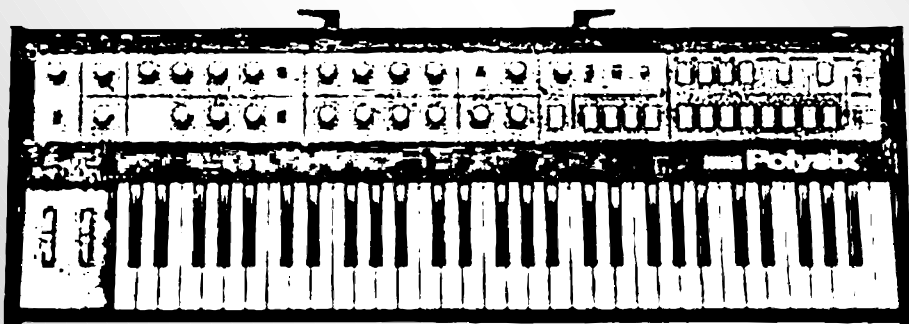


P6-KBD

MIDI Interface for KORG POLYSIX Keyboard

Model 8-431
ver. 1.1



MIDI SYSTEM EXCLUSIVE COMMUNICATION
Rev. 2





Contents	page
1 SYSTEM EXCLUSIVE COMMUNICATION	3
2 SYSEX MESSAGES STRUCTURE	3
2.1 "DEVICE ID" BYTE	3
2.2 "ADDRESS" BYTE	3
2.3 "DATA" BYTE	4
2.3.1 DATA FOR "MIDI CHANNEL" PARAMETER	4
2.3.2 DATA FOR "KEY SHIFT" PARAMETER	4
2.3.3 DATA FOR "KEY PRIORITY" PARAMETER	4
2.3.4 DATA FOR "PITCH BEND RANGE" PARAMETER	4
2.3.5 DATA FOR "ARPEGGIO CLOCK RATE" PARAMETER	4
2.4 "CHECKSUM" BYTE	5
3 SYSEX MESSAGE CREATION	5
3.1 EXAMPLE	5
3.2 CHECKSUM CALCULATOR	6
3.3 SYSEX MESSAGES GENERATOR	6



1 SYSTEM EXCLUSIVE COMMUNICATION

P6-KBD interface disposes of system of System Exclusive communication which enables to receive a SysEx Messages for changes of all parameters of the interface.

Software generator for simple creation of SysEx messages for control of P6-KBD is available on manufacturer's web site www.chd-el.cz. The generator is based on Java scripts. Any message for the interface setting described below can be created with help of this generator (see Owner's Manual).

2 SYSEX MESSAGES STRUCTURE

P6-KBD recognizes own specific messages for changes of parameters setting with this structure:

[hex]	[bin]	byte
F0	11110000	Start SysEx
00	00000000	Manufacturer ID (always 3 bytes)
20	00100000	
21	00100001	
ii	0iiiiiii	Device ID
55	01010101	Model ID (P6-KBD)
aa	0aaaaaaa	Address
dd	0ddddddd	Data byte
xx	0xxxxxxxxx	Checksum
F7	11110111	End SysEx

2.1 "DEVICE ID" BYTE

"Device ID" byte is equal to number of just active MIDI channel (00h for channel Nr. 1, 01h for channel Nr. 2 etc.).

Any of value from 00h to 0Fh of the Device ID is accepted if OMNI mode is active.

Next possible value of the Device ID is 7Fh - universal Device ID recognized whenever independently on just active MIDI channel number or OMNI mode.

For all others values of Device ID (from 10h to 7Eh), the message is evaluated as invalid and it is ignored by the interface.

2.2 "ADDRESS" BYTE

"Address" byte specifies parameter to be changed. Allowed address range is from 00h to 04h:

- aa = 00h** → **MIDI Channel** – temporary change of receiving MIDI channel number
- aa = 01h** → **Key Shift** – temporary change of MIDI keyboard transpose
- aa = 02h** → **Key Priority** – temporary change of MIDI key priority parameter
- aa = 03h** → **Pitch Wheel Range** – temporary change of MIDI pitch bend range in edit buffer
- aa = 04h** → **Arpeggio Clock Rate** – temporary change of source of arpeggiator clock

If the address is out of range 00h to 04h, the message is evaluated as invalid and it is ignored by the interface.



2.3 “DATA” BYTE

Data byte specifies new value for parameter selected by **aa** address. Its value depends on required value of the parameter. For values out of acceptable range (see below), the SysEx message is evaluated as invalid and it is ignored by the interface.

2.3.1 DATA FOR “MIDI CHANNEL” PARAMETER

If **aa** Address is equal to 00h, **dd** Data byte specifies new number of receiving MIDI channel. Valid value of **dd** byte is from 00h to 10h.

Values from 00h to 0Fh represent the MIDI channel numbers from 1 to 16. When the value is equal to 10h, OMNI mode will be active.

2.3.2 DATA FOR “KEY SHIFT” PARAMETER

If **aa** Address is equal to 01h, **dd** Data byte specifies keyboard transpose – it means relation between MIDI Note number and assigned key. Valid value of **dd** byte is from 00h to 67h.

The value specifies number of MIDI note which will be assigned to first key from left (i.e. to lowest C) of the instrument’s keyboard.

2.3.3 DATA FOR “KEY PRIORITY” PARAMETER

If **aa** Address is equal to 02h, **dd** Data byte specifies selected key priority mode. Valid value of **dd** byte is from 00h to 03h.

“Last Key” priority is selected for value equal to 00h, “Higher Key” priority is selected for value equal to 01h, “Lower Key” priority is selected for value equal to 02h and “None” priority is selected for value equal to 03h.

2.3.4 DATA FOR “PITCH BEND RANGE” PARAMETER

If **aa** Address is equal to 03h, one **dd** Data byte adjusts maximal range of MIDI Pitch Wheel (Pitch Bend) controller. Valid value of **dd** byte is from 00h to 18h.

For value 00h, Pitch Bend controller will be inactive – its movement will have no effect. Values from 01h to 18h represent allowed range of Pitch Wheel controller in semitones (from ± 1 to ± 24 semitones).

2.3.5 DATA FOR “ARPEGGIO CLOCK RATE” PARAMETER

If **aa** Address is equal to 04h, **dd** data byte specifies selected clock source / rate for instrument’s arpeggiator. Valid value of **dd** byte is from 00h to 7Fh.

If the value is equal to 00h, internal tempo generator is selected. For values from 01h to 7Fh, the tempo is derived from MIDI Clock (see Owner’s Manual for details).



2.4 “CHECKSUM” BYTE

xx checksum byte confirms validity of System Exclusive message. It must be calculated so that seven-bit sum of bytes from “Model ID” to “Checksum” is equal to zero¹ (i.e. checksum is 7-bit complement of sum from “Model ID” to “Datablock” bytes).

If checksum byte is invalid, the message is evaluated as invalid and it is ignored by the interface.

3 SYSEX MESSAGE CREATION

3.1 EXAMPLE

Task:

Set the receiving MIDI channel to Nr. 1

Solution:

- We don't know what number of MIDI channel is just active so universal Device ID (7Fh) will be used,
- Address byte (parameter number) will be 00h,
- Requested new MIDI channel number is 1 so data byte value will be 00h:

```
Start SysEx :   F0h
Mfr ID :       00h 20h 21h
Device ID :    7Fh
Model ID :     55h
Address :      00h
Data :         00h
Checksum :     xx
End SysEx :    F7h
```

- Now the checksum byte has to be calculated as 7-bit complement of sum from Model ID to Data bytes:
00h – (55h + 00h + 00h) = 2Bh
- Form of whole required System Exclusive message is after checksum refilling:
F0h 00h 20h 21h 7Fh 55h 00h 00h 2Bh F7h

Immediately after that message is received, the interface starts to accept MIDI channel messages on MIDI channel Nr. 1.

¹ See chapter 3.2 for easy calculation of the Checksum byte.



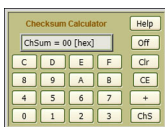
3.2 CHECKSUM CALCULATOR

If you want to create a SysEx message yourself, you need to calculate the 'Checksum' byte. This is difficult for most musicians because calculation with hexadecimal / binary numbers is necessary. For easy calculation of the checksum, special software **Checksum Calculator** is prepared.

The Checksum Calculator is based on Java scripts so it can run on any computer with web browser (Windows, OSX, etc.). Note that scripts and ActiveX elements must be enabled in the web browser for proper function of the calculator.

The Checksum Calculator is available at our website (<http://www.chd-el.cz>) on Support page.

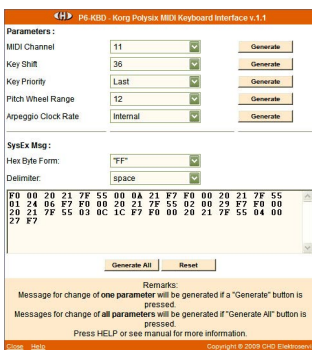
The Checksum Calculator works on-line or it can be downloaded to your computer and then launched from it.



3.3 SYSEX MESSAGES GENERATOR

As a support for the users we have made special software generator to create any SysEx messages to control the interface. Usage of this generator is very easy for any user. Please see Owner's Manual of the interface for detailed description of SysEx Messages Generator.

The SysEx Generator is available at our website (<http://www.chd-el.cz>).





Document :
8431_syxcom_rev2

Manufacturer :

CHD Elektroservis

Nad kundratkou 27, 19000 Praha 9, Czech Republic

info@chd-el.cz

www.chd-el.cz

