

Electronic musical instruments

MIDI

in electronic
musical instruments

MIDI standard

MIDI – *Musical Instruments Digital Interface*

- A standard of communication (data exchange) between digital musical instruments.
- Developed by Roland, Sequential Circuits, Oberheim. First version in 1983.
- **Only control messages are transmitted, never the sound.**
- Allows for control and synchronization of digital devices.
- Ensures compatibility between devices from different vendors.

MIDI controllers and sound modules

MIDI allows for separation instrument functions:

- **sound module**: synthesizer, sampler,
- **controller** – e.g. keyboard, computer software,
- it allows for flexible connections between various controllers and sound modules.



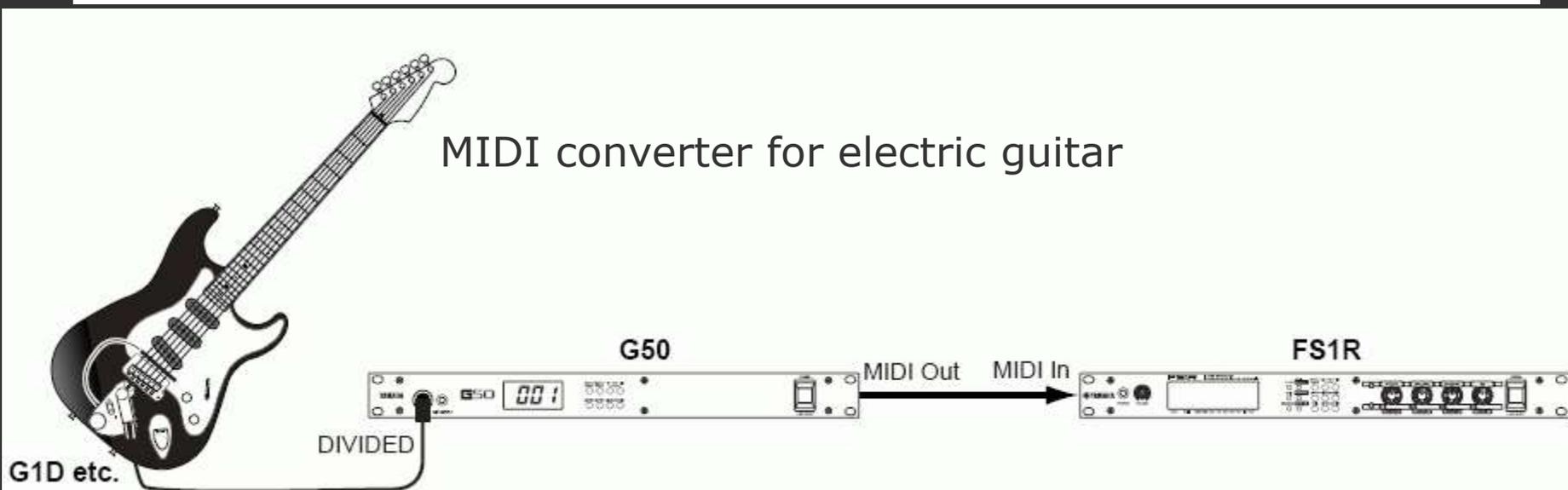
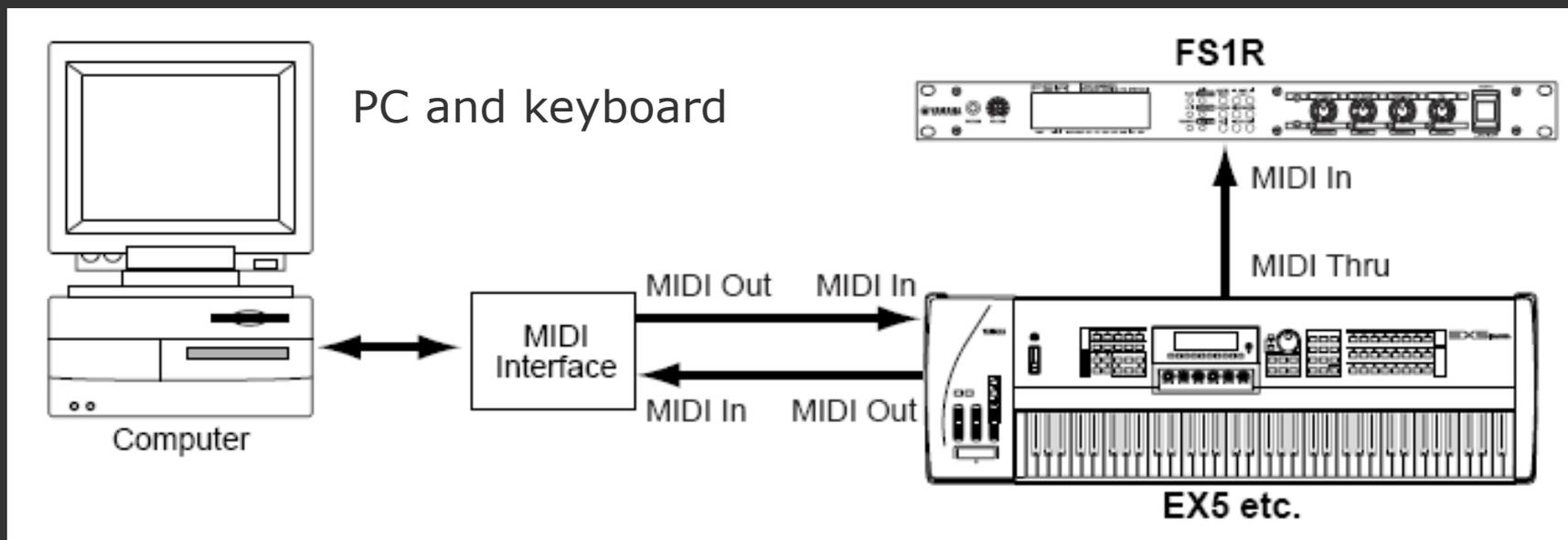
MIDI - a practical example

A rack of sound modules
(synthesizers, samplers,
sound effects),
controlled with a single
keyboard, using MIDI.

(source: Wikipedia)



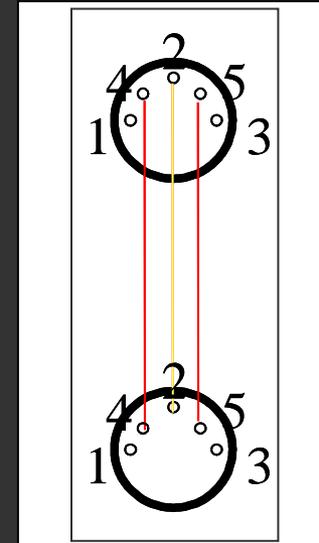
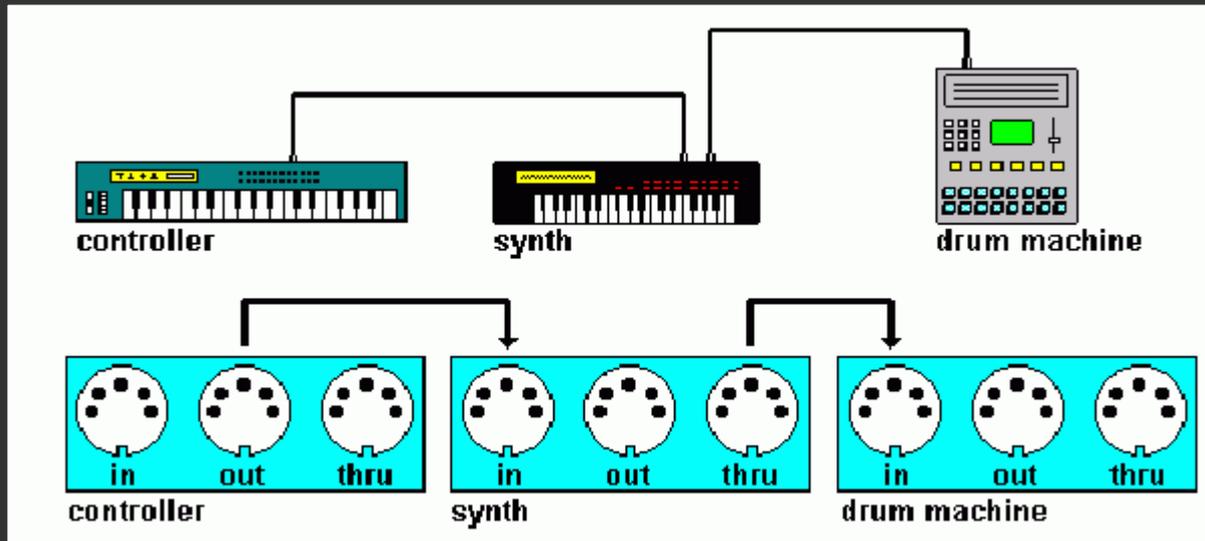
MIDI - practical examples



MIDI connections

MIDI connectors (DIN format):

- **IN** – input
- **OUT** – output
- **THRU** – passthrough output, a copy of IN



MIDI ⇔ PC connection

- Currently, most MIDI controllers use an USB-MIDI interface that can be plugged into an USB port in a computer, no drivers needed (plug & play).
- If a MIDI device has only DIN ports, a MIDI converter is needed:
 - MPU-401 to a parallel port (obsolete),
 - external USB-MIDI converter (DIN to USB).

MIDI latency

- MIDI interface is serial.
- Only one message transmitted at a time.
- If we press e.g. five keys at the same time, five messages are transmitted one after another; time difference between the first and the last one is ca. 4.8 ms.
- Delays that occur during MIDI communication are called **MIDI code latency**.
- There may be a noticeable delay between a key press and the resulting sound creation.
- MIDI messages are prioritized in order to reduce the latency.

MIDI latency

MIDI latency when a PC is used:

- system sound drivers (MME/WDM) have high latency,
- **ASIO** (*Audio Stream Input/Output*)
 - optimized drivers that don't pass data through the OS, they reduce latency (e.g. free ASIO4ALL),
- **DirectSound** (*DirectX*) and **WASAPi** (Windows only)
 - less effective than ASIO, but much better than system drivers, sufficient for not professional use.

A driver type is usually selectable in the software options.
ASIO, WASAPI or DirectX are recommended.

MIDI channels

- MIDI devices can use **16 channels** to communicate.
- If the transmitter and the receiver use the same channel, they can exchange messages.
- Channel number is stored in the message status word.
- A transmitter may send messages to multiple receivers, but each receiver has to use different channel.
- A receiver ignores messages sent to different channels than the one it is configured to.

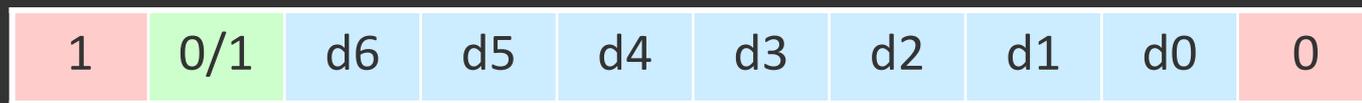
MIDI messages

A general structure of a MIDI message:

- status byte – a type of message,
- data bytes (one or two) – a message content.

Words are 10-bit:

- a start bit (1),
- a word type: 1 - status, 0 – data,
- 7 data bits (0 – 127),
- a stop bit (0).



MIDI message example

An example of a voice message:

note on – a command to start sound generation:

- 1001cccc – state bit (1), *note on* identifier (001), channel number (cccc, 0 to 15)
- 0nnnnnnnn – data bit (0), sound pitch code (nnnnnnnn), e.g. A4 = 69
- 0vvvvvvv – data bit (0), velocity value (vvvvvvv)

For example, *Note On*, pitch A4, channel 2, max. velocity:

10010001 01000101 01111111 = 145, 69, 127

MIDI message types

Channel messages – for a specified channel only:

- voice messages (sound generation),
- mode messages (configuration).

System messages – for all channels:

- common messages (for all devices),
- system exclusive messages (custom messages),
- real time messages (synchronization).

Channel voice messages

Channel voice messages control sounds that the instrument produces.

- *Note On* – start generating a sound (e.g. key pressed).
- *Note Off* – stop generating a sound (e.g. key released).
- *Control Change* – change of a parameter that affects how a sound is created (e.g. a filter cut-off frequency).
- *Pitch Bend* – a smooth pitch variation.
- *Aftertouch* – changing the pressure on a key,
- *Program Change* – change of a program, i.e. the active instrument.

Control Change (CC) messages

- A controller sends a message: CC code, command, parameter.
- CC messages may be generated by knobs, sliders, a modulation wheel, etc., and also by a sequencer.
- CC commands have a type “1011xxxx”:
 - some are defined in the standard (e.g. 01 = *Modulation Wheel*)
 - others may be assigned to a controller (e.g. a code 69 assigned to a knob by the user)
- Non-standard CC codes are device dependent.

Control Change (CC) messages

CC messages may be used to control the instrument parameters. For example:

- a subtractive synthesizer has a code CC 61 assigned to the VCF cut-off frequency,
- we configure the controller so that the selected knob sends CC 61,
- now, we can turn the knob, which sends CC 61 to the synthesizer with the current knob position,
- the synthesizer reads the CC parameter and uses it to set the cut-off frequency in the VCF.

Channel mode messages

Channel mode messages control the instrument:

- *all sound off* – turn off generation of all sounds,
- *all notes off* – send note off to all voices in the channel,
- *reset all controllers* – reset all control parameters to default values,
- *local control on/off* – turn on / off a controller,
- *omni, poly, mono* – select voice mode.

System Exclusive messages

- The MIDI standard cannot describe all needs of instrument vendors.
- System Exclusive messages (SysEx) are non-standard MIDI messages, defined by vendors.
- They can be used to send control messages to a specific device.
- Obviously, they are not portable, different devices may interpret them differently.
- A practical example: loading presets into an instrument.

MIDI instruments

- EMI has a set of **instruments**, grouped in **banks**.
- Usually, only one bank may be used at a time.
- A MIDI bank has at most **128 instruments**.
- We can use any instrument from the loaded bank.
- MIDI messages:
 - *Control Change 0 and 32* – bank change,
 - *Program Change* – instrument change.

MIDI standards

MIDI standards describe instruments (patches) that are assigned to numbers within the main bank.

The most important MIDI standards:

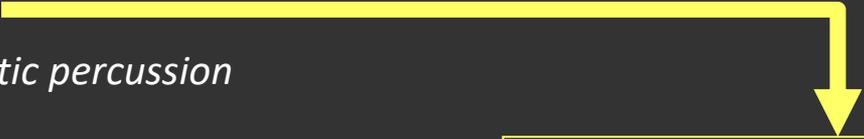
- **GS** – *General Standard* – the initial standard,
- **GM** – *General MIDI* – the most commonly used,
- **MT32** (Roland),
- **XG** (Yamaha; GM extension).

A standard ensures that selecting a given program number results in sounds of a defined instrument.

General MIDI Level 1

128 instruments are divided into 8 groups:

- 0 – 7 *Piano*
- 8 – 15 *Chromatic percussion*
- 16 – 23 *Organ*
- 24 – 31 *Guitar*
- 32 – 39 *Bass*
- 40 – 47 *Strings*
- 48 – 55 *Ensemble*
- 56 – 63 *Brass*
- 64 – 71 *Reed*
- 72 – 79 *Pipe*
- 80 – 87 *Synth Lead*
- 88 – 95 *Synth Pad*
- 96 – 103 *Synth Effects*
- 104 – 111 *Ethnic*
- 112 – 119 *Pecussive*
- 120 – 127 *Sound Effects*

- 
- 0. *Acoustic Grand Piano*
 - 1. *Bright Acoustic Piano*
 - 2. *Electric Grand Piano*
 - 3. *Honky-tonk Piano*
 - 4. *Rhodes Piano*
 - 5. *Chorused Piano*
 - 6. *Harpischord (klawesyn)*
 - 7. *Clavinet (klawikord)*

Percussive sounds

- In GM standard, messages related to generation of unpitched percussive sounds always use **channel 10**.
- Each pitch code corresponds to a different percussive instrument. At most 128 sounds are defined in the bank.
- This convention is used for sounds with undefined pitch.
- The main bank has 47 percussive sounds (codes 35 – 81).
- Different banks may have different sets of percussive sounds.

MIDI Implementation Chart

MIDI Implementation Chart is a specification of messages that a given device can transmit and receive, and interpretation of specific messages by the device.

The chart is typically a table with columns:

- function (message type)
- transmitted – whether a message can be sent,
- recognized – whether a message can be interpreted,
- remarks (e.g. meaning of a CC code).

Symbols: “O” = yes, “X” = no. Optionally, a description may be added.

An example of a MID chart for a synthesizer

	Function	Transmitted	Recognized	Remarks
Basic Channel	Default Changed	1 - 7, 10 X	1 - 16 X	
Mode	Default Messages Altered	Mode 3 *****	Mode 1 Mode 1,2 (M=1)	*1 *2
Note Number	True Voice	0 - 127 *****	0 - 127 0 - 127	
Velocity	Note ON Note OFF	O *3 X	O X	
After Touch	Key's Ch's	X X	O *3 O *3	
Pitch Bend		X	O *3	
Control Change	0,32	X	O *3	Bank select
	1	X	O *3	Modulation
	5	X	O *3	Portamento time
	6,38	X	O *3	Data entry
	7	X	O *3	Volume
	10	X	O *3	Panpot
	11	X	O *3	Expression
	64	X	O *3	Hold 1
	65	X	O *3	Portamento
	66	X	O *3	Sostenuto
	67	X	O *3	Soft
	84	X	O *3	Portamento ctrl
	91	X	O (Reverb) *3	Effect 1 depth
93	X	O (Chorus) *3	Effect 3 depth	
98,99	X	O *3	NRPN lsb,msb	
100,101	X	O *3	RPN lsb,msb	
Program Change	True #	X *****	O *3 0 - 127	Program # 1-128
System Exclusive		O	O	

MIDI Implementation Chart

- In order to check whether a message can be transmitted between devices, we compare charts from both devices.
- We need to find a specific function in both charts and compare the content:
 - *transmitted* column for the sender,
 - *recognised* column for the receiver.
- If either device has **X**, a message cannot be transmitted.
- If **O** is in both charts, a message may be sent and received.
- We should also read all additional notes for this entry.

MIDI controllers

MIDI keyboard:

- piano-type keys, dynamic (with velocity sensors),
- *pitch bend* and *modulation* wheels,
- knobs and sliders (sending CC codes),
- a display,
- foot controller input (*sustain*),
- DIN and USB outputs.



MIDI drivers

Examples of MIDI controllers other than the keyboard



MIDI drum pad



virtual gramophone (for DJs)

MIDI drivers



MIDI drum kit



MIDI guitar

MIDI sequencer

MIDI sequencer is a hardware or (usually) a software that can:

- record,
- edit (modify, process),
- play back

sequences of MIDI codes.

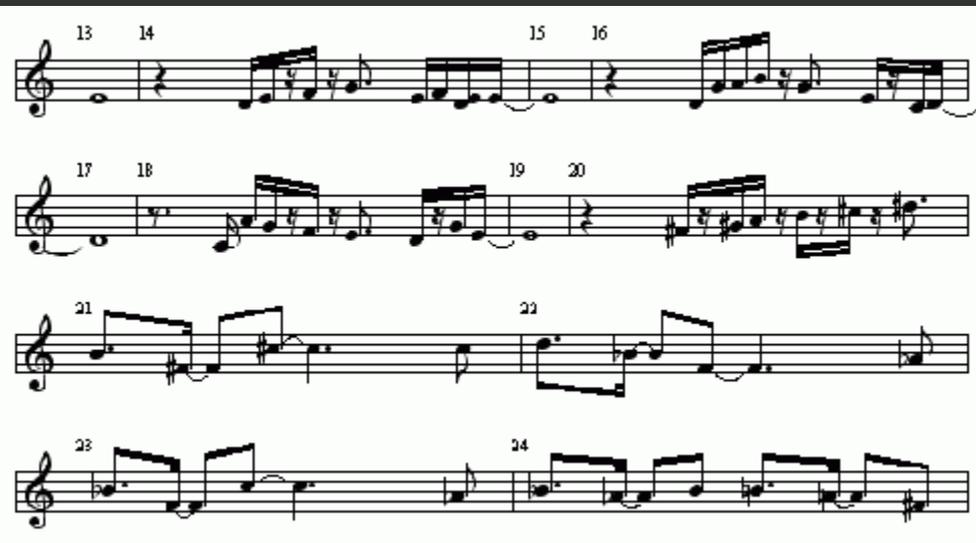
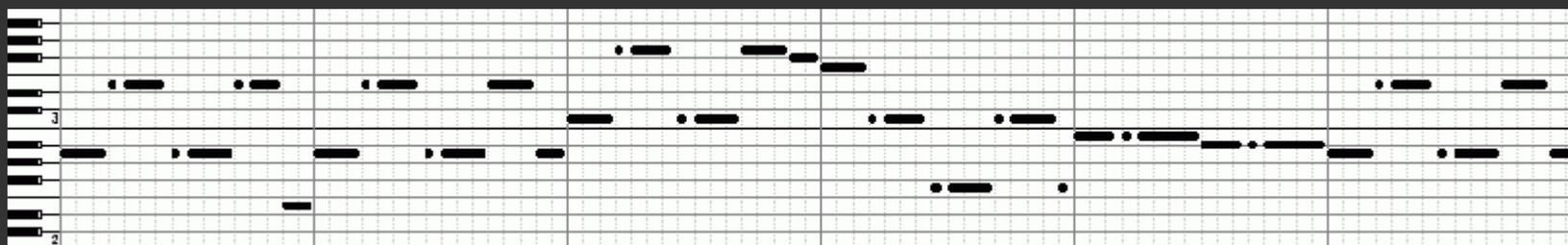
A sequencer controls the instrument, sending MIDI codes to the instrument and playing on it instead of a live musician.

Typical functions of a MIDI sequencer

- Recording MIDI codes, in real time and in step mode.
- Playback of MIDI codes through specified channels.
- MIDI codes presentation:
 - tracks,
 - piano roll,
 - score (sheet),
 - list of MIDI messages.
- Editing functions, e.g. quantization.
- Graphical mixer, metronome, etc.

MIDI codes in a sequencer

R	Type	Name	Pch	Port	Chn	Vol	Trans	
1		Sample	-	1	--		0.0	1
2		Copyrig	-	1	--		0.0	2
3		Voyetra	-	1	--		0.0	3
4		All Righ	-	1	--		0.0	4
5	🥁	Drums	0	1	10		0.0	5
6	🎸	Bass	39	1	4		0.0	6
7	🎹	Melody	12	1	2		0.0	7
8	🎹	Keyboz	5	1	3		0.0	8



Type	Position	Channel	Parameters
Note	003:01:000 00:04:06:65	04	A2 107 064 00:00:000
Note	003:01:360 00:04:16:49	04	E3 107 064 00:00:080
Note	003:02:000 00:04:19:71	04	E3 107 064 00:00:330
Note	003:02:360 00:05:05:36	04	A2 107 064 00:01:000
Note	003:03:360 00:05:18:21	04	E3 107 064 00:00:090
Note	003:04:000 00:05:21:42	04	E3 107 064 00:00:360
Note	003:04:360 00:06:07:07	04	A2 107 064 00:00:095
Note	004:01:000 00:06:10:28	04	A2 107 064 00:00:360
Note	004:01:360 00:06:19:92	04	E3 107 064 00:00:090
Note	004:02:000 00:06:23:14	04	E3 107 064 00:00:330
Note	004:02:360 00:07:08:79	04	A2 107 064 00:00:095
Note	004:03:000 00:07:11:99	04	A2 107 064 00:00:360
Note	004:03:360 00:07:21:64	04	E3 107 064 00:00:080
Note	004:04:000 00:08:00:85	04	E3 107 064 00:00:240
Note	004:04:240 00:08:07:29	04	E2 107 064 00:00:240
Note	005:01:000 00:08:13:71	04	A2 107 064 00:00:360

Editing in a sequencer

MIDI codes editing:

- adding, removing, modification of notes,
- altering the tempo,
- transposition (pitch change),
- quantization – aligning note onsets and duration,
- “humanization” – adding random imperfections.

MIDI sequencers and DAW

Practically all modern MIDI sequencers are software. Many of them combine MIDI sequencer functions with audio editing, into Digital Audio Workstations (DAW).

Examples of DAWs with MIDI sequencing:

- Cubase (Steinberg),
- Cakewalk,
- Logic Audio
- Ableton Live
- Pro Tools

Bibliography

- *MIDI Tech Specs & Info*:
<https://www.midi.org/specifications>
- Wikipedia: <https://en.wikipedia.org/wiki/MIDI>
- David Miles Huber: *The MIDI manual: a practical guide to MIDI in the project studio*. Focal Press 2007.
- *Music Studio Producer*: <http://frieve.com/english/musicstd/musicstd.html>