MIDI and MIDI files
An introduction

Author: Heiko Plate
E-mail: heikoplate@gmx.de

Translated by www.deepl.com/translate

Table of contents

1. What is this writing about?

Everyone who plays, records or edits MIDI files should know how to create sounds, i.e. tones and effects, with these files. In particular, one needs to have elementary knowledge of the internal structure.

This font is intended for beginners, i.e. no previous knowledge is required. The reader will learn, among other things, what MIDI files are, what types are available and how to create acoustic sounds with them.

2. What is a MIDI file and what happens during playback?

MIDI files are computer files that contain encrypted instructions that allow pieces of music to be played on electronic devices. MIDI files do not contain sounds like audio files: The sounds are generated elsewhere, controlled by the instructions in the MIDI file.

Basically, there are two instances of MIDI files in play: A **MIDI player** that reads MIDI files and sends the instructions contained in them, so-called **MIDI events**, at the right time to a sound generator, which is solely responsible for sound generation.

MIDI players are multimedia players running under operating systems, sequencer programs (e.g. Cubase, Sonar) and also tools for editing MIDI files.

On the PC, sound generators are realized by sound cards or by sound chips located on the motherboards. The keyboard software also contains both instances, but they are not directly visible. The MIDI file itself is decisive for the quality of the music arrangement, but the sound quality is solely determined by the sound generator used.

2.1 What are MIDI events?

Each MIDI event consists of a sequence of numbers that are understood and evaluated by the sound generator. A MIDI file now consists of a sequence of MIDI events separated by time intervals.

MIDI-Event1 - Time interval1 - MIDI-Event2 - Time interval2 - MIDI-Event3 - Time interval3...

For example, the MIDI event "151 60 51" causes the sound generator to output the note C.

The time intervals are counted in units of so-called ticks. A tick is the smallest time grid element usable by the MIDI player. The relationship between the tick time and the true time, e.g. in milliseconds, is defined in the MIDI file. The number of ticks that must be specified for a quarter note between "Note On" and "Note Off" is defined in the MIDI file. The specification is called PPQN (pulses per quarter note). When calculating the true time interval, however, the tempo still plays a role: With a PPQN of 400 and a MIDI file played back at tempo 120, a quarter note has a length of half a second, i.e. a tick in this example corresponds to eight hundredths of a second.

2.2 More about sound generators

There are essentially three different types of sound generators.

1. the so-called software synthesizers. These include the sound chip on the PC motherboard, but also special drivers that process MIDI events and generate sounds from them.
2. sound generators that are implemented on sound or audio cards.
3. the sound generators contained in the keyboards.

There is a variety of MIDI events and not every sound generator can evaluate all types. The standard sound generators available on PCs are only able to process the events agreed upon in the General MIDI (GM) specification. More about General MIDI in the following section.

The usual software synthesizers and sound cards are not very suitable for reproducing the complex XG effects and XG voices of Yamaha keyboards, even if they are so-called XG sound cards. Yamaha XG MIDI files should only be played with the devices for which they are specialized.

3. MIDI and General MIDI

MIDI is nothing else than the way MIDI events are passed on via special lines (e.g. MIDI cables). Both the sender and receiver of the data can be keyboards, MIDI sequencers and synthesizers. The stream of MIDI events can also be sent internally in a device or externally from one instance to another, e.g. from a MIDI player to a sound generator or from one keyboard to another.

If MIDI is used to define more or less a hardware and transmission, General MIDI (GM) is used to define the type of data to be transmitted, i.e. which MIDI events with which meanings there are, and how their structure is defined.

GM was established in 1991 as an extension of the then MIDI standard. GM was extended in 1999, which then led to General MIDI 2 (GM 2). GM is a manufacturer-independent agreement, and almost all sound generators are therefore able to play MIDI files that comply with the GM agreements.

In addition to GM, there are manufacturer's own standards for similar purposes, such as the GS standard from Roland or the XG standard from Yamaha. Both offer possibilities that go beyond General MIDI, but are still backwards compatible with the GM standard. So it is possible to play back pieces composed according to XG and GS with any GM sound generator with some quality restrictions.

For example, Yamaha XG knows more than 615 instead of the 128 GM instrument voices and also has wide possibilities to manipulate the sound.

4. Via MIDI channels

The term MIDI channel was defined in the MIDI agreements. The MIDI events forwarded to the tone generator can independently operate up to sixteen of these channels. This means that MIDI-capable electronic musical instruments are capable of producing several different sounds at the same time and outputting them mixed against each other; e.g. an organ sound simultaneously with a string sound.

5. MIDI file formats

Standard MIDI files (file extension .mid) consist of one or more so-called tracks. Each track independently contains a sequence of MIDI events and time intervals, which are arranged chronologically starting at time position 0 (bar 1, beat 1, tick 0).

MIDI Files (SMF for short) are available in the formats SMF 0 and SMF 1, and there was originally another format SMF 2, but it was practically never used.

Format 0 combines all MIDI channels into one track.

MIDI files of format 1 contain MIDI events distributed in several tracks. Each track can have its own name. Often a separate track is created for each MIDI channel; the so-called Meta Events and SysEx, which are not assigned to any channel, are then located in additional tracks. This strategy is not absolutely necessary, however.

SMF 0 is the standard format of Yamaha keyboards. However, the keyboards also accept SMF 1 and obsolete Yamaha-specific formats such as XF and MOD.

6. The MIDI events

There are three groups of MIDI events: **Standard MIDI events**, **Meta-Events** and **System Exclusive Events (SysEx)**. Meta-Events are not forwarded via MIDI, unlike Standard MIDI-Events and SysEx.

6.1 Standard MIDI events

Standard MIDI events only affect one specific MIDI channel at a time. For the simultaneous output of note C to three MIDI channels, a MIDI file must therefore contain three note MIDI events.

6.1.1 Note events

"Note On" and "Note Off"

These two events always appear together: After activating a note with the MIDI event "Note On", the note is output until it is terminated at an appropriate time by an associated "Note Off". Missing Note Off events cause so-called Note Hangers.

Note events contain, in addition to the note and MIDI channel, a further parameter that determines the velocity. Modern sound generators such as Yamaha keyboards respond to high velocity values by increasing the volume and changing the character of the sound.

6.1.2 Program Change

The Program Change Event is used to set an **instrument voice**, e.g. a piano or trumpet, for the selected channel. The following notes will then be played with this voice. Instead of instrumental voice, other names are also common: e.g. voice, timbre, patch, program and others; they all denote the same thing.

With Program Change up to 128 different voices can be set. According to the so-called General MIDI agreements (see section 5: MIDI and General MIDI) are also no longer possible. This deficiency has been eliminated in newer sound generators, e.g. the Yamaha keyboards, by the introduction of so-called **Voice Banks.** A voice bank is set via the two controller events 0 (MSB) and 32 (LSB). The subsequent Program Change Event switches on one of up to 128 voices of the voice bank thus defined.

6.1.3 Controller, RPN and NRPN

Controller events are elementary MIDI instructions that influence the general state of the sound generator or the character of the sound. Controller events are accepted and at least partially interpreted by almost all sound generators and sound cards. The controller events are numbered from 0 to 127, but not all numbers are occupied. RPN and NRPN can be seen as extensions of controllers, because they are realized by sequences of special controllers. Only the most important controller events are described below.

Controller 0 and Controller 32: Bank Select MSB and LSB

A voice bank is a group of a maximum of 128 instrumental voices, numbered from 0 to 127.

General MIDI knows only 128 melody voices, but some sound generators, e.g. the Yamaha keyboards, allow different voice banks to be set, from which the desired voice of the current bank can then be selected by a Program Change Event.

Controller 1: Modulation

Modulation normally creates a vibrato effect of the voice assigned to the channel.

Controller 5: Portamento Time

Portamento is an effect that causes a smooth transition from one tone to another; both tones must overlap slightly. The Portamento Time event defines the intensity of the transition. Portamento is switched on or off with the Controller 65 (Portamento).

Controller 7 Main Volume

The Controller Event Main Volume is used to adjust the volume of the voice assigned to the MIDI channel.

Controller 10: Panpot

With the Controller Event Panpot the panorama position, i.e. the stereo field of the loudspeakers, is adjusted.

Controller 11: Expression

Controller Event Expression changes the volume of a MIDI channel to a certain percentage of the volume currently set with controller 7 (Main Volume).

Controller 64: Sustain On/Off

The Sustain On/Off controller event turns the sustain effect on or off. In the On state, all notes of the MIDI channel are held, even if the notes have been terminated by Note Off.

Controller 65: Portamento

This controller event switches the portamento effect on or off. This activates or deactivates the controller event Portamento Time No. 5

Controller 66: Sostenuto

The controller event Sostenuto works similar to the Sustain effect, but the effect only affects notes that are already sounding at the time of the event; i.e. the sound generator has received Note On events and the associated Note Off events have not yet been sent. All notes that are activated in the Sostenuto On state are not held.

Controller 71: Harmonic Content

This event influences the frequency content of the sound. High parameter values produce eccentric sounds.

Controller 72: Release Time

Release Time affects the time for the sound to decay after Note Off,

Controller 73: Attack Time

Attack Time affects the length of time between Note On and reaching the maximum volume.

Controller 74: Brightness

Brightness affects the brightness (frequency bandwidth) of the sound. Higher parameter values result in a sharper sound.

Controller 75: Decay Time

This controller is used to influence the so-called decay time: When a note is played, different volume ranges are passed through. After Note On, the volume reaches its highest value after a short time (Attack Time) and then, after the "Decay Time" has elapsed, it takes on a lower value which is normally held until Note Off.

Controller 76: Vibrato Rate

Vibrato Rate affects the speed of the Vibrato effect.

Controller 77: Vibrato Depth

Vibrato Depth affects the frequency range of the Vibrato effect.

Controller 78: Vibrato Delay

Vibrato Delay affects the time at which the Vibrato effect kicks in after Note On.

Controller 91: Reverb Send Level

This event determines the strength of the reverb effect of the voice assigned to the channel.

Controller 93: Chorus Send Level

Chorus is a sound effect in which the voice is superimposed with one derived from it. The frequencies are slightly changed. This effect gives the feeling of a chorus, i.e. the interaction of several identical instruments. Chorus Send Level specifies the strength of the Chorus effect.

Controller 94: Variation Send Level

Variation can be used to create a number of different effects such as reverb, chorus, echo, distortion, etc. With Variation Send Level, the strength of the Variation effect can be set separately for each channel in the so-called "Variation Connection System" mode.

Controller 121: Reset All Controllers

With this controller event all controller functions are initialized, i.e. called with default values.

RPN and NRPN

The limited number of 128 possible controller events and their partly inaccurate parameterization was the reason for the introduction of RPN and NRPN instructions. With RPN, manufacturer-independent functions were specified, while NRPN was introduced for manufacturer-specific extensions. To activate an RPN or NRPN function, four successive controller events are required. The first two (no. 101 and 100 for RPN, 99 and 98 for NRPN) specify the function, the following two (no. 6 and 38) set the parameter value in the range 0 to 16383. I

The RPN number is calculated by the formula (128 times Ctrl101 value plus Ctrl100 value); the NRPN number correspondingly by (128 times Ctrl99 value plus Ctrl98 value). The same applies to the RPN/NRPN values: They are calculated by the formula (128 times Ctrl6 value plus Ctrl38 value). In many cases, however, Ctrl. 38 is not evaluated.

RPN events are used, for example, to adjust the sensitivity of the pitch bend wheel or to trigger tuning of instrument voices. For example, a particular NRPN event from Yamaha will change the vibrato speed; see the Keyboard Data List.

6.1.4 Aftertouch

Aftertouch is an additional key function. By pressing the key more strongly after the first keystroke, additional signals are continuously generated, which are used for effect control on modern keyboards.

Aftertouch is realized by two different event types: Channel Aftertouch and Polyphone Aftertouch. When we simply talk about "Aftertouch", we usually mean Channel Aftertouch.

With channel aftertouch, each individual signal value of any key is evaluated for the entire channel. Polyphonic aftertouch only affects the current note.

Example: The aftertouch should trigger a tremolo effect. With Channel Aftertouch, the tones of all currently held keys start to tremble as soon as you press just one single key harder. With Polyphonic Aftertouch, on the other hand, only the tones on the keys that are actually pressed harder start to tremble.

With most electronic keyboard instruments, including Yamaha keyboards, live playing with Left, Right1 etc. is exclusively Channel Aftertouch. In MIDI files, however, Yamaha keyboards also evaluate polyphonic aftertouch events.

6.1.5 Pitch Wheel

The pitch bend wheel on the keyboard is used to change the pitch of the notes played up or down. In MIDI there is a special event that changes the pitch of individual notes.

6.2 System Exclusive Events (SysEx)

SysEx are events that address special functions of the sound generators that are not defined in the standard MIDI agreements. SysEx are transmitted like the standard MIDI events via MIDI coupling. Since they have a fixed structure, all tone generators recognize these events, but will only evaluate them if they are capable of doing so.

A distinction is made between manufacturer-independent and manufacturer-specific SysEx.

The manufacturer-independent SysEx includes, for example, the initialization of GM sound generators.

Manufacturer-dependent SysEx contain a manufacturer code. This allows sound generators to quickly decide whether they can edit the SysEx or have to overlook it.

In this paper we will not go further into the functions that can be controlled with SysEx. Especially under Yamaha XG, there are a large number of SysEx that can be used to adjust effects of instrument voices. Here we have to refer to the so-called Data Lists.

6.3 Meta Events

Meta-events can only be found in MIDI files, because unlike standard MIDI events and SysEx, they do not affect the sound generator but are only processed by the MIDI player. They are therefore not forwarded via MIDI. Meta-events do not influence the sound character but only the behaviour and appearance of the MIDI player's displays.

As with SysEx, a distinction is made between manufacturer-independent and manufacturer-specific meta-events.

6.3.1 Vendor independent meta events

The structure and effects of these events were defined in the 1988 Standard MIDI File Specification. Only the most important ones are described below.

Tempo

This event determines the playback speed. Each tempo change requires a separate tempo event.

time signature

The Time Signature Event defines the time signature. The time signature indicates how time is divided into rhythmic units, such as 3/4 , 4/4 or 6/8.

Key Signature

The Key Signature event defines the key signature. A key signature is specified by a group of accidentals (# or b), which are located directly to the right of the clef.

Sequence/Track

In midifiles of format 0 this event defines the name of the so-called song name (e.g. "In the Mood"). In format 1 files each track can be given its own name (e.g. "Piano", "Strings" etc.)

Lyric

Lyric is the English word for lyrics. A MIDI file, which displays the lyrics during playback, contains a separate lyric event for each lyric element (syllable or line). Line changes are caused by special specifications in the lyric event.

Copyright

With the copyright event the manufacturer of MIDI files can insert a copyright information and thus document his rights.

Marker and Cue Point

Markers and cue points are used to insert named position information into the MIDI file. Example: "Intro", "Refrain". Keyboards normally do not show the markers during playback. Yamaha keyboards use these events for special tasks.

Text

Text events can be used to insert texts, but these are also not displayed during playback. This allows the manufacturer of midifiles to insert internal information.

6.3.2 Yamaha XF Meta-Events

According to the standard MIDI file specification it is allowed that manufacturers can define their own meta-events. Like the manufacturer-specific SysEx, these events contain an identifier with which the manufacturer can be identified. Non-manufacturer MIDI players can thus read over these events.

In 1999, Yamaha introduced a series of its own meta-events under the name XF, which compatibly extend the SMF0 form defined in the standard.

In the following only the most important XF meta-event chord is described. There are a number of other XF Meta-Events. In this font, reference must be made to the keyboard manuals.

Chord

With these events chord names (e.g. F7, Cmin6, G#dim) can be inserted at any time position. During playback the chords are shown on keyboard displays. In addition, they also affect functions of the voice and vocal harmony.

7. In short: What are MIDI sequencers?

The characteristics of sequencers are manifold. In this section, we will restrict ourselves to MIDI sequencers that are implemented as PC software and, as manufacturer-independent programs, specialize exclusively in General MIDI. Yamaha distributed an XG sequencer under the name XGworks years ago, but it is no longer available today.

With these programs, MIDI files can be completely created and edited (edited). Via MIDI coupling, the stream of MIDI events can be recorded and stored as MIDI file. SMF-MIDI files can be loaded for further editing. With the exception of manufacturer-specific meta-events, all MIDI events can be edited. This includes deleting, changing and inserting additional events. It is also possible to create MIDI files by manually entering notes.

MIDI sequencers can be used to play back MIDI files, so they contain their own MIDI player. Normally, the stream of MIDI events is forwarded to an external sound generator (sound card, keyboard, etc.). However, there are also MIDI sequencers that generate the sound internally via so-called VST plug-ins.

End