MIDI System-Exclusive Guide

for

MPOT32

firmware version 2.0

www.midi-hardware.com

Roman Sowa 2012

Table of contents

MIDI controller boards described here are products of:
"MIDI-hardware" Roman Sowa
ul. Azotowa 15B, 41-503 Chorzów, Poland
phone +48 532 425 835
email info@midi-hardware.com
www.midi-hardware.com, www.midimplant.com

boards made and manual printed in Poland, EU

1 Overview

The MPOT32 can be programmed in terms of user settings (not firmware) by MIDI System-Exclusive messages. General syntax of MIDI System-Exclusive received by MPOT32 as follows. All numbers in this guide are in hexadecimal format (hex), as this is commonly used by most software capable of generating System-Exclusive messages. Make sure your Sys-Ex editor is working with such numbers before you start.

1. Sys-Ex header: F0
2. manufacturer ID for MIDI-hardware.com: 00 20 7A
3. product ID for MPOT32: 03
4. input ID: one byte in range 00 .. 63
5. the command: one byte in range 01 .. 11
6. command's parameters, dependent on what command was used
7. Sys-Ex footer: F7

An example Sys-Ex string might look like this:
F0 00 20 7A 03 02 02 01 01 03 F7

2 The mysterious *input ID*

The input ID byte determines which controller - keyboard, or potentiometer, will be affected by sending this Sys-Ex. Lowest ID numbers are reserved for keyboards, that can be added by connecting contact scanner to 4-pin scanner input of MPOT32. All numbers above point to specific analog inputs, internal ones of MPOT32, and up to 64 added by connected scanners. Those can be POT12, LITSW, KEYPAD, and LCD modules. Lowest number is always associated with the scanner connected the closest to MPOT32, and every next one takes higher and higher numbers. Interlacing contact scanners and pot scanners does not change anything regardless of how they are connected.

here are the input IDs for MPOT32:

- 00 and 01: keyboards connected with contact scanners
- 02 and 03: upper splits of those keyboards
- 04 to 23: each of 32 analog inputs on MPOT32 board
- 24 to 64: each of 64 analog or special inputs in chain of connected scanners

Let's say you have MPOT32 and a chain of scanners: BBS-1K, POT12, POT12, LITSW. So the inputs on the first POT12 in chain have input IDs from 24 to 30 (hex), inputs of 2nd POT12 have input IDs from 31 to 3C (hex), and LITSW, if configured in CC value mode, will occupy IDs from 3D to 54. If you remove one of POT12 from this chain, lowest 12 inputs of LITSW will takeover functionality (events and channels) of last POT12. This is because IDs are always counted over all existing inputs.

3 Commands description

All available commands in MPOT32 version 2.0 are:

- **01** - starting note, with 1 parameter byte
- **02** - channel and event, with 3 parameter bytes, one for channel, and 2 for event (MSB and then LSB)
- **05** - split point with 1 parameter byte
- **09** - special settings, with 1 parameter byte
- **10** - write 32 Program Change numbers as "patch recall"

3.1 Starting note command - 01

F0 00 20 7A 03 <input ID> 01 <starting note> F7

this is available only for 4 possible splits of keyboards, so input ID can be only form 00 to 03). It sets the starting note at given number. So if you want standard lowest C in 5-octave keyboard, the *<starting note>* parameter should be 24 (hex). Any other value from 00 to 7F is possible.
Examples:
F0 00 20 7A 03 00 01 24 F7 - set typical range for 5-octave keyboard in 1st scanner
F0 00 20 7A 03 02 01 0C F7 - set upper split of 1st keyboard 2 octaves lower

3.2 Channel and Event - 02

F0 00 20 7A 03 <input ID> 02 <channel> <event MSB> <event LSB> F7

this is available to all controls, so every keboard split and pot can have different setting. The <channel> is given in straingt channel number from 1 to 16, which in hexadecimal means from 01 to 10. Then follow 2 bytes of event, there must be always 2, even for numbers below 7F. This event number is exactly the same as in the manual for MPOT32, section 5.2. Here's the summary:

from: F0 00 20 7A 03 <input ID> 02 <channel> 00 00 F7
to: F0 00 20 7A 03 <input ID> 02 <channel> 00 7F F7 - Control Change
F0 00 20 7A 03 <input ID> 02 <channel> 01 00 F7 - Pitch Bend
F0 00 20 7A 03 <input ID> 02 <channel> 01 01 F7 - Program Change
F0 00 20 7A 03 <input ID> 02 <channel> 01 02 F7 - Channel After Touch
F0 00 20 7A 03 <input ID> 02 <channel> 01 03 F7 - note on/off
F0 00 20 7A 03 <input ID> 02 <channel> 01 04 F7 - note on
F0 00 20 7A 03 <input ID> 02 <channel> 01 05 F7 - note off
F0 00 20 7A 03 <input ID> 02 <channel> 01 06 F7 - one-touch patch recall
F0 00 20 7A 03 <input ID> 02 <channel> 01 07 F7 - CC keyboard
F0 00 20 7A 03 <input ID> 02 <channel> 01 08 F7 - channel shift for all

F0 00 20 7A 03 <input ID> 02 <channel> 01 09 F7 - small transpozer
F0 00 20 7A 03 <input ID> 02 <channel> 01 0A F7 - big transpozer
F0 00 20 7A 03 <input ID> 02 <channel> 01 0B F7 - velocity of all notes
F0 00 20 7A 03 <input ID> 02 <channel> 01 0C F7 - NI's B4 chorus/vibrato
F0 00 20 7A 03 <input ID> 02 <channel> 01 0D F7 - MidiTzer stops control
F0 00 20 7A 03 <input ID> 02 <channel> 01 0E F7 - Ahlborn Archive stops
F0 00 20 7A 03 <input ID> 02 <channel> 01 0F F7 - Ahlborn Organ stops
F0 00 20 7A 03 <input ID> 02 <channel> 01 10 F7 - Program selector (1 of 12)
F0 00 20 7A 03 <input ID> 02 <channel> 01 11 F7 - Bank selector (1 of 10)
F0 00 20 7A 03 <input ID> 02 <channel> 01 12 F7 - Ahlborn common functions
F0 00 20 7A 03 <input ID> 02 <channel> 01 13 F7 - 3-note chord collect
F0 00 20 7A 03 <input ID> 02 <channel> 01 14 F7 - dual note (layered)
F0 00 20 7A 03 <input ID> 02 <channel> 01 15 F7 - set velocity & save
F0 00 20 7A 03 <input ID> 02 <channel> 01 16 F7 - semitone up
F0 00 20 7A 03 <input ID> 02 <channel> 01 17 F7 - semitone down
F0 00 20 7A 03 <input ID> 02 <channel> 01 18 F7 - octave up
F0 00 20 7A 03 <input ID> 02 <channel> 01 19 F7 - octave down
F0 00 20 7A 03 <input ID> 02 <channel> 01 1A F7 - channel up
F0 00 20 7A 03 <input ID> 02 <channel> 01 1B F7 - channel down
F0 00 20 7A 03 <input ID> 02 <channel> 01 1C F7 - global semitone up
F0 00 20 7A 03 <input ID> 02 <channel> 01 1D F7 - global semitone down
F0 00 20 7A 03 <input ID> 02 <channel> 01 1E F7 - global octave up
F0 00 20 7A 03 <input ID> 02 <channel> 01 1F F7 - global octave down
F0 00 20 7A 03 <input ID> 02 <channel> 01 20 F7 - no transpoze note on/off
F0 00 20 7A 03 <input ID> 02 <channel> 01 21 F7 - reversed notes (off/on)
F0 00 20 7A 03 <input ID> 02 <channel> 01 22 F7 - program up (list of 64)
F0 00 20 7A 03 <input ID> 02 <channel> 01 22 F7 - program down (list of 64)

3.3 ***Split point - 05***

F0 00 20 7A 03 <input ID> 05 <key position> F7

this Sys-Ex string sets the split at specific point. Lowest key of the keyboard is selected by *<key position>* parameter equal 00, and rising up to 3F for the last key of any contact scanner. Setting split point with parameter = 00 means that one lowest key will be in 1st split, and all remaining keys in 2nd split.
To turn of the split, simply set it beyond connected keys range.
Examples:
F0 00 20 7A 03 00 05 18 F7 - split 1st keyboard after 2 octaves
F0 00 20 7A 03 01 05 3F F7 - turn off the split in 2nd keyboard

3.4 ***Special settings - 09***

F0 00 20 7A 03 <input ID> 09 <advanced command> F7

using 09 command gives access to most of advanced settings typically available from keypad's sequence #9x. Possible options to set are:

- analog inputs update rate, set for all analog inputs in one board (5-100ms)
- bitwise resolution of each analog input (1-7 bits, 2-128 levels)
- reset to factory defaults
- selected options of MiDisp and miniLCD setup

Analog inputs update rate, latency

from F0 00 20 7A 03 <input ID> 09 50 F7 (fastest)
to F0 00 20 7A 03 <input ID> 09 59 F7 (slowest)

To select update rate, the device ID must be within range belonging to the scanner (or MPOT32 itself), so for example update rate of MPOT32 internal inputs can be changed if you send this command to any ID from range 04 - 23 (hex). This setting is valid for all inputs in one board, so doing it for more than one input from this range makes no sense. To set different update rate in connected scanner, use appropriate ID from its range, in our exaple this is 24 - 30.
Actual scan rate is defined by specific *<advanced command>*. 50 means the fastest scan, while 59 is the slowest. 50-59 in hexadecimal means 80-89 decimal, go to MPOT32 manual section 4.1
Examples:
F0 00 20 7A 03 20 09 51 F7 - set 7ms pot latency for MPOT32 inputs
F0 00 20 7A 03 25 09 54 F7 - set 18ms pot latency for 1st POT12
F0 00 20 7A 03 33 09 59 F7 - set 100ms pot latency for 2nd POT12

Resolution of analog inputs

from F0 00 20 7A 03 <input ID> 09 3D F7 (1-bit)
to F0 00 20 7A 03 <input ID> 09 43 F7 (7-bit)

Bits of resolution can be adjusted for each input in the system individually, so here input ID points to specific one. Use guidelines from chapter 2 to find proper input ID. This setting is available for all inputs in MPOT32 and all scanners in the chain if they contain analog inputs.

Reset to factory defaults

F0 00 20 7A 03 00 09 4F F7

Send the above MIDI string to MPOT32, and it will turn back to default factory settings. That is all MIDI channels, events and splits, note velocity, as well as list of 64 user programmable Program Numbers. Analog inputs update rate and bits

resolution is restored only for the internal inputs of MPOT32, because those settings are actually stored in each board individually.

LCD modules setup

from F0 00 20 7A 03 <input ID> 09 08 F7 (1-bit)
to F0 00 20 7A 03 <input ID> 09 14 F7 (7-bit)

Before you send any SysEx regarding LCD module, it has to be selected first, and the only way to do it is to press its button, or temporarily short the switch input pads at the back of selected LCD module. Also keep in mind that all LCDs in chain share the same two input IDs, not like all other scanners.
Available settings are:
F0 00 20 7A 03 <input ID> 09 08 F7 - works with toggle switch
F0 00 20 7A 03 <input ID> 09 09 F7 - works with momentary button
F0 00 20 7A 03 <input ID> 09 12 F7 - display blink
F0 00 20 7A 03 <input ID> 09 14 F7 - show display ID

*3.5 **Patch Recall memory set - 10***

F0 00 20 7A 03 00 10 [32 bytes for lower bank] F7
F0 00 20 7A 03 01 10 [32 bytes for upper bank] F7

MPOT32 can store 64 Program Change numbers, that can be later recalled either by direct button hit, or in +/-1 sequence. You can quickly store them all in 2 parts, 32 each. The block of 32 bytes should contain numbers in range 00 - 7F, or in decimal 0-127, so you have to lower Program Change numbers (1-128) by one. You must ensure delay of at least 200ms between sending 2 banks of 32 numbers.
Example:
F0 00 20 7A 03 00 10 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 10 11 12 13 14 15 16 17 18 19 1A 1B 1C 1D 1E 1F F7 - write Program numbers from 1 to 32 in lower part of patch recall memory.