depend on very accurate time relationships between events, for example flam techniques on percussion controllers, and especially when information from pedals, joysticks, and other continuous controllers needs to be updated at a rapid rate. Communicating guitar controller information becomes very difficult; just to update pitch bend information 100 times a second for six strings would exceed MIDI bandwidth.

Another category of MIDI limitation stems from its implicit model of musical events. Basically, MIDI assumes that the principle information about a musical event is established at the attack, and comprises simply semitone pitch and velocity (a keyboard concept translating into amplitude). This model derives from MIDI's heritage as a representation for keyboard and percussion events, where it is appropriate. But it is not a good model for instruments where the performer remains in control of pitch, amplitude, and timbre during the course of the entire note. Some features were added to MIDI to try to accommodate continuous information: controllers based on the joystick and foot pedal model; keyboard aftertouch; and pitch bend. These features are not well integrated in the MIDI model however, and different implementations are not always compatible.

People have been complaining about these problems ever since MIDI first began to be used, but prior to our efforts, no one had done anything about them. Faced with the requirement for real-time musical communication suitable at least to the guitar controller, we have developed a new protocol called ZIPI. This protocol addresses the major problems with MIDI that we have mentioned and serves the future needs of a wide variety of different controllers and performance situations.

The ZIPI protocol is designed on the philosophical base of contemporary practice in high-speed digital communications, and conforms to the OSI layered network model. At the physical layer, we define an electrical specification and a 250 kbaud (minimum) serial stream. At the data link layer, we use a token-ring architecture for deterministic performance. In complex setups where there are several controllers and synthesizers, these instruments would normally be connected in a star topology through a hub which would implement the ring and maintain reliable communications even in the presence of transmission errors and events where instruments go on-line or off-line during a performance.

The main content of the ZIPI protocol is at the application layer of the OSI model, where we define several protocols, most importantly a Music Parameter Description Language, MPDL, for communicating real-time music event control (McMillen, Wessel, & Wright 1994). A key feature is the concept of a hierarchical address space. Control messages can be addressed to instruments, to families (of instruments), or to individual notes. Addressability of notes is an important advance over MIDI, since it allows a single note to evolve over time, changing its properties such as pitch, time, and amplitude, in response to ZIPI messages addressed to it.

The messages "trigger" and "release," which start (or rearticulate) a note and stop it respectively do not in themselves contain information about the pitch or amplitude of the note. These properties of a note are established by other ZIPI messages that are sent before or during the note. A large number of different types of messages are given predefined meanings (although detailed semantics are in many cases left to the implementation, for example the mapping of the "amplitude" message value into physical units). Some message types provide redundant ways to control the same physical parameter, for example "pitch," specified in semitones and fractions of semitones, and "frequency," specified in Hertz. Timbre space is addressed by predefined messages such as "brightness," "Even/Odd harmonic balance," "roughness," "attack character," and by messages specifying abstract timbre space coordinates, the meaning of which is left entirely to the implementation of individual instruments. Other messages are provided for spatial location of instruments (a generalization of pan), for modulation (a generalization of vibrato), and for synthesizer housekeeping functions such as defining the priority of a note for allocation to available instrument voices.

In addition to messages whose semantics are defined in terms of parameters that affect the way a synthesizer is to generate note sounds, another class of messages is provided whose semantics are specified in terms of measured performer gestures. Examples of these messages include "key velocity," "pick/bow velocity," "lip pressure," "drum-head position," "position in space" (i.e. position of a magic wand-type controller), etc. In MIDI, there was no distinction between these two classes of message, and in practice the semantics of messages was sometimes interpreted as if they were what we would call synthesizer control messages, sometimes as performer gesture messages; this confusion was the cause of some cases of inconsistent behavior with MIDI.

ZIPI provides communication bandwidth and control flexibility able to be the basis for whole orchestras combining many heterogeneous controllers and synthesizers, and supports a vast range of expressive possibilities for many new types of electronic music performance environments.

Conclusions

We have developed a new generation of electronic music instruments allowing an unprecedented degree of control and expressiveness. A guitar controller with integrated sound synthesis and processing capabilities will be the first member of this family. It will be released as a commercial product early in 1995.

To support our new control concepts and sound generation methods, we are also introducing a new protocol for digital communication between instruments in a performance network. These two elements are enabling technologies, allowing electronic instruments to enter a new age of diverse musical creativity.

References

- International MIDI Association (IMA) (1988) *MIDI 1.0 Detailed Specification, Document Version 4.0*, Los Angeles, CA, IMA.
- Loy, G. (1985) Musicians Make a Standard: The MIDI Phenomenon. *Computer Music Journal*, 9(4), 8-26.
- McMillen, K. (1994). ZIPI—Origins and Motivations. *Computer Music Journal*, 18(4). [in press]
- McMillen, K. Wessel, D. & Wright, M. (1994). ZIPI's Music Parameter Description Language. *Computer Music Journal*, 18(4). [in press]
- McMillen, K. Simon, D. & Wright, M. (1994). ZIPI Network Summary. *Computer Music Journal*, 18(4) [in press]
- Moore, F. R. (1988). The Dysfunctions of MIDI *Computer Music Journal* 12(1), 19-28. In Deutsch, D., ed. *The Psychology of Music 2nd Edition*, Academic Press.
- Scholz, C. (1991). A proposed extension to the MIDI specification concerning tuning. *Computer Music Journal*, 15(1), 49-54.
- Wright, M. (1994). ZIPI Examples. *Computer Music Journal*, 18(4). [in press]
- Wright, M. (1994). MIDI/ZIPI Comparison. *Computer Music Journal*, 18(4). [in press]
- Wright, M. (1994). ZIPI Frequently Asked Questions. *Computer Music Journal*, 18(4). [in press]

Acknowledgements

I would like to acknowledge the creators and implementers of the concepts reported here: Keith McMillen, Chief Scientist, Gibson Western Innovation Zone Labs (G-WIZ), and the staff of G-WIZ, especially Marie Baudot; and David Wessel and the staff of the Center for New Music and Audio Technology (CNMAT), music department, University of California at Berkeley.