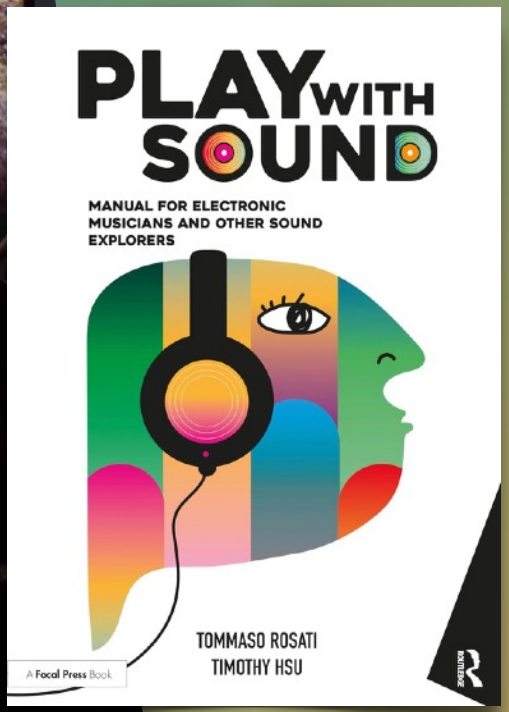
COMMUNICATION PROTOCOLS

MIDI
OSC

TOMMASO ROSATI
SOUND ART

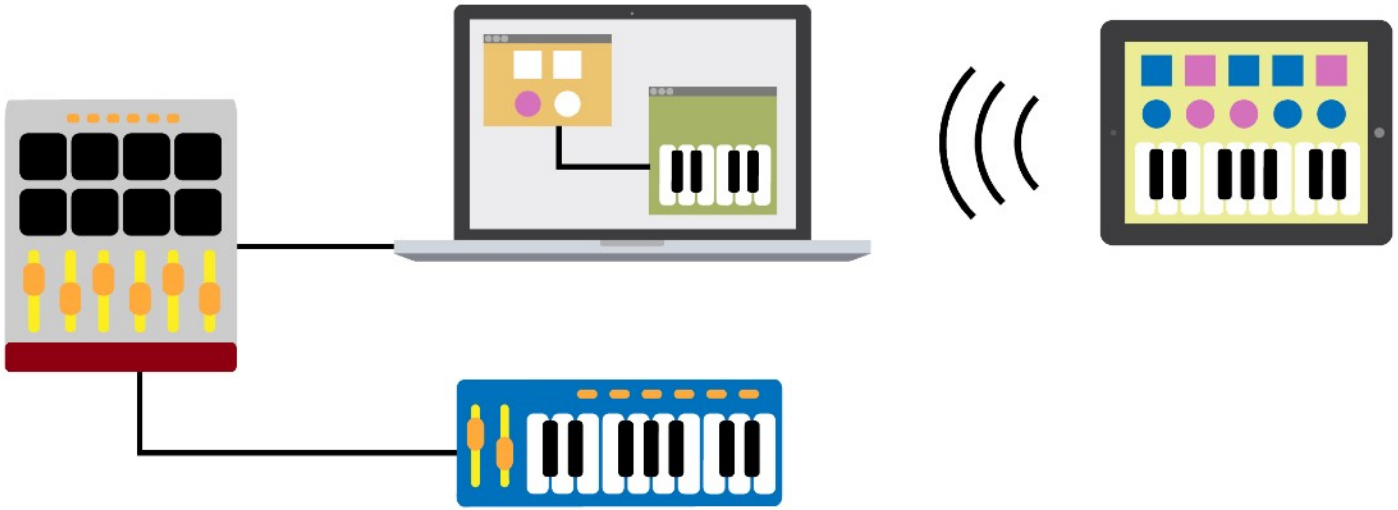
THE BOOK IS NOW AVAILABLE!



MIDI

Musical Instrument Digital Interface

MIDI, short for Musical Instrument Digital Interface, is a standardized protocol consisting of messages and rules for communication between musical instruments and other digital interfaces. It is important to note that MIDI is a series of messages that specify musical events, but it is not audio



MIDI

Musical Instrument Digital Interface

The first instrument to use the MIDI protocol was the **Sequential Circuits Prophet 600** from 1983.

While the MIDI 2.0 update in 2023 was substantial, MIDI's structure remains conceptually the same, proving its value as an easy-to-program language that does not require complex hard-to-find technology.

MIDI

Musical Instrument Digital Interface

Cables

- standard 5-pin MIDI conn

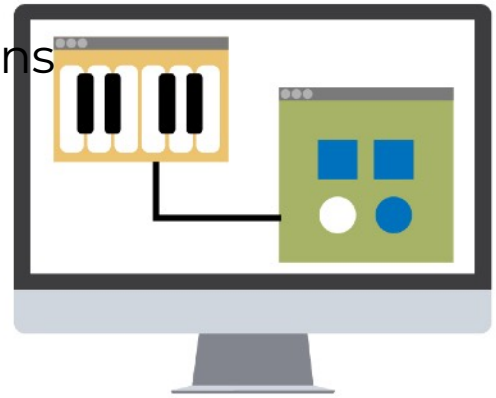
- USB cable

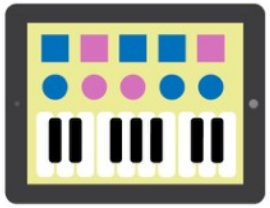
MIDI
Musical Instrument Digital Interface

Can be also transmitted

Virtual software connections

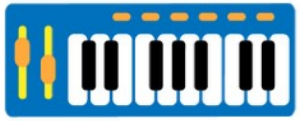
Bridge software like
MIDI Yoke (PC)
or using Driver IAC (Mac)





WIFI or Bluetooth LE

on PC, Mac, Linux, iOS, Android...



Connections

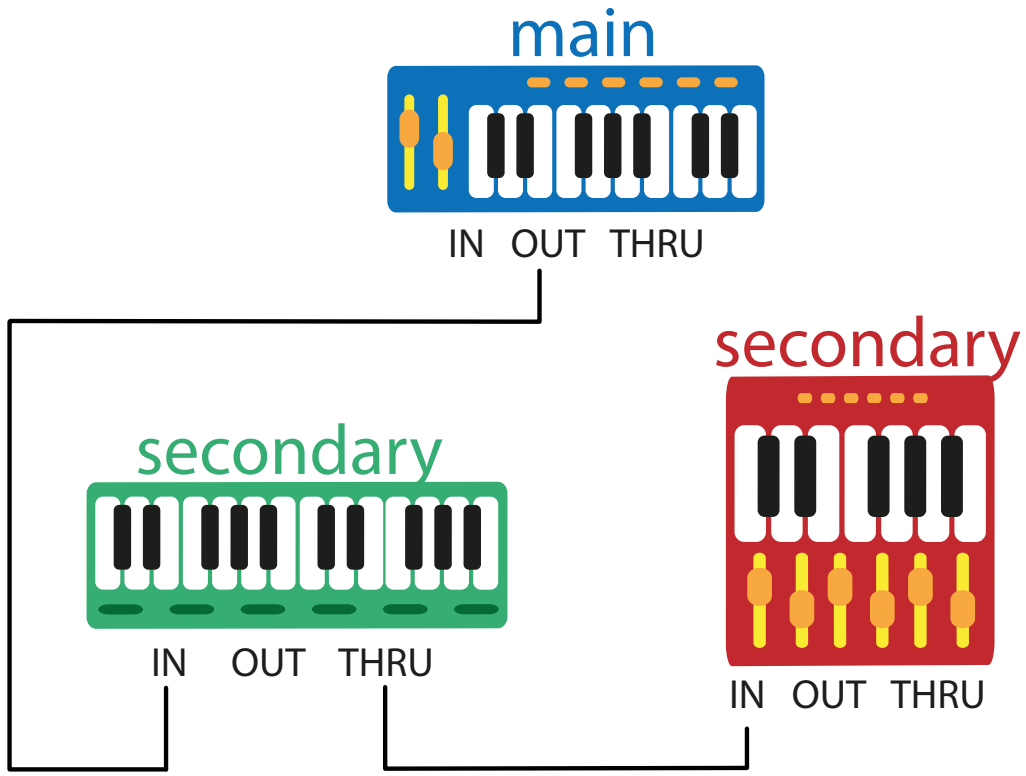
MIDI
Musical Instrument Digital Interface

In MIDI connections with standard 5-pin MIDI cables we can have three type of ports:

MIDI IN, receives MIDI messages from other external devices

MIDI OUT, sends MIDI messages out of the main device

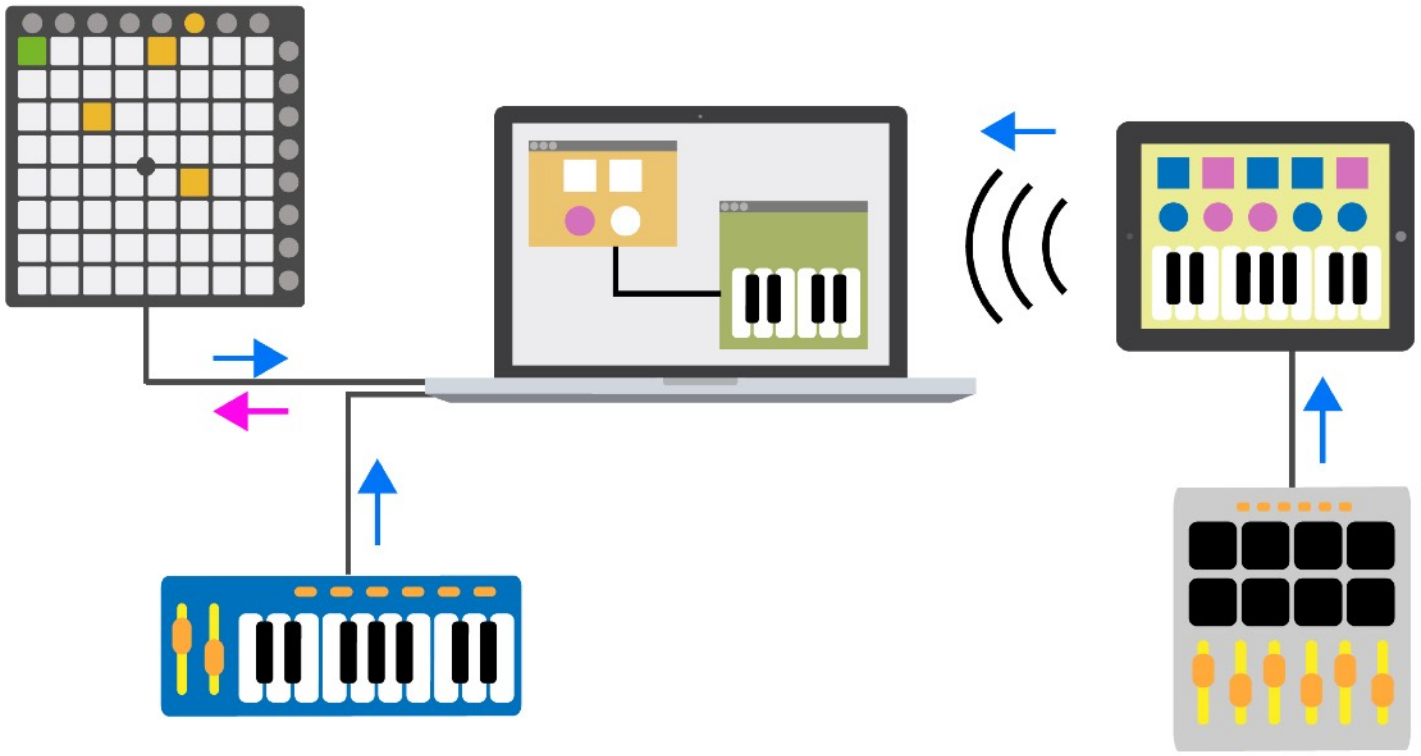
MIDI THRU, passes the MIDI signal from the IN port to the OUT port without applying variations.



Connections

MIDI
Musical Instrument Digital Interface

In systems that are comprised of entirely digital devices, the concept of **main** and **secondary** devices loses some relevance due to the flexibility offered by digital systems.



MIDI
Musical Instrument Digital Interface

Types of messages

Channel Voice Message
control the parameters that are needed to convert musical events into MIDI and can be routed to one of the **16 channels** available in MIDI

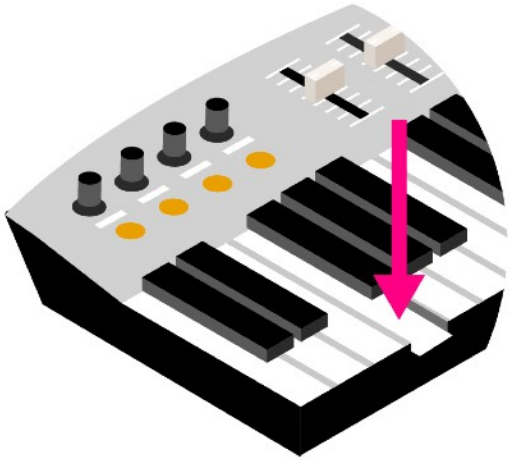
System Message
apply globally within the system and not to a specific device

Types of messages
Channel Voice Message

MIDI
Musical Instrument Digital Interface

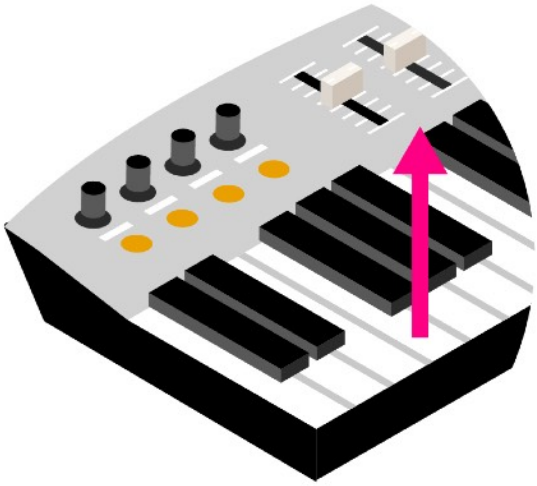
status: NOTE ON
1st data byte: NOTE
2nd data byte: VELOCITY

NOTE ON – When a key is pressed on the keyboard, a Note ON (note pressed) message is sent, consisting of two numbers: **pitch (0-127)** and **velocity (0-127)**.



status: NOTE OFF
1st data byte: NOTE
2nd data byte: VELOCITY

NOTE OFF – When the key that was previously pressed is now released, a Note OFF message is generated that also consists of two numbers: **pitch (0-127)** and **velocity (0-127)** the key is released.

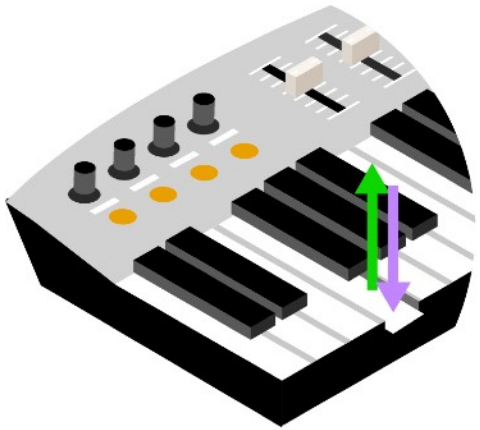


Types of messages
Channel Voice Message

MIDI
Musical Instrument Digital Interface

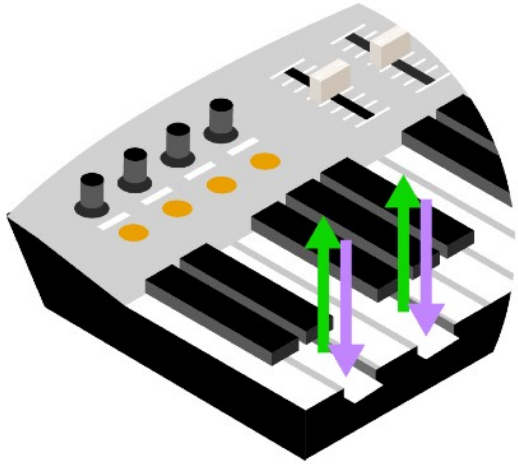
status: AFTERTOUCH
1st data byte: AFTERTOUCH VALUE

AFTERTOUCH or CHANNEL PRESSURE – This parameter takes the value of a sensor and transmits any changes in the pressure exerted on a previously pressed note in real time. This pressure, applied to the entire keyboard.



status: AFTERTOUCH
1st data byte: NOTE
2nd data byte: AFTERTOUCH VALUE

POLYPHONIC AFTERTOUCH or POLYPHONIC KEY PRESSURE – It is similar to the Aftertouch but refers to independent pressure sensors for each note.

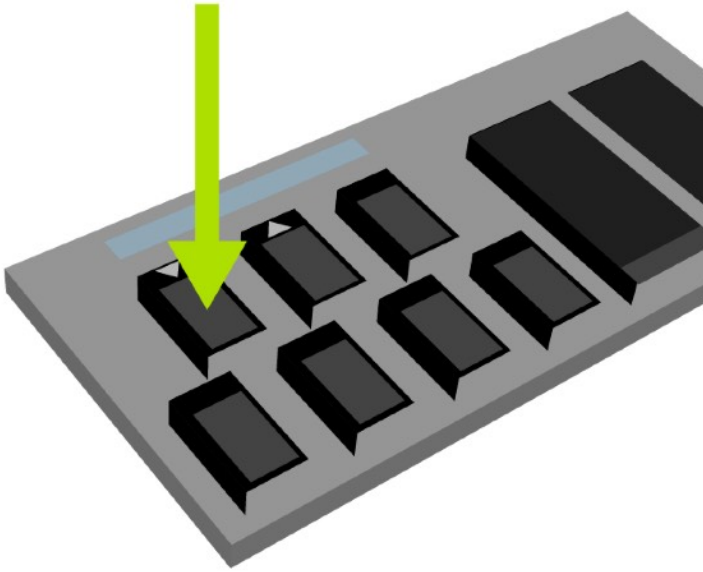


Types of messages
Channel Voice Message

MIDI
Musical Instrument Digital Interface

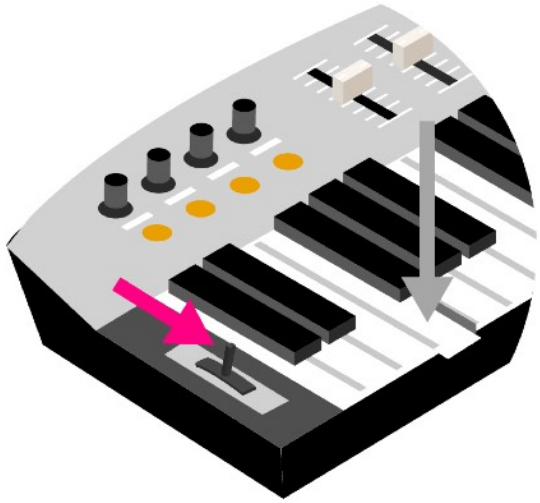
status: PROGRAM CHANGE
1st data byte: PRESET NUMBER

PROGRAM CHANGE - We use it to change a preset, or program.



status: PITCH BENDER
1st data byte: VALUE A (MSB)
2nd data byte: VALUE B (LSB)

PITCH BEND – MIDI keyboards have sliders, wheels, or joysticks which can be moved to vary the pitch during the sustain of a note.

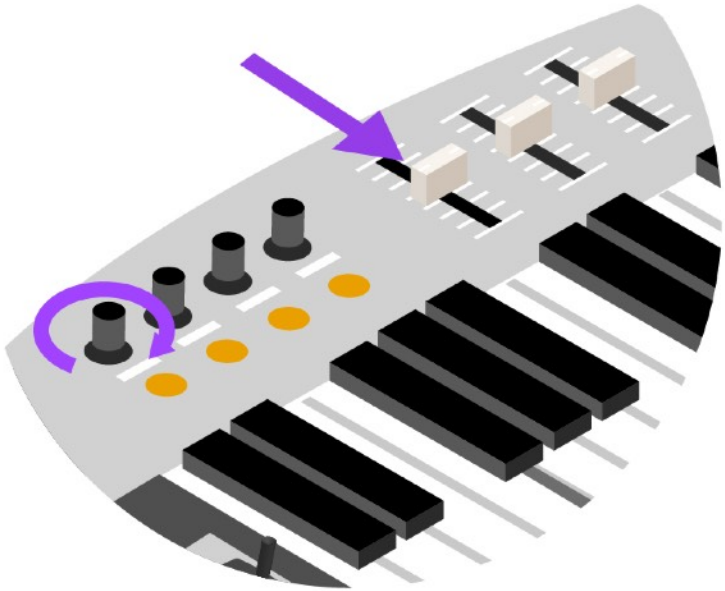


Types of messages
Channel Voice Message

MIDI
Musical Instrument Digital Interface

status: CONTROL CHANGE
1st data byte: CC NUMBER
2nd data byte: VALUE

CONTROL CHANGE – This message, consisting of two numbers (CC 0-127 and VALUE 0-127), allows control values (up to 128 controllers for each channel) to be transmitted.

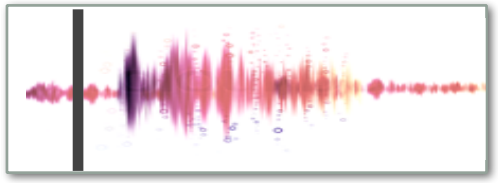


Types of messages
System Message

MIDI
Musical Instrument Digital Interface

COMMON MESSAGE

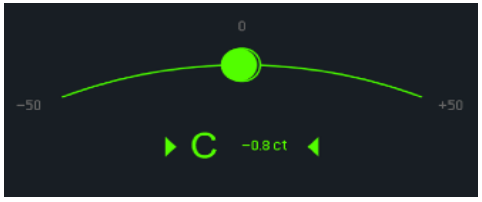
MIDI Time Code Quarter Frame **synchronizes** multiple streams of audio and video in the format hours:minutes:seconds:frames.



Song Position Pointer indicates the **playback** or cursor's position when playing a MIDI file.

1223050	Grace Flows Down	David E. Bell,Louie Giglio,Rod Padgett
3111376	Amazing Grace	John Newton,Nathan Pellingham
4737522	Amazing	John Newton,John P. Rees,Mark Roach
64533	All Because Of God's Amazing Grace	Stephen R. Adams
3270152	Amazing Grace	John Newton,John P. Rees,Ken Barker,Word Music G...
4985666	Amazing Grace	John Newton,John P. Rees,Shannon Anderson
4639462	Amazing Grace	John Newton,Jon Bauer
666072	Amazing Grace	Edwin Othello Excell,John Newton,John P. Rees,O. D.,...

Song Select allows you to **select a song** within a sequencer.



Tune Request is used to send the **tuning value** of an instrument.

Types of messages
System Message

MIDI
Musical Instrument Digital Interface

REAL TIME

MIDI Clock is used to **synchronize** the BPM (beats per minute) of multiple connected MIDI instruments.

Start is used to bring all instruments to the same **starting position** of the song

Stop **stops** the recording or playback of instruments connected to the sequencer.

Active Sensing is sent every 300 milliseconds to **keep the connection** between main and secondary devices active.

System Reset **resets** the secondary devices to the default conditions.

Types of messages
System Message

MIDI
Musical Instrument Digital Interface

EXCLUSIVE MESSAGE

System Exclusive (SysEx) messages control each instrument's global functionality and are used according to the manufacturer's requirements. Since they are specific to a particular device, they can only be interpreted by compatible devices.

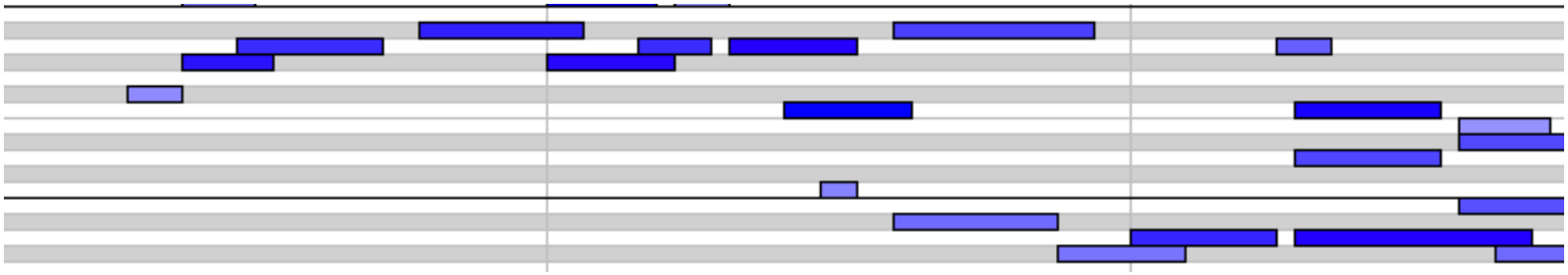
MIDI file

MIDI
Musical Instrument Digital Interface

nomefile.mid

A **MIDI file**, **.mid**, is a file format that contains a sequence of MIDI data, consisting of several messages distributed in various channels, capable of being interpreted by any MIDI compatible hardware device or software instrument.

It's NOT an audio file!



Standard MIDI
General MIDI

MIDI
Musical Instrument Digital Interface

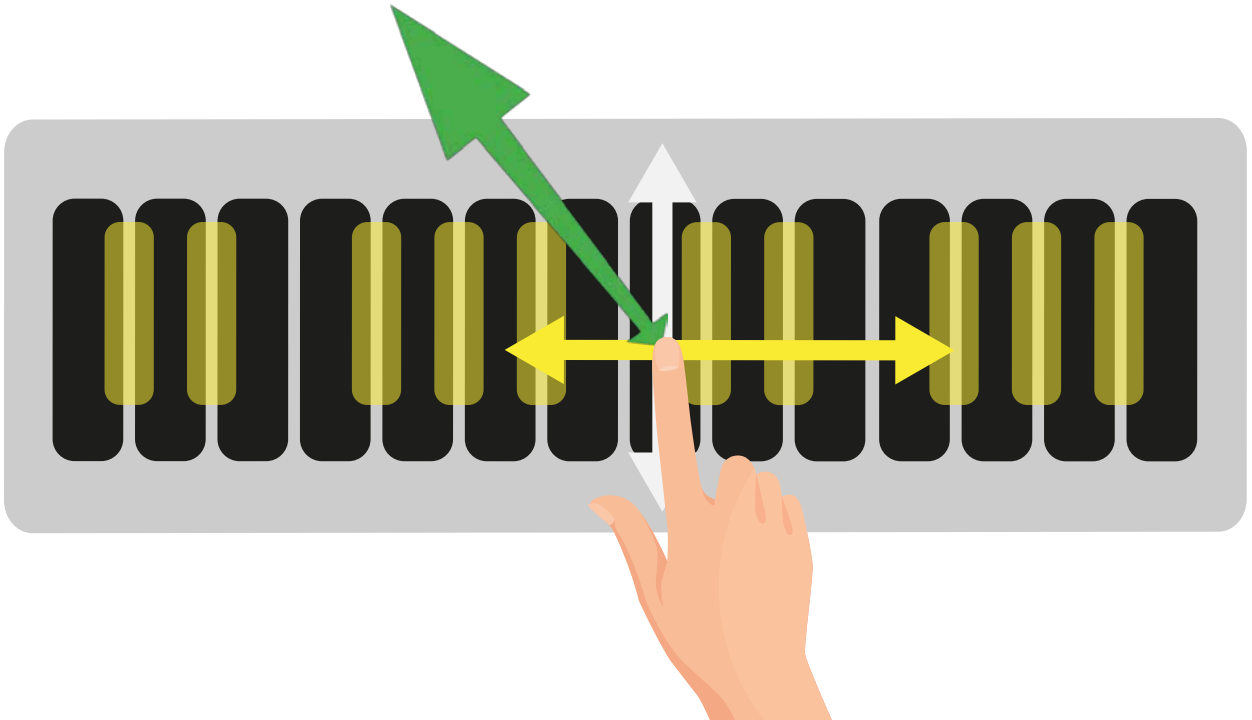
Some standards, like **General MIDI**, handle the way instruments are identified, allowing us to quickly assign tracks to various instruments of our choice.

MPE

MIDI
Musical Instrument Digital Interface

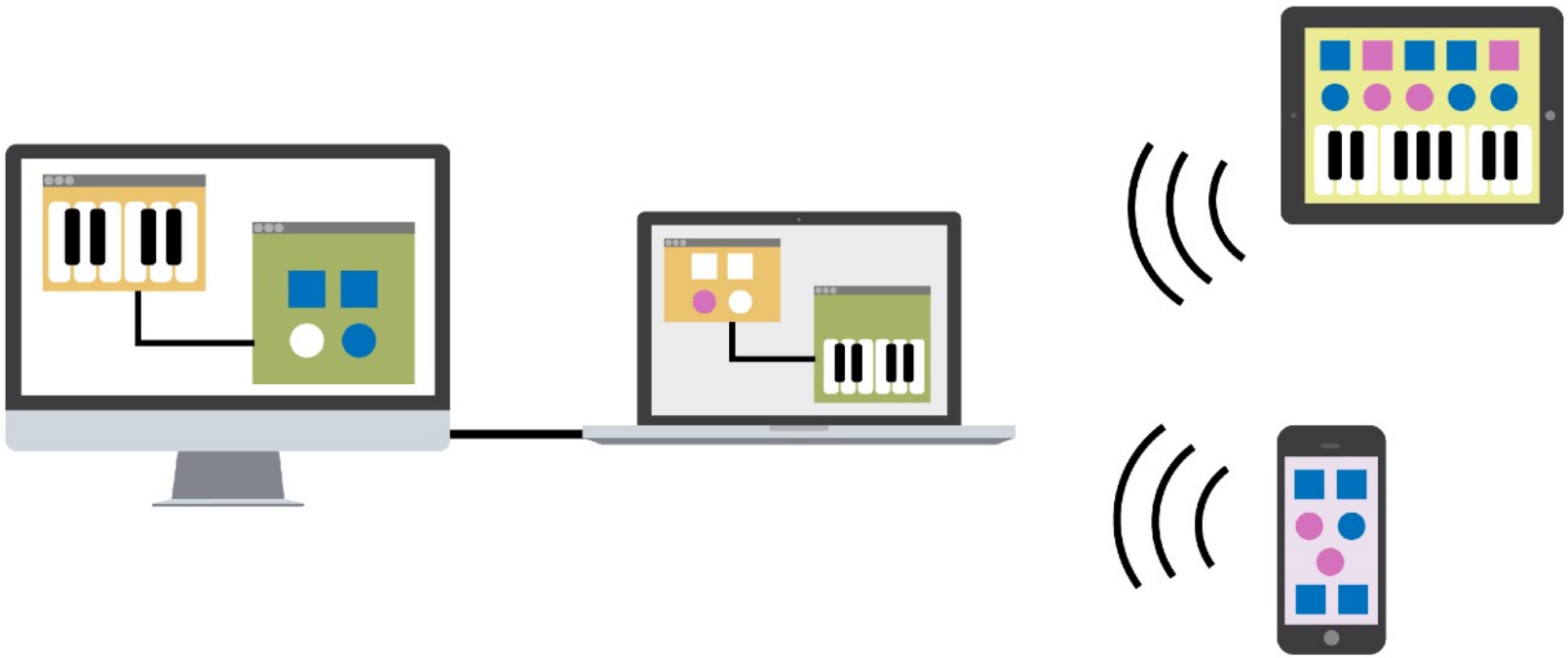
MPE merges multiple MIDI channels to give each note **more gestural possibilities**, like controlling horizontal and vertical finger position and pressure.

So, with one finger, I can potentially control the note, its intensity, variations of its timbre, and even the amount of reverb I am applying to it.



OSC
Open Sound Control

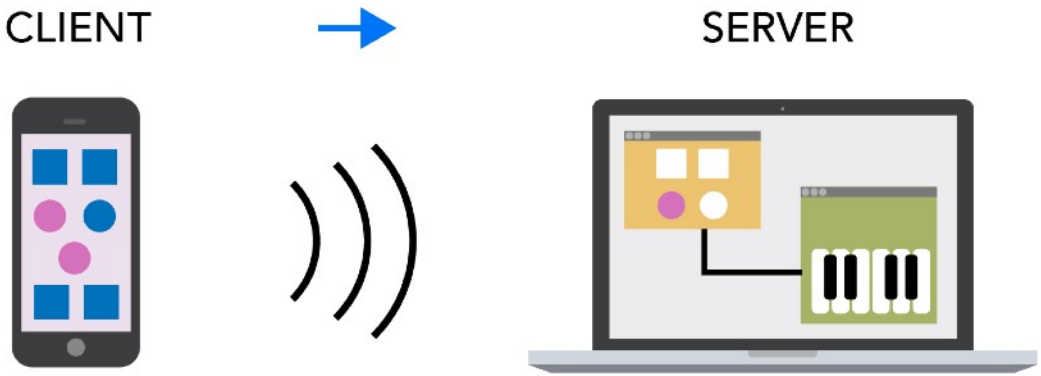
OpenSound Control (OSC), created by CNMAT in 1997, is an open-source protocol for communication between computers, audio synthesizers, and other multimedia devices over a network.



OSC
Open Sound Control

OSC-type messages can be passed back and forth using the classic network architecture type: **client/server**.

Typically clients send messages to the server and servers receive and execute OSC messages.



OSC messages

OSC
Open Sound Control

OSC messages are composed of three parts:

address pattern is a string that specifies the address relative to the data that is transmitted

example: **/squarewave/parzial/one**

type tag string specifies the data type of each argument

example: **float** (number with a decimal place)

arguments are the parameter values contained in the message

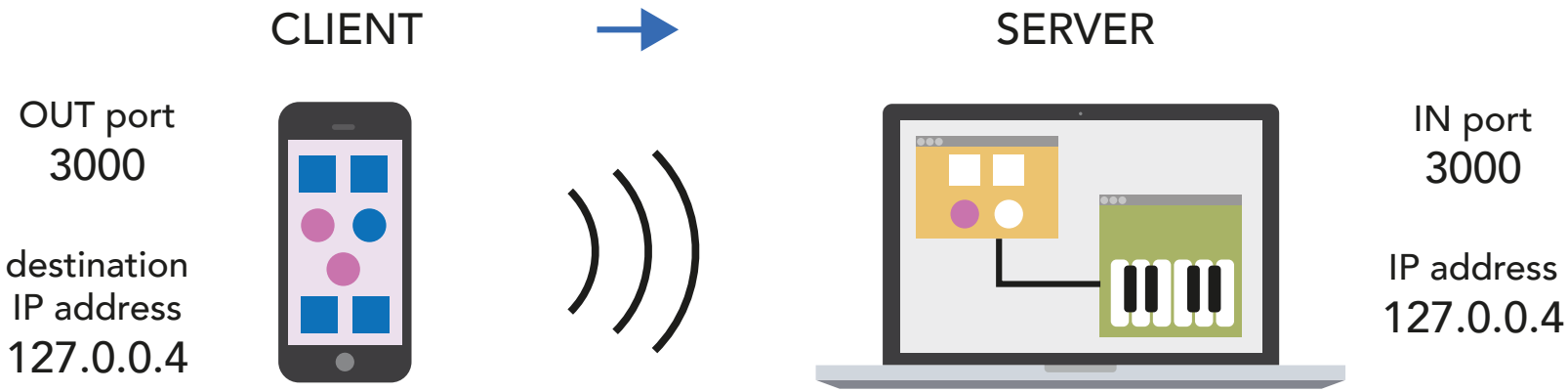
example: **440.5**

OSC messages

OSC
Open Sound Control

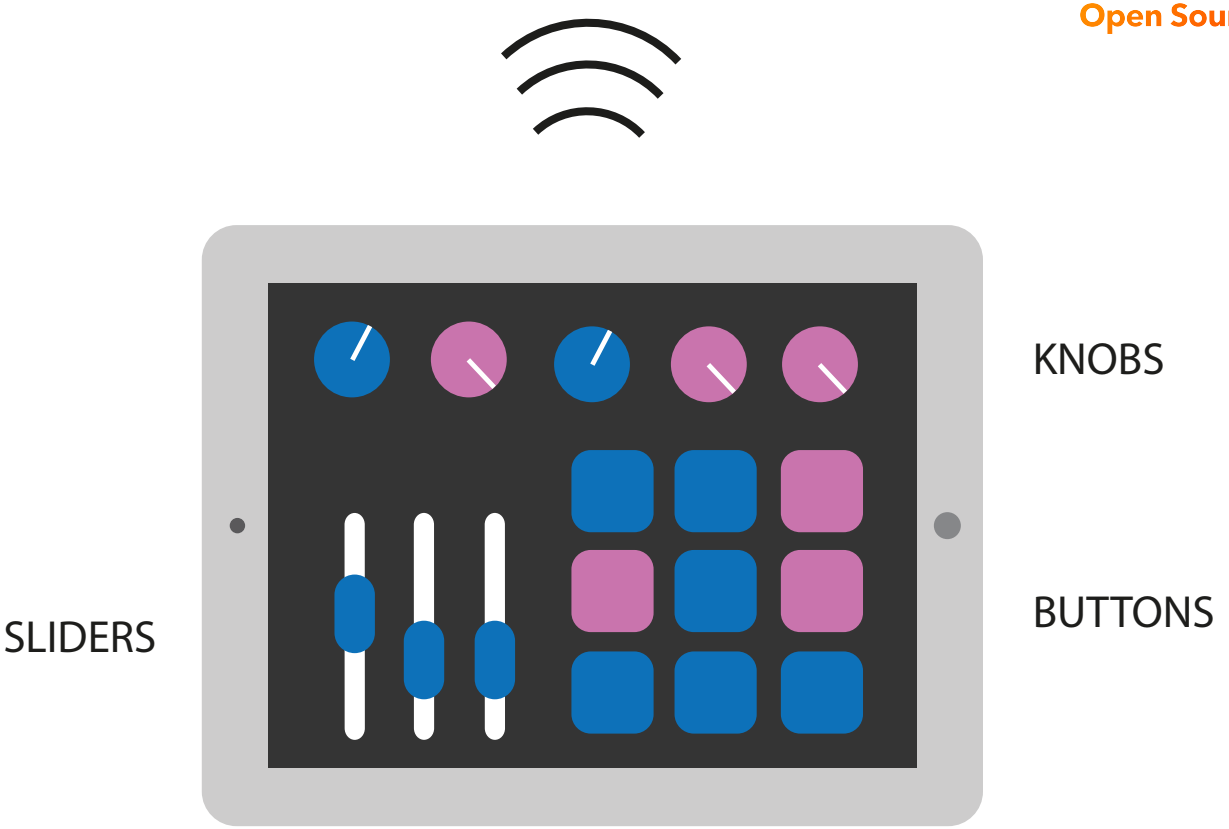
For clients to send OSC messages, they need to set the destination/ server **IP addresses**.

Both clients and servers should also be configured with a send and a receive **port**.



OSC messages

OSC
Open Sound Control

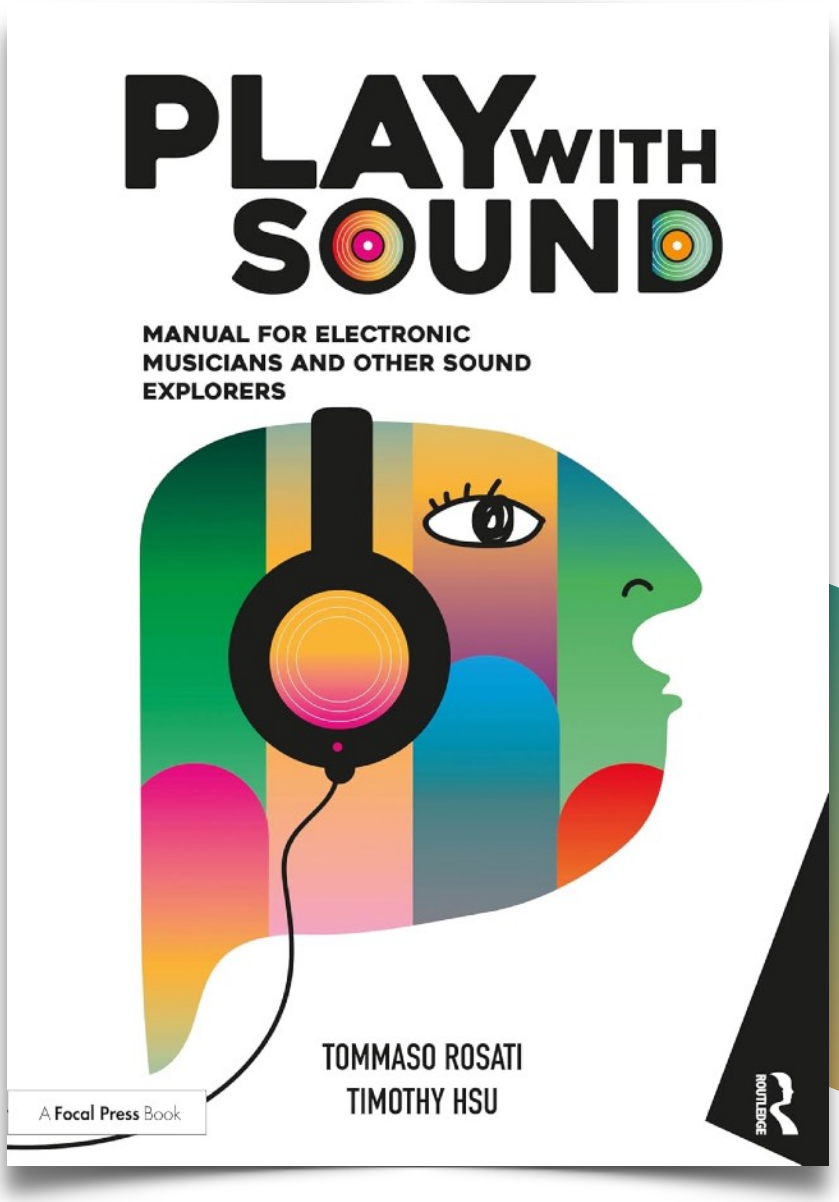


OSC is popular on smartphones and tablets, transmitting data via Wi-Fi. OSC-compatible apps let users create custom control interfaces with sliders, knobs, and buttons. These apps can also transmit OSC parameters based on the acceleration and rotation sensors within smartphones or tablets for live electronics performance.

OSC messages

OSC
Open Sound Control

OSC communicates data quickly over networks and has higher resolution than MIDI, using 32-bit processing for more flexibility. However, it is less ubiquitous than MIDI and requires specifying the address and port for device connections.



THE BOOK IS NOW AVAILABLE!