MIDI

From Wikipedia, the free encyclopedia

MIDI allows multiple instruments to be played from a single controller, which makes stage setups much more portable. This system fits into a single rack case, but prior to the advent of MIDI would have required four separate keyboard instruments, plus outboard mixing and effects.

MIDI (/ˈmɪdi/; short for **Musical Instrument Digital Interface**) is a technical standard that describes a protocol, digital interface and connectors and allows a wide variety of electronic musical instruments, computers and other related devices to connect and communicate with one another. A single MIDI link can carry up to sixteen channels of information, each of which can be routed to a separate device.

MIDI carries event messages that specify notation, pitch and velocity, control signals for parameters such as volume, vibrato, audio panning, cues, and clock signals that set and synchronize tempo between multiple devices. These messages are sent to other devices where they control sound generation and other features. This data can also be recorded into a hardware or software device called a sequencer, which can be used to edit the data and to play it back at a later time.

MIDI technology was standardized in 1983 by a panel of music industry representatives, and is maintained by the MIDI Manufacturers Association (MMA). All official MIDI standards are jointly developed and published by the MMA in Los Angeles, California, US, and for Japan, the MIDI Committee of the Association of Musical Electronics Industry (AMEI) in Tokyo.

Advantages of MIDI include compactness (an entire song can be coded in a few hundred lines, i.e. in a few kilobytes), ease of modification and manipulation and choice of instruments.

MIDI

The development of MIDI

By the end of the 1970s, electronic musical devices were becoming increasingly common and affordable in North America, Europe and Japan. Early analog synthesizers were usually monophonic, and controlled via a voltage produced by their keyboards. Manufacturers used this voltage to link instruments together so that one device could control one or more others, but this system was inadequate for control of newer polyphonic and digital synthesizers. Some manufacturers created systems that allowed their own equipment to interconnect, but the systems were incompatible, so one manufacturer's systems could not synchronize with those of another.

Sequential Circuits engineers and synthesizer designers Dave Smith and Chet Wood devised a universal synthesizer interface, which would allow direct communication between equipment from different manufacturers. Smith proposed this standard at the Audio Engineering Society show in November 1981. Over the next two years, the standard was discussed and modified by representatives of companies such as Roland, Yamaha, Korg, Kawai, Oberheim, and Sequential Circuits, and was renamed Musical Instrument Digital Interface. MIDI's development was announced to the public by Robert Moog, in the October 1982 edition of Keyboard magazine.

By the time of the January 1983 Winter NAMM Show, Smith was able to demonstrate a MIDI connection between his Prophet 600 analog synthesizer and a Roland JP-6. The MIDI Specification was published in August 1983.

MIDI's impact on the music industry

MIDI's appeal was originally limited to those who wanted to use electronic instruments in the production of popular music. The standard allowed different instruments to speak with each other and with computers, and this spurred a rapid expansion of the sales and production of electronic instruments and music software. This intercompatibility allowed one device to be controlled from another, which rid musicians of the need for excessive hardware. MIDI's introduction coincided with the dawn of the personal computer era and the introductions of samplers, whose ability to play back prerecorded sounds allowed stage performances to include effects that previously were unobtainable outside of the studio, and digital synthesizers, which allowed pre-programmed sounds to be stored and recalled with the press of a button. The creative possibilities brought about by MIDI technology have been credited as having helped to revive the music industry in the 1980s.

MIDI introduced many capabilities which transformed the way musicians work. MIDI sequencing makes it possible for a user with no notation skills to build complex arrangements. A musical act with as few as one or two members, each operating multiple MIDI-enabled devices, can deliver a performance which sounds similar to that of a much larger group of musicians. The expense of hiring outside musicians for a project can be reduced or eliminated, and complex productions can be realized on a system as small as a single MIDI workstation, a synthesizer with integrated keyboard and sequencer. Professional musicians can do this in an environment such as a home recording space, without the need to rent a professional recording studio and staff. By performing preproduction in such an environment, an artist can reduce recording costs by arriving at a recording studio with a work that is already partially completed. Rhythm and background parts can be sequenced in advance, and then played back onstage. Performances require less haulage and set-up/tear-down time, due to the reduced amount and variety of equipment and associated connections necessary to produce a variety of sounds. Computer-aided learning enabled by MIDI has transformed the way technology is used in music education.

In 2012, Ikutaro Kakehashi and Dave Smith were awarded a Technical Grammy Award for the development of MIDI in 1983.

Applications

Standard applications

MIDI was invented so that musical instruments could communicate with each other, and that one instrument can control another. When a note is played on a MIDI instrument, it generates a digital signal that can be used to trigger a note on another instrument. The capability for remote control allows full-sized instruments to be replaced with smaller sound modules, and allows musicians to combine instruments to achieve a fuller sound, or to create combinations such as acoustic piano and strings. MIDI also enables an instrument's sounds to be programmed remotely. Synthesizers and samplers contain various tools for shaping a sound. Filters adjust timbre, and envelopes automate the way a sound evolves over time. The frequency of a filter and the envelope attack, or the time it takes for a sound to reach its maximum level, are examples of synthesizer parameters, and can be controlled remotely through MIDI. Effects devices have different parameters, such as delay feedback or reverb time. When a MIDI continuous controller number is assigned to one of these parameters, the device will respond to any messages it receives that are identified by that number. Controls such as knobs, switches, and pedals can be used to send these messages. A set of adjusted parameters can be saved to a device's internal memory as a "patch", and these patches can be remotely selected by MIDI program changes. The MIDI standard allows selection of 128 different programs, but devices can provide more by arranging their patches into banks of 128 programs each, and combining a program change message with a bank select message. MIDI consists of messages about these parameters, and contains no sound. It has nothing to say about the quality of sound that the receiving device produces, or what technique it uses to generate sound. Analog synthesizers that have no digital component and were built prior to MIDI's development can be retrofit with kits that convert MIDI messages into analog control voltages.

MIDI events can be sequenced with computer software, or in specialized hardware music workstations. Sequencers allow composers to audition and edit their work much more quickly and efficiently than did older solutions, such as multitrack recording. They improve the efficiency of composers who lack strong pianistic abilities, and allow untrained individuals the opportunity to create polished arrangements. Because MIDI is a set of commands that create sound, MIDI sequences can be manipulated in ways that prerecorded audio cannot. It is possible to change the key, instrumentation or tempo of a MIDI arrangement, and to reorder its individual sections. The ability to compose ideas and quickly hear them played back enables composers to experiment. Algorithmic composition programs provide computer-generated performances that can be used as song ideas or accompaniment. Some composers may take advantage of MIDI 1.0 and General MIDI (GM) technology to allow musical data files to be shared among various electronic instruments by using a standard, portable set of commands and parameters. On the other hand, composers of complex, detailed works to be distributed as produced audio typically use MIDI to control the performance of high-quality digital audio samples and/or external hardware or software synthesizers.

Many digital audio workstations (DAWs) are specifically designed to work with MIDI as an integral component. MIDI piano rolls have been developed in many DAWs so that the recorded MIDI messages can be extensively modified. Virtual Instruments created by third party companies in one of a number of commonly used formats (for example, VST or RTAS) may be loaded as plug-ins thus providing a virtually limitless supply of sounds for a musician, and are designed to be commanded by MIDI controllers, especially in the DAW environment. The data composed via the sequenced MIDI recordings can then be saved as a Standard MIDI File (SMF), digitally distributed, and reproduced by any computer or electronic instrument that also adheres to the same MIDI, GM, and SMF standards.

MIDI data files are much smaller than recorded audio waveforms. Many computer-sequencing programs allow manipulation of the musical data such that composing for an entire orchestra of sounds is possible. This ability to manipulate musical data has also introduced the concept of surrogate orchestras, providing a combination of half sequenced MIDI recordings and half musicians to make up an entire orchestral arrangement; however, scholars believe surrogate orchestras have the possibility of affecting future live musical performances in which the use of live musicians in orchestral arrangements may cease entirely because the composition of music via MIDI recordings proves to be more efficient and less expensive.

MIDI and computers

At the time of MIDI's introduction, the computing industry was mainly devoted to mainframe computers, and personal computers were not commonly owned. The personal computer market stabilized at the same time that MIDI appeared, and computers became a viable option for music production. In the years immediately after the 1983 ratification of the MIDI specification, MIDI features were adapted to several early computer platforms, including Apple II Plus, IIe and Macintosh, Commodore 64 and Amiga, Atari ST, Acorn Archimedes, and PC DOS. The Macintosh was the favorite among US musicians, as it was marketed at a competitive price, and would be several years before PC systems would catch up to its efficiency and graphical interface. The Atari ST was favored in Europe, where Macintoshes were more expensive. Apple computers included audio hardware that was more advanced than that of their competitors. The Apple IIGS used a digital sound chip designed for the Ensoniq Mirage synthesizer, and later models used a custom sound system and upgraded processors, which drove other companies to improve their own offerings. The Atari ST was favored for its MIDI ports that were built directly into the computer. Most music software in MIDI's first decade was published for either the Apple or the Atari. By the time of Windows 3.0's 1990 release, PCs had caught up in processing power and had acquired a graphical interface, and software titles began to see release on multiple platforms.

Standard MIDI files

The Standard MIDI File (SMF) is a file format that provides a standardized way for sequences to be saved, transported, and opened in other systems. The compact size of these files has led to their widespread use in early computers, mobile phone ringtones, webpage authoring and greeting cards. They are intended for universal use, and include such information as note values, timing and track names. Lyrics may be included as metadata, and can be displayed by karaoke machines. The SMF specification was developed and is maintained by the MMA. SMFs are created as an export format of software sequencers or hardware workstations. They organize MIDI messages into one or more parallel tracks, and timestamp the events so that they can be played back in sequence. A header contains the arrangement's setup data, which may include such things as tempo and instrumentation, and information such as the song's composer. The header also specifies which of three SMF formats the file is in. A type 0 file contains the entire performance, merged onto a single track, while type 1 files may contain any number of tracks. Type 2 files are rarely used. Microsoft Windows bundles SMFs together with Downloadable Sounds (DLS) in a Resource Interchange File Format (RIFF) wrapper, as RMID files with a .rmi extension. RIFF-RMID has been deprecated in favor of Extensible Music Files (XMF).

File sharing

A MIDI file is not a recording of actual music. Rather, it is a spreadsheet-like set of instructions, and can use a thousand times less disk space than the equivalent recorded audio. This made MIDI file arrangements an attractive way to share music, before the advent of broadband internet access and multi-gigabyte hard drives. Licensed MIDI files on floppy disks were commonly available in stores in Europe and Japan during the 1990s. The major drawback to this is the wide variation in quality of users' audio cards, and in the actual audio contained as samples or synthesized sound in the card that the MIDI data only refers to symbolically. Even a sound card that contains high-quality sampled sounds can have inconsistent quality from one instrument to another, while different model cards have no guarantee of consistent sound of the same instrument. Early budget cards, such as the AdLib and the Sound Blaster and its compatibles, used a stripped-down version of Yamaha's frequency modulation synthesis (FM synthesis) technology played back through low-quality digital-to-analog converters. The low-fidelity reproduction of these ubiquitous cards was often assumed to somehow be a property of MIDI itself. This created a perception of MIDI as low-quality audio, while in reality MIDI itself contains no sound, and the quality of its playback depends entirely on the quality of the sound-producing device (and sound recording of samples in the device).

MIDI software

The main advantage of the personal computer in a MIDI system is that it can serve a number of different purposes, depending on the software that is loaded. Multitasking allows simultaneous operation of programs that may be able to share data with each other.

Sequencers

Sequencing software provides a number of benefits to a composer or arranger. It allows recorded MIDI to be manipulated using standard computer editing features such as cut, copy and paste and drag and drop. Keyboard shortcuts can be used to streamline workflow, and editing functions are often selectable via MIDI commands. The sequencer allows each channel to be set to play a different sound, and gives a graphical overview of the arrangement. A variety of editing tools are made available, including a notation display that can be used to create printed parts for musicians. Tools such as looping, quantization, randomization, and transposition simplify the arranging process. Beat creation is simplified, and groove templates can be used to duplicate another track's rhythmic feel. Realistic expression can be added through the manipulation of real-time controllers. Mixing can be performed, and MIDI can be synchronized with recorded audio and video tracks. Work can be saved, and transported between different computers or studios.

Sequencers may take alternate forms, such as drum pattern editors that allow users to create beats by clicking on pattern grids, and loop sequencers such as ACID Pro, which allow MIDI to be combined with prerecorded audio loops whose tempos and keys are matched to each other. Cue list sequencing is used to trigger dialogue, sound effect, and music cues in stage and broadcast production.

Notation/scoring software

With MIDI, notes played on a keyboard can automatically be transcribed to sheet music. Scorewriting, or notation software typically lacks advanced sequencing tools, and is optimized for the creation of a neat, professional printout designed for live instrumentalists. These programs provide support for dynamics and expression markings, chord and lyric display, and complex score styles. Software is available that can print scores in braille. Musitek's SmartScore (formerly MIDIScan) software performs the reverse process, and can produce MIDI files from scanned sheet music. Prominent notation programs include Finale, published by MakeMusic, and Encore, originally published by Passport Designs Inc., but now by GVOX. Sibelius, originally created for RISC-based Acorn computers, was so well-regarded that, before Windows and Macintosh versions were available, composers would purchase Acorns for the sole purpose of using Sibelius.

Editor/librarians

Patch editors allow users to program their equipment through the computer interface. These became essential with the appearance of complex synthesizers such as the Yamaha FS1R, which contained several thousand programmable parameters, but had an interface that consisted of fifteen tiny buttons, four knobs and a small LCD. Digital instruments typically discourage users from experimentation, due to their lack of the feedback and direct control that switches and knobs would provide, but patch editors give owners of hardware instruments and effects devices the same editing functionality that is available to users of software synthesizers. Some editors are designed for a specific instrument or effects device, while other, "universal" editors support a variety of equipment, and ideally can control the parameters of every device in a setup through the use of System Exclusive commands.

Patch librarians have the specialized function of organizing the sounds in a collection of equipment, and allow transmission of entire banks of sounds between an instrument and a computer. This allows the user to augment the device's limited patch storage with a computer's much greater disk capacity, and to share custom patches with other owners of the same instrument. Universal editor/librarians that combine the two functions were once common, and included Opcode Systems' Galaxy and eMagic's SoundDiver. These programs have been largely abandoned with the trend toward computer-based synthesis, although Mark of the

Unicorn's (MOTU)'s Unisyn and Sound Quest's Midi Quest remain available. Native Instruments' Kore was an effort to bring the editor/librarian concept into the age of software instruments.

Auto-accompaniment programs

Programs that can dynamically generate accompaniment tracks are called "auto-accompaniment" programs. These create a full band arrangement in a style that the user selects, and send the result to a MIDI sound generating device for playback. The generated tracks can be used as educational or practice tools, as accompaniment for live performances, or as a songwriting aid. Examples include Band-in-a-Box, which originated on the Atari platform in the 1980s, One Man Band, Busker, MiBAC Jazz, SoundTrek JAMMER and DigiBand.

Synthesis and sampling

Computers can use software to generate sounds, which are then passed through a digital-to-analog convertor (DAC) to a loudspeaker system. Polyphony, the number of sounds that can be played simultaneously, is dependent on the power of the computer's Central Processing Unit, as are the sample rate and bit depth of playback, which directly affect the quality of the sound. Synthesizers implemented in software are subject to timing issues that are not present with hardware instruments, whose dedicated operating systems are not subject to interruption from background tasks as desktop operating systems are. These timing issues can cause distortion as recorded tracks lose synchronization with each other, and clicks and pops when sample playback is interrupted. Software synthesizers also exhibit a noticeable delay in their sound generation, because computers use an audio buffer that delays playback and disrupts MIDI timing.

Software synthesis' roots go back as far as the 1950s, when Max Mathews of Bell Labs wrote the MUSIC-N programming language, which was capable of non-real-time sound generation. The first synthesizer to run directly on a host computer's CPU was Reality, by Dave Smith's Seer Systems, which achieved a low latency through tight driver integration, and therefore could run only on Creative Labs soundcards. Some systems use dedicated hardware to reduce the load on the host CPU, as with Symbolic Sound Corporation's Kyma System, and the Creamware/Sonic Core Pulsar/SCOPE systems, which used several DSP chips hosted on a PCI card to power an entire studio's worth of instruments, effects, and mixers.

The ability to construct full MIDI arrangements entirely in computer software allows a composer to render a finalized result directly as an audio file.

Game music

Early PC games were distributed on floppy disks, and the small size of MIDI files made them a viable means of providing soundtracks. Games of the DOS and early Windows eras typically required compatibility with either Ad Lib or SoundBlaster audio cards. These cards used FM synthesis, which generates sound through modulation of sine waves. John Chowning, the technique's pioneer, theorized that the technology would be capable of accurate recreation of any sound if enough sine waves were used, but budget computer audio cards performed FM synthesis with only two sine waves. Combined with the cards' 8-bit audio, this resulted in a sound described as "artificial" and "primitive". Wavetable daughterboards that were later available provided audio samples that could be used in place of the FM sound. These were expensive, but often used the sounds from respected MIDI instruments such as the E-mu Proteus. The computer industry moved in the mid-1990s toward wavetable-based soundcards with 16-bit playback, but standardized on a 2MB ROM, a space too small in which to fit good-quality samples of 128 instruments plus drum kits. Some manufacturers used 12-bit samples, and padded those to 16 bits.

Other applications

MIDI has been adopted as a control protocol in a number of non-musical applications. MIDI Show Control uses MIDI commands to direct stage lighting systems and to trigger cued events in theatrical productions. VJs and turntablists use it to cue clips, and to synchronize equipment, and recording systems use it for synchronization and automation. Apple Motion allows control of animation parameters through MIDI. The 1987 first-person shooter game *MIDI Maze* and the 1990 Atari ST computer puzzle game *Oxyd* used MIDI to network computers together, and kits are available that allow MIDI control over home lighting and appliances.

MIDI is usually associated with music devices however MIDI can control any device that can read and process a MIDI command. It is therefore possible to send a spacecraft from earth to another destination in space, control home lighting, heating and air conditioning and even sequence traffic light signals all through MIDI commands. The receiving device or object would require a General MIDI processor, however in this instance, the program changes would trigger a function on that device rather than notes from MIDI instrument. Each function can be set to a timer (also controlled by MIDI) or other condition determined by the devices creator.

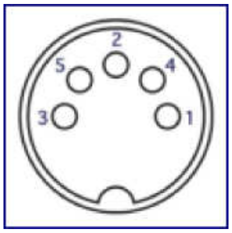
MIDI devices

Connectors

MIDI connectors and a MIDI cable.

The cables terminate in a 180°five-pin DIN connector . Standard applications use only three of the five conductors: a ground wire, and a balanced pair of conductors that carry a +5 volt signal. This connector configuration can only carry messages in one direction, so a second cable is necessary for two-way communication. Some proprietary applications, such as phantom-powered footswitch controllers, use the spare pins for direct current (DC) power transmission.

Opto-isolators keep MIDI devices electrically separated from their connectors, which prevents the occurrence of ground loops and protects equipment from voltage spikes. There is no error detection capability in MIDI, so the maximum cable length is set at 15 meters (50 feet) in order to limit interference.



A MIDI connector, showing the pins as numbered.

Most devices do not copy messages from their input to their output port. A third type of port, the "thru" port, emits a copy of everything received at the input port, allowing data to be forwarded to another instrument in a "daisy chain" arrangement. Not all devices contain thru ports, and devices that lack the ability to generate MIDI data, such as effects units and sound modules, may not include out ports.

Management devices

Each device in a daisy chain adds delay to the system. This is avoided with a MIDI thru box, which contains several outputs that provide an exact copy of the box's input signal. A MIDI merger is able to combine the input from multiple devices into a single stream, and allows multiple controllers to be connected to a single device. A MIDI switcher allows switching between multiple devices, and eliminates the need to physically repatch cables. MIDI patch bays combine all of these functions. They contain multiple inputs and outputs, and allow any combination of input channels to be routed to any combination of output channels. Routing setups can be created using computer software, stored in memory, and selected by MIDI program change commands. This enables the devices to function as standalone MIDI routers in situations where no computer is present. MIDI patch bays also clean up any skewing of MIDI data bits that occurs at the input stage.

MIDI data processors are used for utility tasks and special effects. These include MIDI filters, which remove unwanted MIDI data from the stream, and MIDI delays, effects which send a repeated copy of the input data at a set time.

Interfaces

A computer MIDI interface's main function is to match clock speeds between the MIDI device and the computer. Some computer sound cards include a standard MIDI connector, whereas others connect by any of various means that include the D-subminiature DA-15 game port, USB, FireWire, Ethernet or a proprietary connection. The increasing use of USB connectors in the 2000s has led to the availability of MIDI-to-USB data interfaces that can transfer MIDI channels to USB-equipped computers. Some MIDI keyboard controllers are equipped with USB jacks, and can be plugged into computers that run music software.

MIDI's serial transmission leads to timing problems. Experienced musicians can detect time differences of as small as 1/3 of a millisecond (ms) (which is how long it takes sound to travel 4 inches), and a three-byte MIDI message requires nearly 1ms for transmission. Because MIDI is serial, it can only send one event at a time. If an event is sent on two channels at once, the event on the higher-numbered channel cannot transmit until the first one is finished, and so is delayed by 1ms. If an event is sent on all channels at the same time, the highest-numbered channel's transmission will be delayed by as much as 16ms. This contributed to the rise of MIDI interfaces with multiple in- and out-ports, because timing improves when events are spread between multiple ports as opposed to multiple channels on the same port. The term "MIDI slop" refers to audible timing errors that result when MIDI transmission is delayed.

Controllers

There are two types of MIDI controllers: performance controllers that generate notes and are used to perform music, and controllers which may not send notes, but transmit other types of real-time events. Many devices are some combination of the two types.

Performance controllers

Two-octave MIDI controllers are popular for use with laptop computers, due to their portability.

MIDI

This unit provides a variety of real-time controllers, which can manipulate various sound design parameters of computer-based or standalone hardware instruments, effects, mixers and recording devices.

MIDI was designed with keyboards in mind, and any controller that is not a keyboard is considered an "alternative" controller. This was seen as a limitation by composers who were not interested in keyboard-based music, but the standard proved flexible, and MIDI compatibility was introduced to other types of controllers, including wind instruments and drum machines.

Keyboards

Keyboards are by far the most common type of MIDI controller. These are available in sizes that range from 25-key, 2-octave models, to full-sized 88-key instruments. Some are keyboard-only controllers, though many include other real-time controllers such as sliders, knobs, and wheels. Commonly, there are also connections for sustain and expression pedals. Most keyboard controllers offer the ability to split the playing area into "zones", which can be of any desired size and can overlap with each other. Each zone can respond to a different MIDI channel and a different set of performance controllers, and can be set to play any desired range of notes. This allows a single playing surface to target a number of different devices. MIDI capabilities can also be built into traditional keyboard instruments, such as grand pianos and Rhodes pianos. Pedal keyboards can operate the pedal tones of a MIDI organ, or can drive a bass synthesizer such as the revived Moog Taurus.

MIDI wind controllers can produce expressive, natural-sounding performances in a way that is difficult to achieve with keyboard controllers.

Wind controllers

Wind controllers allow MIDI parts to be played with the same kind of expression and articulation that is available to players of wind and brass instruments. They allow breath and pitch glide control that provide a more human and realistic kind of phrasing, particularly when playing sampled or physically modeled wind instrument parts. A typical wind controller has a sensor that converts breath pressure to volume information, and may allow pitch control through a lip pressure sensor and a pitch-bend wheel. Some models include a configurable key layout that can emulate different instruments' fingering systems. Examples of such controllers include Akai's Electronic Wind Instrument (EWI) and Electronic Valve Instrument (EVI). The EWI uses a system of keypads and rollers modeled after a traditional woodwind instrument, while the EVI is based on an acoustic brass instrument, and has three switches that emulate a trumpet's valves.

Drum and percussion controllers

Drum controllers, such as the Roland V-Drums, are often built in the form of an actual drum kit. The unit's sound module is mounted to the left.

Keyboards can be used to trigger drum sounds, but are impractical for playing repeated patterns such as rolls, due to the length of key travel. After keyboards, drum pads are the next most significant MIDI performance controllers. Drum controllers may be built into drum machines, may be standalone control surfaces, or may emulate the look and feel of acoustic percussion instruments. The pads built into drum machines are typically too small and fragile to be played with sticks, and are played with fingers. Dedicated drum pads such as the Roland Octapad or the DrumKAT are playable with the hands or with sticks, and are often built in the form of a drum kit. There are also percussion controllers such as the vibraphone-style MalletKAT, and Don Buchla's Marimba Lumina. MIDI triggers can also be installed into acoustic drum and percussion instruments. Pads that can trigger a MIDI device can be homemade from a piezoelectric sensor and a practice pad or other piece of foam rubber.

Stringed instrument controllers

A guitar can be fit with special pickups that digitize the instrument's output, and allow it to play a synthesizer's sounds. These assign a separate MIDI channel for each string, and may give the player the choice of triggering the same sound from all six strings, or playing a different sound from each. Some models, such as Yamaha's G10, dispense with the traditional guitar body and replace it with electronics. Other systems, such as Roland's MIDI pickups, are included with or can be retrofitted to a standard instrument. Max Mathews designed a MIDI violin for Laurie Anderson in the mid-1980s, and MIDI-equipped violas, cellos, contrabasses, and mandolins also exist.

A MIDI controller designed for use with an iPhone. The phone docks in the center.

Specialized performance controllers

DJ digital controllers may be standalone units such as the Faderfox or the Allen & Heath Xone 3D, or may be integrated with a specific piece of software, such as Traktor or Scratch Live. These typically respond to MIDI clock sync, and provide control over mixing, looping, effects, and sample playback.

MIDI triggers attached to shoes or clothing are sometimes used by stage performers. The Kroonde Gamma wireless sensor can capture physical motion as MIDI signals. Sensors built into a dance floor at the University of Texas at Austin convert dancers' movements into MIDI messages, and David Rokeby's *Very Nervous System* art installation created music from the movements of passers-through. Software applications exist which enable the use of iOS devices as gesture controllers.

Numerous experimental controllers exist which abandon traditional musical interfaces entirely. These include the gesture-controlled Buchla Thunder, sonomes such as the C-Thru Music Axis, which rearrange the scale tones into an isometric layout, and Haken Audio's keyless, touch-sensitive Continuum playing surface. Experimental MIDI controllers may be created from unusual objects, such as an ironing board with heat sensors installed, or a sofa equipped with pressure sensors.

Auxiliary controllers

Software synthesizers offer great power and versatility, but some players feel that division of attention between a MIDI keyboard and a computer keyboard and mouse robs some of the immediacy from the playing experience. Devices dedicated to real-time MIDI control provide an ergonomic benefit, and can provide a greater sense of connection with the instrument than can an interface that is accessed through a mouse or a push-button digital menu. Controllers may be general-purpose devices that are designed to work with a variety of equipment, or they may be designed to work with a specific piece of software. Examples of the latter include Akai's APC40 controller for Ableton Live, and Korg's MS-20ic controller that is a reproduction of their MS-20 analog synthesizer. The MS-20ic controller includes patch cables that can be used to control signal routing in their virtual reproduction of the MS-20 synthesizer, and can also control third-party devices.

Control surfaces

Control surfaces are hardware devices that provide a variety of controls that transmit real-time controller messages. These enable software instruments to be programmed without the discomfort of excessive mouse movements, or adjustment of hardware devices without the need to step through layered menus. Buttons, sliders, and knobs are the most common controllers provided, but rotary encoders, transport controls, joysticks, ribbon controllers, vector touchpads in the style of Korg's Kaoss pad, and optical controllers such as Roland's D-Beam may also be present. Control surfaces may be used for mixing, sequencer automation, turntablism, and lighting control.

Specialized real-time controllers

Audio control surfaces often resemble mixing consoles in appearance, and enable a level of hands-on control for changing parameters such as sound levels and effects applied to individual tracks of a multitrack recording or live performance output.

MIDI footswitches are commonly used to send MIDI program change commands to effects devices, but may be combined with pedals in a pedalboard that allows detailed programming of effects units. Pedals are available in the form of on/off switches, either momentary or latching, or as "rocker" pedals whose position determines the value of a MIDI continuous controller.

Drawbar controllers are for use with MIDI and virtual organs. Along with a set of drawbars for timbre control, they may provide controls for standard organ effects such as rotating speaker speed, vibrato and chorus.

Instruments

A sound module, which requires an external controller to trigger its sounds. These devices are highly portable, but their limited programming interface requires computer-based tools for comfortable access to their sound parameters.

A MIDI instrument contains ports to send and receive MIDI signals, a CPU to process those signals, an interface that allows user programming, audio circuitry to generate sound, and controllers. The operating system and factory sounds are often stored in a Read-only memory (ROM) unit.

A MIDI Instrument can also be a stand alone module (without a piano style keyboard) consisting of a General MIDI soundboard (GM, GS and/XG), onboard editing, including transposing/pitch changes, MIDI instrument changes and adjusting volume, pan, reverb levels and other MIDI controllers. Typically, the MIDI Module will include a large screen, enabling the user to view information depending on the function selected at that time. Features can include scrolling lyrics, usually embedded in a MIDI File or Karaoke MIDI, playlists, song library and editing screens. Some MIDI Modules include a Harmonizer and the ability to playback and transpose MP3 audio files.

Synthesizers

Synthesizers may employ any of a variety of sound generation techniques. They may include an integrated keyboard, or may exist as "sound modules" or "expanders" that generate sounds when triggered by an external controller. Sound modules are typically designed to be mounted in a 19-inch rack. Manufacturers commonly produce a synthesizer in both standalone and rack-mounted versions, and often offer the keyboard version in a variety of sizes.

Samplers

A sampler can record and digitize audio, store it in random-access memory (RAM), and play it back. Samplers typically allow a user to edit a sample and save it to a hard disk, apply effects to it, and shape it with the same tools that synthesizers use. They also may be available in either keyboard or rack-mounted form. Instruments that generate sounds through sample playback, but have no recording capabilities, are known as "ROMplers".

Samplers did not become established as viable MIDI instruments as quickly as synthesizers did, due to the expense of memory and processing power at the time. The first low-cost MIDI sampler was the Ensoniq Mirage, introduced in 1984. MIDI samplers are typically limited by displays that are too small to use to edit sampled waveforms, although some can be connected to a computer monitor.

Drum machines

Drum machines typically are sample playback devices that specialize in drum and percussion sounds. They commonly contain a sequencer that allows the creation of drum patterns, and allows them to be arranged into a song. There often are multiple audio outputs, so that each sound or group of sounds can be routed to a separate output. The individual drum voices may be playable from another MIDI instrument, or from a sequencer.

Workstations and hardware sequencers

Yamaha's Tenori-on controller allows arrangements to be built by "drawing" on its array of lighted buttons. The resulting arrangements can be played back using its internal sounds or external sound sources, or recorded in a computer-based sequencer.

Sequencer technology predates MIDI. Analog sequencers use CV/Gate signals to control pre-MIDI analog synthesizers. MIDI sequencers typically are operated by transport features modeled after those of tape decks. They are capable of recording MIDI performances, and arranging them into individual tracks along a multitrack recording concept. Music workstations combine controller keyboards with an internal sound generator and a sequencer. These can be used to build complete arrangements and play them back using their own internal sounds, and function as self-contained music production studios. They commonly include file storage and transfer capabilities.

Effects devices

Audio effects units that are frequently used in stage and recording, such as reverbs, delays and choruses, can be remotely adjusted via MIDI signals. Some units allow only a limited number of parameters to be controlled this way, but most will respond to program change messages. The Eventide H3000 Ultra-harmonizer is an example of a unit that allows such extensive MIDI control that it is playable as a synthesizer.

Technical specifications

MIDI messages are made up of 8-bit *word*s (commonly called *bytes*) that are transmitted serially at 31.25 kbaud. This rate was chosen because it is an exact division of 1 MHz, the speed at which many early microprocessors operated. The first bit of each word identifies whether the word is a status byte or a data byte, and is followed by seven bits of information. A start bit and a stop bit are added to each byte for framing purposes, so a MIDI byte requires ten bits for transmission.

A MIDI link can carry sixteen independent channels of information. The channels are numbered 1-16, but their actual corresponding binary encoding is 0-15. A device can be configured to only listen to specific channels and to ignore the messages sent on other channels (“Omni Off” mode), or it can listen to all channels, effectively ignoring the channel address (“Omni On”). An individual device may be monophonic (the start of a new “note-on” MIDI command implies the termination of the previous note), or polyphonic (multiple notes may be sounding at once, until the polyphony limit of the instrument is reached, or the notes reach the end of their decay envelope, or explicit “note-off” MIDI commands are received). Receiving devices can typically be set to all four combinations of “omni off/on” versus “mono/poly” modes.

Messages

A MIDI message is an instruction that controls some aspect of the receiving device. A MIDI message consists of a status byte, which indicates the type of the message, followed by up to two data bytes that contain the parameters. MIDI messages can be "channel messages", which are sent on only one of the 16 channels and can be heard only by devices receiving on that channel, or "system messages", which are heard by all devices. Any data not relevant to a receiving device is ignored. There are five types of message: Channel Voice, Channel Mode, System Common, System Real-Time, and System Exclusive.

Channel Voice messages transmit real-time performance data over a single channel. Examples include "note-on" messages which contain a MIDI note number that specifies the note's pitch, a velocity value that indicates how forcefully the note was played, and the channel number; "note-off" messages that end a note; program change messages that change a device's patch; and control changes that allow adjustment of an instrument's parameters. Channel Mode messages include the Omni/mono/poly mode on and off messages, as well as messages to reset all controllers to their default state or to send "note-off" messages for all notes. System messages do not include channel numbers, and are received by every device in the MIDI chain. MIDI time code is an example of a System Common message. System Real-Time messages provide for synchronization, and include MIDI clock and Active Sensing.[2]:18–35

MIDI

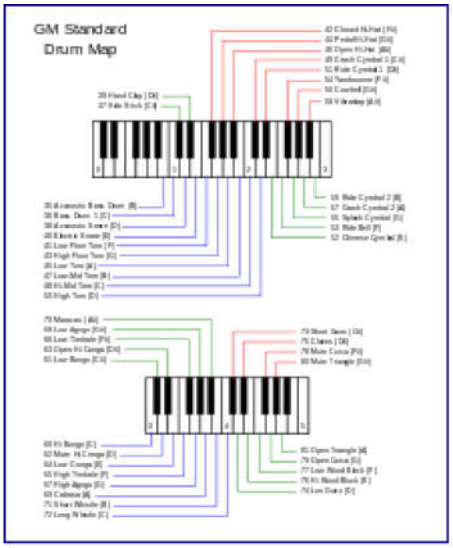
System Exclusive messages

System Exclusive (SysEx) messages are a major reason for the flexibility and longevity of the MIDI standard. They allow manufacturers to create proprietary messages which provide control over their equipment in a way that is more thorough than is provided for by standard MIDI messages. SysEx messages are addressed to a specific device in a system. Each manufacturer has a unique identifier that is included in its SysEx messages, which helps ensure that the messages will be heard only by the targeted device, and ignored by all others. Many instruments also include a SysEx ID setting, which allows two devices of the same model to be addressed independently while connected to the same system. SysEx messages may include functionality beyond what the MIDI standard provides. They are targeted at a specific instrument, and are ignored by all other devices on the system.

The MIDI Implementation Chart

Devices typically do not respond to every type of message defined by the MIDI specification. The MIDI Implementation Chart was standardized by the MMA as a way for users to see what specific capabilities an instrument has, and how it responds to messages. A specific MIDI Implementation Chart is published for each MIDI device.

Extensions



The GM Standard Drum Map, which specifies the percussion sound that a given note will trigger.

MIDI's flexibility and widespread adoption have led to many refinements of the standard, and have enabled its application to purposes beyond those for which it was originally intended.

MIDI

General MIDI

MIDI allows selection of an instrument's sounds through program change messages, but there is no guarantee that any two instruments have the same sound at a given program location. Program #0 may be a piano on one instrument, or a flute on another. The General MIDI (GM) standard was established in 1991, and provides a standardized sound bank that allows a Standard MIDI File created on one device to sound similar when played back on another. GM specifies a bank of 128 sounds arranged into 16 families of eight related instruments, and assigns a specific program number to each instrument. Percussion instruments are placed on channel 10, and a specific MIDI note value is mapped to each percussion sound. GM-compliant devices must offer 24-note polyphony. Any given program change will select the same instrument sound on any GM-compatible instrument.

The GM standard eliminates variation in note mapping. Some manufacturers had disagreed over what note number should represent middle C, but GM specifies that note number 69 plays A440, which in turn fixes middle C as note number 60. GM-compatible devices are required to respond to velocity, aftertouch, and pitch bend, to be set to specified default values at startup, and to support certain controller numbers such as for sustain pedal, and Registered Parameter Numbers. A simplified version of GM, called "GM Lite", is used in mobile phones and other devices with limited processing power.

GS, XG, and GM2

A general opinion quickly formed that the GM's 128-instrument sound set was not large enough. Roland's General Standard, or GS, system included additional sounds, drumkits and effects, provided a "bank select" command that could be used to access them, and used MIDI Non-Registered Parameter Numbers (NRPNs) to access its new features. Yamaha's Extended General MIDI, or XG, followed in 1994. XG similarly offered extra sounds, drumkits and effects, but used standard controllers instead of NRPNs for editing, and increased polyphony to 32 voices. Both standards feature backward compatibility with the GM specification, but are not compatible with each other. Neither standard has been adopted beyond its creator, but both are commonly supported by music software titles.

Member companies of Japan's AMEI developed the General MIDI Level 2 specification in 1999. GM2 maintains backward compatibility with GM, but increases polyphony to 32 voices, standardizes several controller numbers such as for sostenuto and soft pedal (*una corda*), RPNs and Universal System Exclusive Messages, and incorporates the MIDI Tuning Standard. GM2 is the basis of the instrument selection mechanism in Scalable Polyphony MIDI (SP-MIDI), a MIDI variant for low power devices that allows the device's polyphony to scale according to its processing power.

MIDI Tuning Standard

Most MIDI synthesizers use equal temperament tuning. The MIDI Tuning Standard (MTS),ratified in 1992, allows alternate tunings. MTS allows microtunings that can be loaded from a bank of up to 128 patches, and allows real-time adjustment of note pitches. Manufacturers are not required to support the standard. Those who do are not required to implement all of its features.

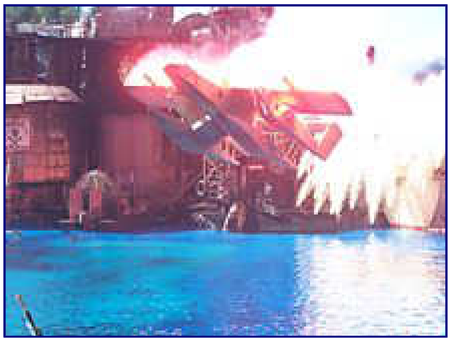
MIDI Time Code

A sequencer can drive a MIDI system with its internal clock, but when a system contains multiple sequencers, they must synchronize to a common clock. MIDI Time Code (MTC), developed by Digidesign, implements SysEx messages that have been developed specifically for timing purposes, and is able to translate to and from the SMPTE time code standard. MIDI Clock is based on tempo, but SMPTE time code is based on frames per second, and is independent of tempo. MTC, like SMPTE code, includes position information, and can adjust itself if a timing pulse is lost. MIDI interfaces such as Mark of the Unicorn's MIDI Timepiece can convert SMPTE code to MTC.

MIDI Machine Control

MIDI Machine Control (MMC) consists of a set of SysEx commands that operate the transport controls of hardware recording devices. MMC allows a sequencer to send "Start", "Stop", and "Record" commands to a connected tape deck or hard disk recording system, and to fast-forward or rewind the device so that it starts playback at the same point as the sequencer. No synchronization data is involved, although the devices may synchronize through MTC.

MIDI Show Control



MIDI Show Control is used to cue and synchronize lighting and effects for theatrical events, such as the Waterworld attraction at Universal Studios Hollywood.

MIDI Show Control (MSC) is a set of SysEx commands which allows sequencing and remote cueing of show control devices such as lighting, music and sound playback, and motion control systems. Applications include stage productions, museum exhibits, recording studio control systems, and amusement park attractions.

MIDI timestamping

One solution to MIDI timing problems is to mark MIDI events with the times they are to be played, and store them in a buffer in the MIDI interface ahead of time. Sending data beforehand reduces the likelihood that a busy passage will send a large amount of information that will overwhelm the transmission link. Once stored in the interface, the information is no longer subject to timing issues associated with USB jitter and computer operating system interrupts, and can be transmitted with a high degree of accuracy. MIDI timestamping only works when both the hardware and software support it. MOTU's MTS, eMagic's AMT, and Steinberg's Midex 8 were implementations that were incompatible with each other, and required users to own software and hardware manufactured by the same company in order to gain its benefits. Timestamping is built into FireWire MIDI interfaces and Mac OS X Core Audio.

MIDI Sample Dump Standard

An unforeseen capability of SysEx messages was their use for transporting audio samples between instruments. SysEx is poorly suited to this purpose, as MIDI words are limited to seven bits of information, so an 8-bit sample requires two bytes for transmission instead of one. This led to the development of the Sample Dump Standard (SDS), which established a protocol for sample transmission. The SDS was later augmented with a pair of commands that allow the transmission of information about sample loop points, without requiring that the entire sample be transmitted.

Downloadable Sounds

The Downloadable Sounds (DLS) specification, ratified in 1997, allows mobile devices and computer sound cards to expand their wave tables with downloadable sound sets. The DLS Level 2 Specification followed in 2006, and defined a standardized synthesizer architecture. The Mobile DLS standard calls for DLS banks to be combined with SP-MIDI, as self-contained Mobile XMF files.

Alternative hardware transports

In addition to the original 31.25 kbit/s current-loop transported on 5-pin DIN, other connectors have been used for the same electrical data, and transmission of MIDI streams in different forms over USB, IEEE 1394 a.k.a. FireWire, and Ethernet is now common. Some samplers and hard drive recorders can also pass MIDI data between each other over SCSI.

USB and FireWire

Members of the USB-IF in 1999 developed a standard for MIDI over USB, the "Universal Serial Bus Device Class Definition for MIDI Devices" MIDI over USB has become increasingly common as other interfaces that had been used for MIDI connections (serial, joystick, etc.) disappeared from personal computers. Microsoft Windows, Macintosh OSX, and Apple iOS operating systems include standard class drivers to support devices that use the "Universal Serial Bus Device Class Definition for MIDI Devices". Drivers are also available for Linux. Some manufacturers choose to implement a MIDI interface over USB that is designed to operate differently from the class specification, using custom drivers.

Apple Computer developed the FireWire interface during the 1990s. It began to appear on digital video cameras toward the end of the decade, and on G3 Macintosh models in 1999. It was created for use with multimedia applications. Unlike USB, FireWire uses intelligent controllers that can manage their own transmission without attention from the main CPU. As with standard MIDI devices, FireWire devices can communicate with each other with no computer present.

XLR connectors

The Octave-Plateau Voyetra-8 synthesizer was an early MIDI implementation using XLR3 connectors in place of the 5-pin DIN. It was released in the pre-MIDI years and later retrofitted with a MIDI interface but keeping its XLR connector.

Serial parallel, and joystick port MIDI

As computer-based studio setups became common, MIDI devices that could connect directly to a computer became available. These typically used the 8-pin mini-DIN connector that was used by Apple for serial and printer ports prior to the introduction of the Blue & White G3 models. MIDI interfaces intended for use as the centerpiece of a studio, such as the Mark of the Unicorn MIDI Time Piece, were made possible by a "fast" transmission mode that could take advantage of these serial ports' ability to operate at 20 times the standard MIDI speed. Mini-DIN ports were built into some late-1990s MIDI instruments, and enabled such devices to be connected directly to a computer. Some devices connected via PCs' DB-25 parallel port, or through the joystick port found in many PC sound cards.

mLAN

Yamaha introduced the mLAN protocol in 1999. It was conceived as a Local Area Network for musical instruments using FireWire as the transport, and was designed to carry multiple MIDI channels together with multichannel digital audio, data file transfers, and time code. mLan was used in a number of Yamaha products, notably digital mixing consoles and the Motif synthesizer, and in third-party products such as the PreSonus FIREstation and the Korg Triton Studio. No new mLan products have been released since 2007.

Ethernet

The computer network implementation of MIDI provides network routing capabilities, and provides the high-bandwidth channel that earlier alternatives to MIDI, such as ZIPI, were intended to bring. Proprietary implementations have existed since the 1980s, some of which use fiber optic cables for transmission. The Internet Engineering Task Force's RTP MIDI open specification is gaining industry support, as proprietary MIDI/IP protocols require expensive licensing fees, or provide no advantage, apart from speed, over the original MIDI protocol. Apple has supported this protocol from Mac OS X 10.4 onwards, and a Windows driver based on Apple's implementation exists for Windows XP and newer versions.

OSC

The Open Sound Control (OSC) protocol was developed at the Center for New Music and Audio Technologies (CNMAT) of the University of California, Berkeley, and is supported by software programs such as Reaktor, Max/MSP and Csound, and some newer hardware controllers, including the Lemur Input Device. OSC can be transported over Ethernet connections at broadband speeds, but is poorly suited for use as a whole-studio solution, as to date it lacks widespread support from hardware and mainstream software. The increased size of OSC messages as compared to MIDI messages make OSC an impractical solution for many mobile devices, and its touted speed advantages over MIDI disappear when both are transmitted over equal media. OSC is not proprietary, but is not maintained by a standards organization.

Wireless MIDI

Systems for wireless MIDI transmission have been available since the 1980s. Several commercially available transmitters allow wireless transmission of MIDI and OSC signals over Wi-Fi and Bluetooth. iOS devices are able to function as MIDI control surfaces, using Wi-Fi and OSC. An XBee radio can be used to build a wireless MIDI transceiver as a do-it-yourself project. Android devices are able to function as full MIDI control surfaces using several different protocols over Wi-Fi and Bluetooth.

The future of MIDI

A new version of MIDI tentatively called "HD Protocol" or "High-Definition Protocol" has been under discussion since 2005, when it was announced as "HD-MIDI". This new standard offers full backward compatibility with MIDI 1.0 and is intended to support higher-speed transports, allow plug-and-play device discovery and enumeration, and provide greater data range and resolution. The numbers of channels and controllers are to be increased, new kinds of events are to be added, and messages are to be simplified. Entirely new kinds of events will be supported, such as a Note Update message and Direct Pitch in the Note message which are aimed at guitar controllers. Proposed physical layer transports include Ethernet-based protocols such as RTP MIDI and Audio Video Bridging. The HD Protocol and a User Datagram Protocol (UDP)-based transport are under review by MMA's High-Definition Protocol Working Group (HDWG), which includes representatives from all sizes and types of companies. Prototype devices based on the draft standard were shown privately at Winter NAMM 2013 using wired and wireless connections. It is currently not known if and when the new protocol will be picked up by the industry. Because the cost of data storage in consumer products has dropped so much, MIDI music is more and more replaced by wave audio in computers, tablets and phones. MIDI connectivity and a software synthesizer is still included in Windows, OSX and iOS but not in Android.

References

1. Swift, Andrew. (May-Jun 1997.), "A brief Introduction to MIDI", *SURPRISE* (Imperial College of Science Technology and Medicine), retrieved 22 August 2012
2. Huber, David Miles. "The MIDI Manual". Carmel, Indiana: SAMS, 1991.
3. http://www.instructables.com/id/What-is-MIDI/
4. Holmes, Thom. *Electronic and Experimental Music: Pioneers in Technology and Composition*. New York: Routledge, 2003
5. Manning, Peter. *Electronic and Computer Music*. 1985. Oxford: Oxford University Press, 1994. Print.
6. Chadabe, Joel (1 May 2000). "Part IV: The Seeds of the Future". *Electronic Musician* (Penton Media) **XVI** (5).
7. Paul, Craner (Oct 1991). "New Tool for an Ancient Art: The Computer and Music". *Computers and the Humanities* **25** (5): 308–309. JSTOR 30204425.
8. Macan, Edward. *Rocking the Classics: English Progressive Rock and the Counterculture*. New York: Oxford University Press, 1997. p.191
9. Shuker, Roy. *Understanding Popular Music*. London: Routledge, 1994. p.286
10. Demorest, Steven M. *Building Choral Excellence: Teaching Sight-Singing in the Choral Rehearsal*. New York: Oxford University Press, 2003. p. 17
11. Pertout, Andrian. *Mixdown Monthly*, #26. 26 June 1996. Web. 22 August 2012
12. Crawford, Renee. *An Australian Perspective: Technology in Secondary School Music".* Journal of Historical Research in Music Education*. Vol. 30, No. 2. Apr 2009. Print.*
13. "IKUTARO KAKEHASHI & DAVE SMITH AND ROYER LABS TO RECEIVE TECHNICAL GRAMMY® AWARD". Retrieved 2012-12-20.
14. Lau, Paul. "Why Still MIDI?." – via HighBeam Research (subscription required) Canadian Musician. Norris-Whitney Communications Inc. 2008. HighBeam Research. 4 September 2012
15. Sasso, Len. "Sound Programming 101". *Electronic Musician*. NewBay Media. 1 October 2002. Web. 4 September 2012.
16. Anderton, Craig. "MIDI For Guitarists: A Crash Course In MIDI Effects Control". *Sound On Sound*. SOS Publications. May 1995.
17. McCutchan, Ann. *The Muse That Sings: Composers Speak about the Creative Process*. New York: Oxford University Press, 1999. p. 67-68,72
18. Brewster, Stephen. "Nonspeech Auditory Output". *The Human-Computer Interaction Handbook: Fundamentals, Evolving Technologies, and Emerging Applications*. Ed. Julie A. Jacko; Andrew Sears. Mahwah: Lawrence Erlbaum Associates, 2003. p.227
19. Campbell, Drew. ""Click, Click. Audio" *Stage Directions*. Vol. 16, No. 3. Mar 2003.
20. "Digital audio workstation – Intro". Archived from the original on 10 January 2012.
21. "the-digital-orchestra".
22. Hass, Jeffrey. "Chapter Three: How MIDI works 10". Indiana University Jacobs School of Music. 2010. Web 13 August 2012
23. "MIDI Files". *midi.org* Music Manufacturers Association. n.d. Web. 27 August 2012
24. "RIFF-based MIDI File Format". *digitalpreservation.gov*. Library of Congress. 26 March 2012. Web. 18 August 2012
25. Crawford, Walt. "MIDI and Wave: Coping with the Language". *Online*. Vol. 20, No. 1.

Jan/Feb 1996

26."MIDI Assoc. pushes for new licensing agreement. (MIDI Manufacturers Association)." Music Trades. Music Trades Corp. 1996. HighBeam Research. 4 September 2012 – via HighBeam Research (subscription required)

27.Wiffen, Paul. "Synth School, Part 3: Digital Synthesis (FM, PD & VPM)". *Sound on Sound* Sep 1997. Print.

28.Battino, David. "Finally: MIDI 2.0" O'Reilly Digital Media Blog. O'Reilly Media, Inc. 6 October 2005. Web. 22 August 2012

29.Gellerman, Elizabeth. "Audio Editing SW Is Music to Multimedia Developers' Ears". *Technical Horizons in Education Journal*. Vol. 22, No. 2. Sep 1994

30.Desmond, Peter. "ICT in the Secondary Music Curriculum". *Aspects of Teaching Secondary Music: Perspectives on Practice*. ed. Gary Spruce. New York: RoutledgeFalmer, 2002

31.Solomon, Karen. "You Gotta Feel the Music". *wired.com*. Condé Nast. 27 February 2000. Web. 13 August 2012.

32.Cook, Janet Harniman. "Musitek Midiscan v2.51". *Sound On Sound*. SOS Publications. Dec 1998. Print.

33."The Social Impact of Computers- One Man's Story". *computinghistory.org.uk*. n.p. n.d. 1 August 2012

34.Johnson, Derek. "Yamaha FS1R Editor Software". *Sound on Sound*. Mar 1999.

35.Johnson, Derek, and Debbie Poyser. "Yamaha FS1R". *Sound on Sound*. Dec 1998.

36.Gibbs, Jonathan (Rev. by Peter Howell) "Electronic Music". *Sound Recording Practice*, 4th Ed. Ed. John Borwick. Oxford: Oxford University Press, 1996

37."Sound Quest MIDI Quest 11 Universal Editor". *squest.com*. n.p. n.d. Web. 21 August 2012

38."Desktop Music Handbook – MIDI". *cakewalk.com*. Cakewalk, Inc. 26 November 2010. Web. Retrieved 7 August 2012.

39.Simon Price. "Price, Simon. "Native Instruments Kore". "Sound on Sound" Jul 06". Soundonsound.com. Retrieved 2012-11-27.

40.Bozeman, William C. *Educational Technology: Best Practices from America's Schools*. Larchmont: Eye on Education, 1999.

41."One Man Band v11 - Live-on-stage arranger"

42."Busker - Score editor and player"

43."Jammer Professional 6". *soundtrek.com*. SoundTrek. 11 August 2008. Web. Retrieved 7 August 2012.

44."AthTek DigiBand 1.4". *AthTek.com*. AthTek. 01 May 2013.

45.Lehrman, Paul D. "Software Synthesis: The Wave Of The Future?" *Sound On Sound*. SOS Publications. Oct 1995. Print.

46.Walker, Martin. "Identifying & Solving PC MIDI & Audio Timing Problems". *Sound On Sound*. SOS Publications. Mar 2001. Print.

47.Miller, Dennis. "Sound Synthesis On A Computer, Part 2". *Sound On Sound*. SOS Publications. May 1997. Print.

48."MIDI Ancestors and Milestones". *keyboardmag.com*. New Bay Media. n.d. Web. 6 August 2012.

49.Walker, Martin. "Reality PC". *Sound On Sound*. SOS Publications. Nov 1997. Print.

50.Wherry, Mark. "Creamware SCOPE". *Sound On Sound*. SOS Publications. Jun 2003.

Print.
51.Anderton, Craig. "Sonic Core SCOPE Xite-1". *keyboardmag.com*. New Bay Media, LLC. n.d. Web.
52.David Nicholson. "HARDWARE." – via HighBeam Research (subscription required) The Washington Post. Washingtonpost Newsweek Interactive. 1993. HighBeam Research. 4 September 2012
53.Levy, David S. "[http://www.highbeam.com/doc/1G1-14803399.html Aztech's WavePower daughtercard improves FM reception. (Aztech Labs Inc.'s wavetable synthesis add-on card for Sound Blaster 16 or Sound Galaxy Pro 16 sound cards) (Hardware Review) (Evaluation)." Computer Shopper. SX2 Media Labs LLC. 1994. HighBeam Research. 4 September 2012 – via HighBeam Research (subscription required)
54.Labriola, Don. "MIDI masters: wavetable synthesis brings sonic realism to inexpensive sound cards. (review of eight Musical Instrument Digital Interface sound cards) (includes related articles about testing methodology, pitfalls of wavetable technology, future wavetable developments) (Hardware Review) (Evaluation)." – via HighBeam Research (subscription required) Computer Shopper. SX2 Media Labs LLC. 1994. HighBeam Research. 4 September 2012
55."Interface Circuits". MIDI Kits. n.p. 30 August 2012. Web. 30 August 2012.
56.Lockwood, Dave. "TC Electronic G Major". *Sound On Sound*. SOS Publications. Dec 2001. Print.
57.Mornington-West, Allen. "Digital Theory". *Sound Recording Practice*. 4th Ed. Ed. John Borwick. Oxford: Oxford University Press, 1996.
58."Richmond Sound Design – Frequently Asked Questions". *richmondsounddesign.com*. Web. 5 August 2012.
59.Hass, Jeffrey. "Chapter Three: How MIDI works 2". Indiana University Jacobs School of Music. 2010. Web. 13 August 2012.
60.Robinson, Herbie. "Re: core midi time stamping". *Apple Coreaudio-api Mailing List*. Apple, Inc. 18 July 2005. 8 August 2012.
61.Shirak, Rob. "Mark of the Unicorn". *emusician.com*. New Bay Media. 1 October 2000. Web. Retrieved 8 August 2012.
62."MIDI Performance Instruments". *Instruments of Change*. Vol. 3, No. 1 (Winter 1999). Roland Corporation, U.S.
63."MIDI Products". *midi.org*. MIDI Manufacturers Association. n.d. 1 August 1012
64."The beginner's guide to: MIDI controllers". Computer Music Specials. Retrieved 11 July 2011.
65."Rhodes Mark 7". *keyboardmag.com*. New Bay Media, LLC. n.d. Web. Retrieved 7 August 2012.
66.White, Paul. "Yamaha WX5". *Sound On Sound*. SOS Publications. Jul 1998. Print.
67.""Marimba Lumina Described". "buchla.com". n.p. n.d. Web". Buchla.com. Retrieved 2012-11-27.
68.White, Paul. "DIY Drum Pads And Pedal Triggers". *Sound On Sound* SOS Publications. Aug 1995. Print.
69.Goldberg, Roselee. *Laurie Anderson*. New York: Abrams Books, 2000. p.80
70.Batcho, Jim. "Best of Both Worlds". *Strings* 17.4 (2002): n.a. Print.
71.Price, Simon. "Allen & Heath Xone 3D". *Sound On Sound*. SOS Publications. Sep 2006. Print.

72.. "kirstybeilharz.com". n.p. n.d. 13 Aug 2012 "Projects". Kirstybeilharz.com.au. 2010-04-30. Retrieved 2012-11-27.

73.Pinkston, Kerkhoff, & McQuilken.]"The U. T. Touch-Sensitive Dance Floor and MIDI Controller". The University of Texas at Austin. n.d. Web. 10 August 2012.

74.Cooper, Douglas. "Very Nervous System". *Wired*. Condé Nast. 3.03: Mar 1995.

75."The Glimpse". *midiinmotion.fschwehn.com*. n.p. n.d. Web. 20 August 2012

76.""Buchla Thunder". "buchla.com". Buchla and Associates. n.d. Web". Buchla.com. Retrieved 2012-11-27.

77."MIDI Products". *midi.org*. The MIDI Manufacturers Association. n.d. Web. 10 August 2012.

78.""Note pattern". "theshapeofmusic.com". n.p. n.d. Web. 10 Aug 2012". Theshapeofmusic.com. Retrieved 2012-11-27.

79.""Overview". "hakenaudio.com". Haken Audio. n.d. Web. 10 Aug 2012". Hakenaudio.com. Retrieved 2012-11-27.

80.Gamboa, Glenn. "MIDI Ironing Boards, Theremin Crutches Squeal at Handmade Music Event". *Wired.com*. Condé Nast. 27 September 2007. 13 August 2012. Web.

81."MIDI Products". *midi.org*. MIDI Manufacturers Association. n.d. 13 August 2012. Web.

82.Preve, Francis. "Dave Smith", in "The 1st Annual *Keyboard* Hall of Fame". *Keyboard* (US). NewBay Media, LLC. Sep 2012. Print. p.18

83."Korg Legacy Collection". *vintagesynth.com*. Vintage Synth Explorer. n.d. Web. 21 August 2012

84.Walker, Martin. "Controlling Influence". *Sound On Sound*. SOS Publications. Oct 2001. Print.

85.""Drawbar Manual v. 1.2". Voce, Inc. n.d. Web. 10 Aug 2012". Voceinc.com. Retrieved 2012-11-27.

86.Hass, Jeffrey. "Chapter Three: How MIDI works 3". Indiana University Jacobs School of Music. 2010. Web. 13 August 2012.

87.Hass, Jeffrey. "Chapter Three: How MIDI works 9". Indiana University Jacobs School of Music. 2010. Web. 13 August 2012.

88.Bello, Juan P. "MIDI: sound control". *nyu.edu*. New York University. n.d. Web. 18 August 2012

89.""General MIDI Standard". "www.harfesoft.de". n.p. n.d. Web". Harfesoft.de. Retrieved 2012-11-27.

90."General MIDI Standard". *pgcc.edu*. Prince George's Community College. n.d. Web.

91.Glatt, Jeff. "General MIDI". *The MIDI Technical Fanatic's Brainwashing Center*. n.p. n.d. Web. 17 August 2012

92.Nagle, Paul. "Yamaha MU50 & Yamaha CBX-K1". *Sound On Sound*. SOS Publications. Sep 1995. Print.

93."About General MIDI". *midi.org*. MIDI Manufacturers Association. n.d. Web. 17 August 2012

94."The MIDI Tuning Standard". *microtonal-synthesis.com*. n.p. n.d. Web. 17 August 2012

95."MIDI Tuning". *midi.org*. MIDI Manufacturers Association. n.d. Web. 17 August 2012

96.Glatt, Jeff. "The beginnings of MIDI"[http://home.roadrunner.com/~jgglatt/tech/mtc.htm . *The MIDI Technical Fanatic's Brainwashing Center*. n.p. n.d. Web. 13 August 2012.

97.Glatt, Jeff. "MIDI Time Code". *The MIDI Technical Fanatic's Brainwashing Center*. n.p. n.d. Web. 13 August 2012.

98.White, Paul. "SMPTE & MTC (MIDI Time Code)" *Sound On Sound*. SOS Publications. Jun 1996. Print.

99.""Q & A". "Sweet Notes". Sweetwater Sound. Summer 1996. Web". Sweetwater.com. Retrieved 2012-11-27.

100.Glatt, Jeff. "MIDI Machine Control (MMC)". *The MIDI Technical Fanatic's Brainwashing Center*. n.p. n.d. Web.

101."Glossary: MIDI Machine Control (MMC)". *sweetwater.com*. Sweetwater Sound. n.d. Web. 15 August 2012.

102."News Page". *richmondsounddesign.com*. Richmond Sound Design, Ltd. 17 July 2012. Web. 17 August 2012

103."An Inexpensive MIDI show-control System". *Lighting TechNotes*. The University of Virginia. 25 October 2004. Web. 17 August 2012.

104."Glossary: MTS (MIDI Time Stamping)". *sweetwater.com*. Sweetwater Sound. n.d. Web. 17 August 2012

105.Walker, Martin. "The Truth About Latency: Part 2". *Sound On Sound*. SOS Publications. Oct 2002. Print.

106.Glatt, Jeff. "Sample Dump Standard. *The MIDI Technical Fanatic's Brainwashing Center*. n.p. n.d. Web. 13 August 2012.

107."Massey, Howard. "DLS Overview". "midi.org". n.d. Web. 27 Aug 2012". Midi.org. Retrieved 2012-11-27.

108.""DLS 1 Spec". "midi.org". n.d. Web. 27 Aug 2012". Midi.org. Retrieved 2012-11-27.

109.Ashour, Gal, et al. . *usb.org* USB Implementers Forum. 1 November 1999. Web. 22 August 2012

110.Wiffen, Paul. "An Introduction To mLAN, Part 1". *Sound On Sound*. SOS Publications. Aug 2000. Print.

111.Wiffen, Paul. "An Introduction To mLAN, Part 2". *Sound On Sound*. SOS Publications. Sep 2000. Print.

112."MIDI Cables & Transports". *midi.org*. Music Manufacturers Association. n.d. Web. 27 August 2012.

113.Vail, Mark. "Voyetra 8: The original rackmount analog polysynth". *Keyboardmagazine*. Turtle Beach. Retrieved 21 May 2013.

114."CS2x Control Synthesizer Owner's Manual". Yamaha Corporation, 1998.

115.""PreSonus FIREstation". "presonus.com". n.p. n.d. Web. 18 Aug 2012". Presonus.com. Retrieved 2012-11-27.

116."rtpMIDI". *tobias-erichsen.de*. n.p. n.d. Web. 22 August 2012 Windows RTP-MIDI driver download

117.Wright, Matt. "OpenSoundControl". *cnmat.berkely.edu*. 23 June 2004. Web. 27 August 2012.

118.Fraietta, Angelo. "Open Sound Control: Constraints and Limitations". New Interfaces for Musical Expression 8th International Conference. 5–7 June 2008, Genoa, Italy. Web.

119.Kirn, Peter. "Golden Age of Wireless: Korg iOS Sync, Android + MIDI Hardware, Enter Bluetooth MIDI?". *createdigitalmusic.com*. n.p. 25 March 2011. Web.

120.""TouchOSC". "hexler.net" n.p. n.d. Web. 20 Aug 2012". Hexler.net. Retrieved 2012-11-27.

121."XBee Adapter – wireless Arduino programming". *ladyada.net*. n.p. 17 May 2011. Web. 20 Aug 2012.

122. http://www.humatic.de/htools/touchdaw/. Missing or empty |title= (help)
123. "MMA HD Protocol Announcement". *midi.org*. MIDI Manufacturers Association. n.d. Web. 22 August 2012
124. "General Meeting for MIDI developers by MMA". *pro-music-news.com*. Pro-Music-News. n.d. 22 August 2012
125. NAMM 2013 Panel discussion: Past, present and future of MIDI - FutureMusicMagazine