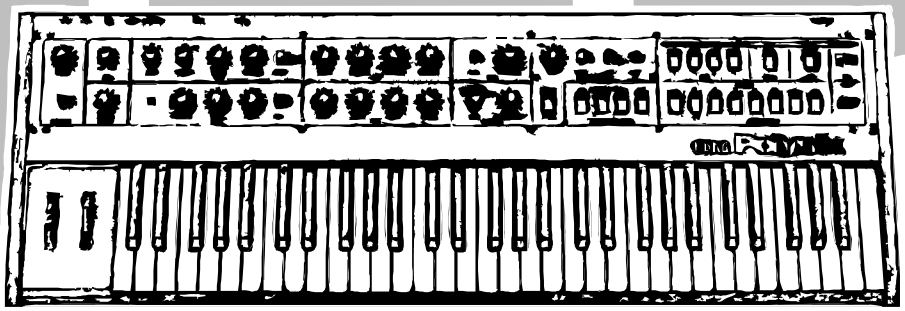
P6-M

MIDI Interface for Korg Polysix Synthesizer

Model 8-427
Version 1.0
(rev. 2)



MIDI SysEx Communication

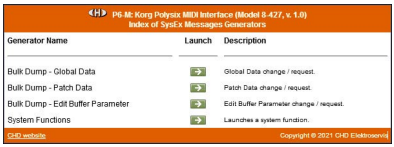
www.chd-el.cz

Contents Page

1 SUPPORT SOFTWARE

1.1 SYSEX MESSAGES GENERATOR

Fig. 1 – CHD SysEx Messages Generator Index

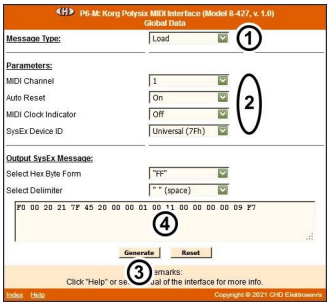


Special software generator for an easy creation of any necessary SysEx messages for control and programming of the P6-M interface can be used. Visit our web site and download the "**P6-M_syxgen.zip**" archive. Expand the archive to a selected folder on your computer's hard drive (i.e. "**P6-M_index.html**", "**P6-M_global.html**", "**P6-M_patch.html**", "**P6-M_edit.html**", "**P6-M_system.html**" and "**P6-M _help.html**" files and "**media**" sub-folder).

To launch the SysEx messages generator, simply open the "**P6-M_index.html**" file in your web browser (e.g. by clicking on the file icon). The generator window opens (fig. 1). Then click on arrow button of selected generator type.

1.1.1 GLOBAL DATA

Fig. 2 – Global Data window



Open the **Bulk Dump - Global Data** window (fig. 2) from the index table.

1.1.1.1 GLOBAL DATA SETTING

To change / adjust the Global Data values (global parameters in global memory bank):

1. Select "**Load**" in the Message Type select-box ①.
2. Select requested values of all global parameters ②.
3. Click the "**Generate**" button ③.
4. The required hexadecimal SysEx message is generated as a text in text field on bottom of the window ④.
5. Copy the text in clipboard (CTRL+C) and paste (CTRL+V) to a MIDI Sysex software.
6. Send the message to the interface.
7. Press interface's RESET button or switch the instrument off and then on again after a moment. Now the interface starts operation with the new global data settings.

1.1.1.2 GLOBAL DATA REQUEST

To request actual Global Data values (global parameters from global memory bank):

1. Select "**Request**" in the Message Type select-box ①.
2. Click the "**Generate**" button ③.
3. The required hexadecimal SysEx message is generated as a text in text field on bottom of the window ④.
4. Copy the text in clipboard (CTRL+C) and paste (CTRL+V) to a MIDI Sysex software.
5. Send the message to the interface.
6. The interface responds immediately – it sends "Bulk Dump - Global Data Load" type SysEx message.
7. Now you can save the received message in your computer for further use.

1.1.2 PATCH DATA

Open the **Bulk Dump - Patch Data** window (fig. 3) from the index table.

1.1.2.1 PATCH DATA SETTING

To change / adjust the Patch Data values (patch parameters and patch name in a patch memory bank):

1. Select "**Load**" in the Message Type select-box ①.
2. Select number of requested patch to be changed ②.
3. Select requested values of all patch parameters and name ③.
4. Click the "**Generate**" button ④.
5. The required hexadecimal SysEx message is generated as a text in text field on bottom of the window ⑤.
6. Copy the text in clipboard (CTRL+C) and paste (CTRL+V) to a MIDI Sysex software.
7. Send the message to the interface.
8. The interface will work with the new patch data till after the changed patch is recalled.

1.1.2.2 PATCH DATA REQUEST

To request actual Patch Data values (patch parameters and patch name from a patch memory bank):

1. Select "**Request**" in the Message Type select-box ①.
2. Select number of requested patch ②.
3. Click the "**Generate**" button ④.
4. The required hexadecimal SysEx message is generated as a text in text field on bottom of the window ⑤.
5. **Copy** the text in clipboard (CTRL+C) and **paste** (CTRL+V) to a MIDI Sysex software.
6. Send the message to the interface.
7. The interface responds immediately – it sends "Bulk Dump - Patch Data Load" type SysEx message.
8. Now you can save the received message in your computer for further use.

1.1.3 EDIT BUFFER PARAMETERS

Open the **Bulk Dump – Edit Buffer Parameters** window (fig. 4) from the index table.

1.1.3.1 EDIT BUFFER PARAMETER SETTING

To change / adjust value of a parameter in edit buffer:

1. Select "**Load**" in the Message Type select-box ①.
2. Select new value of the necessary parameter ②.
3. Click the "**Generate**" button in the same row ③.
4. The required hexadecimal SysEx message is generated as a text in text field on bottom of the window ④.
5. Copy the text in clipboard (CTRL+C) and paste (CTRL+V) to a MIDI Sysex software.
6. Send the message to the interface.

Fig. 3 – Patch Data window

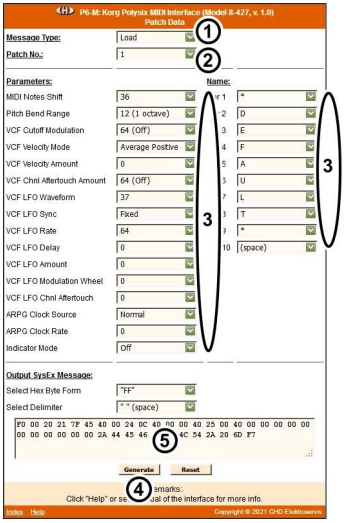
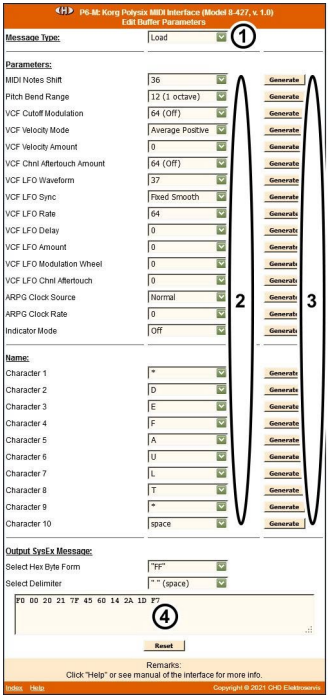


Fig. 4 – Edit Buffer Parameters window



7. The interface will work with the new parameter value immediately. Note that the change is temporary only.

1.1.3.2 EDIT BUFFER PARAMETER REQUEST

To request actual value of a parameter from edit buffer:

1. Select "**Request**" in the Message Type select-box ①.
2. Click the "**Generate**" button in row of the requested parameter ③.
3. The required hexadecimal SysEx message is generated as a text in text field on bottom of the window ④.
4. Copy the text in clipboard (CTRL+C) and paste (CTRL+V) to a MIDI Sysex software.
5. Send the message to the interface.
6. The interface responds immediately – it sends "Bulk Dump – Edit Buffer Parameter Load" type SysEx message.
7. Now you can save the received message in your computer for further use.

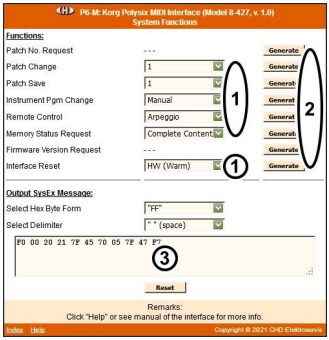
1.1.4 SYSTEM FUNCTIONS PROCESSING

Open the **System Functions** window (fig. 5) from the index table.

To execute a process with interface's presets etc. (System Functions section of the generator window):

1. Select value or type[1] of one requested system function ①.
2. Click the corresponding "**Generate**" button (in the same row) ②.
3. The required hexadecimal SysEx message is generated as a text in text field on bottom of the window ③.
4. Copy the text in clipboard (CTRL+C) and paste (CTRL+V) to a MIDI Sysex software.
5. Send the message to the interface.
6. The interface executes requested function immediately.

Fig. 5 – System Functions window

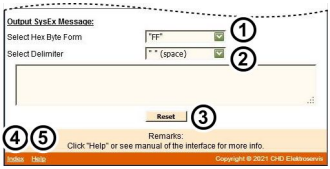


1.1.5 OTHER FUNCTIONS

Each of the generator windows include the same select / control elements on bottom of the window (fig. 6):

1. Select "**Hex Byte Form**" ① and "**Delimiter**" character ② as required for your MIDI SysEx software[2]. Default setting of the generator is optimized for the recommended utility (see chapter 1.2). However some DAW or MIDI SysEx utilities require different format of the generated text message.
2. To clear the text field and return all values to their defaults, click the "**Reset**" button ③.
3. "**Index**" link ④ returns to the index window.
4. "**Help**" link ⑤ opens new window with brief help.

Fig. 6 – Common controls



[1] Some functions don't have a select box – skip this step. For details about System Functions, see chapter 3.7.
[2] See the documentation of your DAW for required format. Form of hex bytes and delimiter (see chapter 1.1.5) must be set before a Generate button is pressed.

1.2 RECOMMENDED THIRD PARTY SOFTWARE

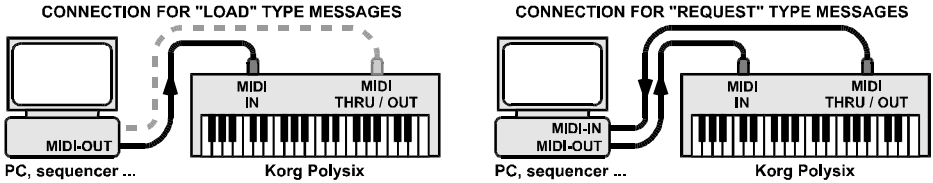
The hexadecimal MIDI SysEx messages created in a SysEx Generator (as described in chapter 1) are in plain text format. The text can not be saved as a *.syx or *.mid file directly, hence a text to SysEx capable utility or DAW is needed.

Pocket MIDI[3] is a utility that can be used to send the text as a SysEx message. It is a simple MIDI monitoring tool for both Windows and Mac OSX platforms. This is a freeware for commercial, non-profit or private use.

1.2.1 SETTING UP THE COMPUTER AND SOFTWARE

1. Download the **Pocket MIDI** utility at **https://www.morson.jp/pocketmidi-webpage/**
2. Install the utility in your computer.
3. Select the MIDI interface **Outputs** to device where the interface is connected to[4] (drop-down menu **Views → MIDI Settings → Input Port / Output Port**).
4. Connect the computer with the interface (accordingly to the fig. 7)[5].
5. Send and receive required SysEx messages to / from the interface

Fig. 7 – Connection for Bulk Dump

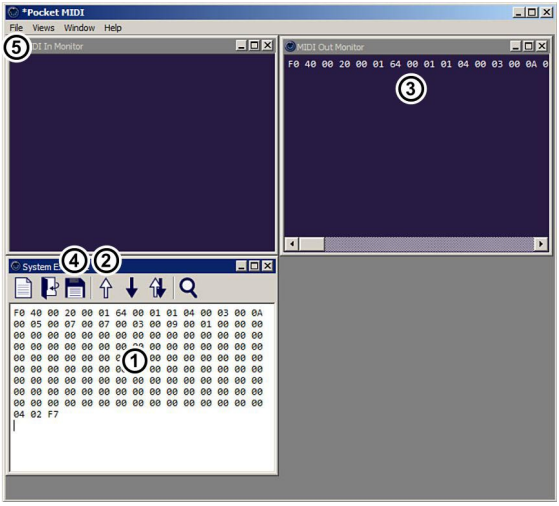


1.2.2 SEND THE TEXT SYSEX MESSAGE TO THE INTERFACE

Basic procedure to send any SysEx data to a MIDI device:

1. Generate required ("**Load**" or "**Function**" type) SysEx message in the SysEx Messages Generator (as described in Chapter 1)[6].
2. Copy the text in clipboard (CTR+C) and paste (CTRL+V) in "**System Exclusive**" window①.
3. Click the "↑" ("**Transmit**") arrow ② to send the data to the interface.
4. The sent message appears in the "**MIDI Out Monitor**" window ③.
5. Alternatively you can save the file for future use (either by the floppy icon ④ as a *.txt file or in the drop-down "**File**" menu ⑤ as a *.pocketmidi file).

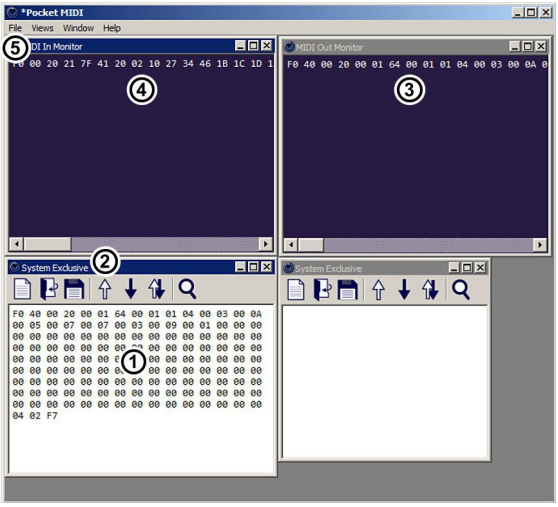
Fig. 8 – Pocket MIDI for data sending



[3] Pocket MIDI is Copyright © MORSON JAPAN Co.,Ltd. All rights reserved.
[4] Computer's MIDI interface must be active / switched on and all necessary MIDI drivers correctly installed.
[5] MIDI cable from interface's MIDI output to computer's MIDI input is not necessary for "Load" type messages but it can remain connected.
[6] The correct "**FF**" Hex Byte Form and "**' ' (space)**" for Delimiter are the initial values after the SysEx Generator is launched, so there is no need to change them for Pocket MIDI.

1.2.3 REQUEST THE SYSEX DATA FROM THE INTERFACE

Fig. 9 – Pocket MIDI for data receiving



Basic procedure to request a SysEx data from a MIDI device and to record the received data:

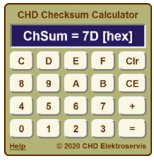
1. Generate required ("**Request**"[7] or "**Function**"[8] type) SysEx message in the SysEx Messages Generator (as described in Chapter 1)[9].
2. Copy the text in clipboard (CTR+C) and paste (CTRL+V) in "System Exclusive" window ①.
3. Click the "↑" ("**Transmit**") arrow ② to send the data to the interface.
4. The sent message appears in the "**MIDI Out Monitor**" window ③.
5. The interface responds with "**Load**" type SysEx message. The received message appears in the "**MIDI In Monitor**" window ④.
6. Click on the "**MIDI In Monitor**" window heading and save the data (File downdrop menu ⑤) as a *.pocketmidi file for future use.
7. Alternatively you can copy the data (CTRL+C) from "**MIDI In Monitor**" window ④ and paste (CTRL+V) to any text editor and save as a *.txt or document file.

1.2.4 ALTERNATIVE SOFTWARE SOLUTIONS

There are various DAW and MIDI utilites that can be used for the communication with P6-M, both commercial and free. Among the others:

- MIDI-OX - advanced MIDI tool for MS Windows computers: **http://www.midiox.com/**
- Bome SendSX - simple and effective MS Windows MIDI utility: **https://www.bome.com/products/sendsx**
- InerziaSysEx - advanced MIDI tool for Mac OSX: available on iTunes
- Steinberg Cubase Pro - DAW for both MS Windows and Mac OSX computers: **http://www.steinberg.net**
- Apple Logic Pro - Mac OSX DAW with full sysex support: **https://www.apple.com/logic-pro/**
- Cakewalk Sonar - MS Windows DAW with full sysex support: **http://www.cakewalk.com/**
- etc.

1.3 CHECKSUM CALCULATOR



If you need to calculate simply the checksum byte necessary for any SysEx message, you can use a special software **CHD Checksum Calculator**. The calculator is based on Java script so it can be launched under any platform (PC, MAC, ...) which includes a web browser. The Calculator with guide how to use it is available at our "Support" web page.

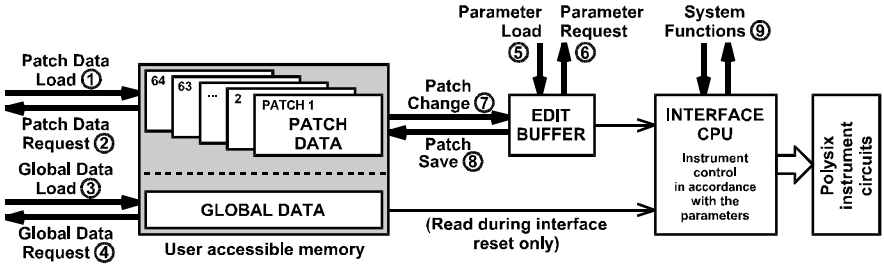
[7] I.e. **Bulk Dump – Global Data Request** or **Bulk Dump – Patch Data Request** or **Bulk Dump – Edit Buffer Parameter Request** message.
[8] I.e. **Patch No. Request** or **Memory Status Request** or **Firmware Version Request** message.
[9] The correct "**FF**" Hex Byte Form and "**' ' (space)**" for Delimiter are the initial values after the generator is launched, so there is no need to change them for Pocket MIDI.

2 SYSTEM EXCLUSIVE COMMUNICATION DESCRIPTION

2.1 SYSEX MESSAGES SYSTEM

P6-M interface uses System Exclusive communication to receive and to transmit a SysEx Messages for the control of all parameters in global and patch[10] memory banks of the interface. It is also possible to list an actual memory banks content. These data can be backed up in the PC. Certain system and testing functions can be launched by SysEx commands as well.

Fig. 10 – SysEx Messages system



① **Bulk Dump - Patch Data Load**
Response to the "Bulk Dump - Patch Data Request" message or change of the content of a patch in the internal interface's memory (chapter 3.4).

② **Bulk Dump - Patch Data Request**
Request for the patch data in a patch in the internal interface's memory (chapter 3.3).

③ **Bulk Dump - Global Data Load**
Response to the "Bulk Dump - Global Data Request" message or change of the content of global data memory bank in the internal interface's memory (chapter 3.2).

④ **Bulk Dump - Global Data Request**
Request for the content of global data memory bank in the internal interface's memory (chapter 3.1).

⑤ **Bulk Dump - Edit Buffer Parameter Load**
Response to the "Bulk Dump – Edit Buffer Parameter Request" message or change of a parameter value in the edit buffer. (chapter 3.6).

⑥ **Bulk Dump - Edit Buffer Parameter Request**
Request for a parameter value in the edit buffer (chapter 3.5).

⑦ **System Function - Patch Change**
Command for a change of actual interface patch number (chapter 3.7.2).

⑧ **System Function - Patch Save**
Command for saving of the data from the edit buffer to an interface patch memory (chapter 3.7.3).

⑨ **System Functions - Others**
Launches a system or a service function. Exact procedure is specified by the values of the "Address" and "Data" bytes (chapter 3.7).

[10] Note that all user patches can be named (up to 10 characters are available) although the name is not displayed on the interface any way.

2.2 SYSEX MESSAGES STRUCTURE

P6-M receives / transmits own specific messages with the following structure:

[hex]	[bin]	byte
F0	11110000	Start SysEx
00 20 21	00000000 00100000 00100001	Manufacturer ID (3 byte)
ii	0iiiiiii	Device ID (00h ~ 0Fh or "Universal ID" 7Fh)
45	01000101	Model ID
cc	0ccc0000	Command (10h / 20h / 30h / 40h / 50h / 60h / 70h)
aa	0aaaaaaa	Address
dd . . . dd	0ddddddd . . . 0ddddddd	Datablock (0 ~ 16 data bytes in dependence on the message type)
xx	0xxxxxxx	Checksum
F7	11110111	End SysEx

2.2.1 "SOX" BYTE

Each SysEx message must begin with the **F0h** "Start of SysEx" byte.

2.2.2 "MANUFACTURER ID" BYTES

The three Manufacturer ID bytes are always **00h 20h 21h**.

2.2.3 "DEVICE ID" BYTE

The device ID byte identifies the MIDI device exactly. It is equal to a number of an active MIDI channel (00h for channel Nr. 1, 01h for channel Nr. 2 etc. up to 0Fh for channel Nr. 16). The next possible value of the Device ID is 7Fh – universal Device ID always accepted by the interface regardless on an active MIDI channel number. For all other values of the Device ID (from 10h to 7Eh) the message is considered to be invalid and it is ignored by the interface.

2.2.4 "MODEL ID" BYTE

The Model ID byte is always **45h** for the P6-M interface.

2.2.5 "COMMAND" BYTE

The "**cc**" command byte specifies the Bulk Dump message type or the interface function type (see chapter 3). It defines the interface activity after the whole message is received.

2.2.6 "ADDRESS" BYTE

The "**aa**" address byte specifies the memory area or selects the system function accordingly to a type of the SysEx message – see below.

2.2.7 "DATA" BYTES

The block of data bytes "**dd…dd**" contains the values of a parameters or a function defined by the SysEx message type. The number of bytes in the data block "**dd…dd**" is a different accordingly to a type of the SysEx message. It consists from none to 30 data bytes.

2.2.8 "CHECKSUM" BYTE

The "**xx**" checksum byte confirms validity of the System Exclusive message. It must be calculated as 7-bit complement of the sum of bytes from 'Model ID' to 'Data' (by other words, seven-bit sum of bytes from the "Model ID" to "Checksum" must be equal to zero). If the checksum byte is invalid, the whole message is invalid too and it is ignored by the interface.

2.2.9 "EOX" BYTE

Each SysEx message must be ended with the **F7h** "End of SysEx" byte.

3 SYSEX COMMANDS OVERVIEW

Valid values of the "**cc**" command byte are:

Table 1 – Commands			
"cc" Byte	**Command Name**	**"cc" Byte**	**Command Name**
00h	Service (user inaccessible – see chpter 3.8)	40h	Bulk Dump - Patch Data Load
10h	Bulk Dump - Global Data Request	50h	Bulk Dump - Edit Buffer Parameter Request
20h	Bulk Dump - Global Data Load	60h	Bulk Dump - Edit Buffer Parameter Load
30h	Bulk Dump - Patch Data Request	70h	System Functions

3.1 BULK DUMP - GLOBAL DATA REQUEST

The "Bulk Dump - Global Data Request" command ("**cc**"=10h, ④ on fig. 10) is request for a content of the global data memory bank. It launches the listing of the data from the global memory bank (i.e. global parameters stored in user memory).

Form of the message:

[hex]	**[bin]**	**byte**
F0	11110000	Start SysEx
00 20 21	00000000 00100000 00100001	Manufacturer ID (3 byte)
ii	0iiiiiii	Device ID (00h ~ 0Fh or “Universal ID” 7Fh)
45	01000101	Model ID
10	00010000	Command (Bulk Dump - Global Data Request)
00	00000000	Address (always 00h)
xx	0xxxxxxx	Checksum
F7	11110111	End SysEx

The "**aa**" address is always 00h. The message is considered to be invalid for all other values and it is ignored by the interface.

Data bytes “**dd**“ are not used in this type of the SysEx message.

Immediately after the “Bulk Dump - Global Data Request“ message reception, the interface sends the "Bulk Dump - Global Data Load" message (see chapter 3.1) as a reply - data of all parameters of system memory bank are included in this message.

3.2 BULK DUMP - GLOBAL DATA LOAD

Transmitting the "Bulk Dump - Global Data Load" message ("**cc**"=20h, ③ on fig. 10) to the interface changes the content of global memory bank saved in the user interface’s memory. This SysEx message is also sent from the interface as a reply to the "Bulk Dump - Global Data Request" message (see chapter 3.1).

Form of the message:

[hex]	**[bin]**	**byte**
F0	11110000	Start SysEx
00 20 21	00000000 00100000 00100001	Manufacturer ID (3 byte)
ii	0iiiiiii	Device ID (00h ~ 0Fh or “Universal ID” 7Fh)
45	01000101	Model ID
20	00100000	Command (Bulk Dump - Global Data Load)
00	00000000	Address (always 00h)
d1 . . . d4	0ddddddd . . . 0ddddddd	Data block (4 data bytes)
xx	0xxxxxxx	Checksum
F7	11110111	End SysEx

The "**aa**" address is always 00h. The message is considered to be invalid for all other values and it is ignored by the interface.

The data block "**d1…d4**" always contains 4 bytes with values of the global parameters:

Table 2 – Global Data					
Byte order	**Parameter**	**Valid Range**	**Byte order**	**Parameter**	**Valid Range**
1st	MIDI Channel	0 ~ 15	3rd	MIDI Clock Indicator	0 ~ 1
2nd	Auto Reset	0 ~ 1	4th	SysEx Device ID	0 ~ 17

If received value of any parameter exceeds the valid range listed above, the value is limited to allowed range. For detailed description of the global parameters see the Owner's Manual.

Remark: *Note that the interface works with changed parameters till after next interface reset!*

3.3 BULK DUMP - PATCH DATA REQUEST

The "Bulk Dump - Patch Data Request" command ("**cc**"=30h, ② on fig. 10) is a request for a content of a preset in the preset memory bank. It launches the listing of the data from a preset of preset memory bank stored in user memory.

Form of the message:

[hex]	[bin]	byte
F0	11110000	Start SysEx
00 20 21	00000000 00100000 00100001	Manufacturer ID (3 byte)
ii	0iiiiiii	Device ID (00h ~ 0Fh or "Universal ID" 7Fh)
45	01000101	Model ID
30	00110000	Command (Bulk Dump - Patch Data Request)
aa	0xxxxxxx	Address (00h ~ 3Fh → patch number 1 ~ 64)
xx	0xxxxxxx	Checksum
F7	11110111	End SysEx

The address "**aa**" is from 00h to 3Fh. It corresponds to the number from 1 to 64 of a requested preset.

Databytes "**dd**" are not used in this type of the SysEx message.

Immediately after the "Bulk Dump - Patch Data Request" message reception, the interface sends the "Bulk Dump - Patch Data Load" message (see chapter 3.1) as a reply - data of all parameters of requested patch from patch memory bank are included in this message.

3.4 BULK DUMP - PATCH DATA LOAD

Transmitting the "Bulk Dump - Patch Data Load" message ("**cc**"=40h, ① on fig. 10) to the interface changes the content of a patch in the patch memory bank saved in the user interface's memory. This SysEx Msg message is also sent from the interface as reply to the "Bulk Dump - Patch Data Request" message (see chapter 3.3).

Form of the message:

[hex]	[bin]	byte
F0	11110000	Start SysEx
00 20 21	00000000 00100000 00100001	Manufacturer ID (3 byte)
ii	0iiiiiii	Device ID (00h ~ 0Fh or "Universal ID" 7Fh)
45	01000101	Model ID
40	01000000	Command (Bulk Dump - Patch Data Load)
aa	0xxxxxxx	Address ((00h ~ 3Fh → patch number 1 ~ 64)
d1 . . . d30	0ddddddd . . . 0ddddddd	Data block (30 data bytes)
xx	0xxxxxxx	Checksum
F7	11110111	End SysEx

The "**aa**" address is from 00h to 3Fh. It corresponds to the number from 1 to 64 of a requested preset.

The data block "**d1…d30**" always contains 30 bytes with values of the patch parameters and the patch name:

Table 3 – Patch Data					
Byte order	**Parameter**	**Valid Range**	**Byte order**	**Parameter**	**Valid Range**
1st	MIDI Notes Shift	0 ~ 67	16th	Indicator Mode	0 ~ 3
2nd	Pitch Bend Range	0 ~ 24	17th	[reserve]	0
3rd	VCF Cutoff Modulation	0 ~ 127	18th	[reserve]	0
4th	VCF Velocity Mode	0 ~ 3	19th	[reserve]	0
5th	VCF Velocity Amount	0 ~ 127	20th	[reserve]	0
6th	VCF Chnl Aftertouch Amount	0 ~ 127	21th	Name – Character 1 Code	32 ~ 126 *)
7th	VCF LFO Waveform	0 ~ 63	22th	Name – Character 2 Code	32 ~ 126 *)
8th	VCF LFO Sync	0 ~ 3	23th	Name – Character 3 Code	32 ~ 126 *)
9th	VCF LFO Rate	0 ~ 127	24th	Name – Character 4 Code	32 ~ 126 *)
10th	VCF LFO Delay	0 ~ 127	25th	Name – Character 5 Code	32 ~ 126 *)
11th	VCF LFO Amount	0 ~ 127	26th	Name – Character 6 Code	32 ~ 126 *)
12th	VCF LFO Modulation Wheel	0 ~ 127	27th	Name – Character 7 Code	32 ~ 126 *)
13th	VCF LFO Chnl Aftertouch	0 ~ 127	28th	Name – Character 8 Code	32 ~ 126 *)
14th	ARPG Clock Source	0 ~ 3	29th	Name – Character 9 Code	32 ~ 126 *)
15th	ARPG Clock Rate	0 ~ 127	30th	Name – Character 10 Code	32 ~ 126 *)
***)** See table 4 below.					

If received value of any parameter exceeds the valid range listed above, the value is limited to allowed range. For detailed description of the parameters see the Owner's Manual.

Table 4 – Valid ASCII Codes for Patch Name Characters													
Code	**Char**	**Code**	**Char**	**Code**	**Char**	**Code**	**Char**	**Code**	**Char**	**Code**	**Char**	**Code**	**Char**
32	space	46	.	60	<	74	J	88	X	102	f	116	t
33	!	47	/	61	=	75	K	89	Y	103	g	117	u
34	"	48	0	62	>	76	L	90	Z	104	h	118	v
35	#	49	1	63	?	77	M	91	[	105	I	119	w
36	$	50	2	64	@	78	N	92	\	106	j	120	x
37	%	51	3	65	A	79	O	93	]	107	k	121	y
38	&	52	4	66	B	80	P	94	^	108	l	122	z
39	'	53	5	67	C	81	Q	95	_	109	m	123	{
40	(	54	6	68	D	82	R	96	`	110	n	124	|
41	)	55	7	69	E	83	S	97	a	111	o	125	}
42	*	56	8	70	F	84	T	98	b	112	p	126	~
43	+	57	9	71	G	85	U	99	c	113	q		
44	,	58	:	72	H	86	V	100	d	114	r		
45	-	59	;	73	I	87	W	101	e	115	s		

Remark: *Note that the interface works with changed parameters till after the changed patch is recalled!*

3.5 BULK DUMP - EDIT BUFFER PARAMETER REQUEST

The "Bulk Dump - Edit Buffer Parameter Request" command ("**cc**"=50h, ⑥ on fig. 10) is a request for value of a parameter[11] in the interface's edit buffer. It launches the listing of the parameter value from the edit buffer. Form of the message:

[11] Also patch name characters can be requested this way.

[hex]	[bin]	byte
F0	11110000	Start SysEx
00 20 21	00000000 00100000 00100001	Manufacturer ID (3 byte)
ii	0iiiiiii	Device ID (00h ~ 0Fh or "Universal ID" 7Fh)
45	01000101	Model ID
50	01010000	Command (Bulk Dump - Patch Data Load)
aa	0xxxxxxx	Address ((00h ~ 1Dh → parameter number 1 ~ 30)
xx	0xxxxxxx	Checksum
F7	11110111	End SysEx

The address "**aa**" is from 00h to 1Dh. It corresponds to the number of a requested parameter – see table 5 below.

Databytes "**dd**" are not used in this type of the SysEx message.

Immediately after the "Bulk Dump - Edit Buffer Parameter Request" message reception, the interface sends the "Bulk Dump - Edit Buffer Parameter Load" message (see chapter 3.6) as a reply – value of the requested parameter is included in this message.

3.6 BULK DUMP - EDIT BUFFER PARAMETER LOAD

Transmitting the "Bulk Dump - Edit Buffer Parameter Data Load" message ("**cc**"=60h, ⑤ on fig. 10) to the interface changes value of a parameter[12] in the interface's edit buffer[13]. This SysEx Msg message is also sent from the interface as reply to the "Bulk Dump - Edit Buffer Parameter Request" message (see chapter 3.5).

Form of the message:

[hex]	[bin]	byte
F0	11110000	Start SysEx
00 20 21	00000000 00100000 00100001	Manufacturer ID (3 byte)
ii	0iiiiiii	Device ID (00h ~ 0Fh or "Universal ID" 7Fh)
45	01000101	Model ID
60	01100000	Command (Bulk Dump - Patch Data Load)
aa	0xxxxxxx	Address ((00h ~ 1Dh → parameter number 1 ~ 30)
dd	0xxxxxxx	Data block (1 data byte → parameter value)
xx	0xxxxxxx	Checksum
F7	11110111	End SysEx

The address "**aa**" is from 00h to 1Dh. It corresponds to the number of a parameter (1 to 30) to be changed (see table 5):

Table 5 – Edit Buffer Parameters

"aa" Byte	Parameter	Valid Range	"aa" Byte	Parameter	Valid Range
00h	MIDI Notes Shift	0 ~ 67	09h	VCF LFO Delay	0 ~ 127
01h	Pitch Bend Range	0 ~ 24	0Ah	VCF LFO Amount	0 ~ 127
02h	VCF Cutoff Modulation	0 ~ 127	0Bh	VCF LFO Modulation Wheel	0 ~ 127
03h	VCF Velocity Mode	0 ~ 3	0Ch	VCF LFO Chnl Aftertouch	0 ~ 127
04h	VCF Velocity Amount	0 ~ 127	0Dh	ARPG Clock Source	0 ~ 3
05h	VCF Chnl Aftertouch Amount	0 ~ 127	0Eh	ARPG Clock Rate	0 ~ 127
06h	VCF LFO Waveform	0 ~ 63	0Fh	Indicator Mode	0 ~ 3
07h	VCF LFO Sync	0 ~ 3	10h	[reserve]	0
08h	VCF LFO Rate	0 ~ 127	11h	[reserve]	0

[12] Also patch name characters can be changed this way.
[13] For addresses from 0 to 15, the message is an equivalent for the "CC #16 ~ CC #31" standard MIDI commands – see owner's manual.

Table 5 – Edit Buffer Parameters - continue					
"aa" Byte	Parameter	Valid Range	"aa" Byte	Parameter	Valid Range
12h	[reserve]	0	18h	Name – Character 5 Code	32 ~ 126 *)
13h	[reserve]	0	19h	Name – Character 6 Code	32 ~ 126 *)
14h	Name – Character 1 Code	32 ~ 126 *)	1Ah	Name – Character 7 Code	32 ~ 126 *)
15h	Name – Character 2 Code	32 ~ 126 *)	1Bh	Name – Character 8 Code	32 ~ 126 *)
16h	Name – Character 3 Code	32 ~ 126 *)	1Ch	Name – Character 9 Code	32 ~ 126 *)
17h	Name – Character 4 Code	32 ~ 126 *)	1Dh	Name – Character 10 Code	32 ~ 126 *)
***)** See table 4 above.					

The data block "**dd**" always contains only one databyte which new value of the requested parameter (see tables 4 and 5). If received value of the parameter exceeds the valid range listed above, the value is limited to allowed range.

See the Owner's Manual of the interface for detailed description of the parameters and their functions.

3.7 SYSTEM FUNCTIONS

The "System Function" commands ("**cc**"=70h, ⑨ on fig. 10) are direct commands for the execution of a system function.

Form of the message:

[hex]	[bin]	byte
F0	11110000	Start SysEx
00 20 21	00000000 00100000 00100001	Manufacturer ID (3 byte)
ii	0iiiiiii	Device ID (00h ~ 0Fh or "Universal ID" 7Fh)
45	01000101	Model ID
70	01110000	Command (System Function)
aa	0xxxxxxx	Address (00h ~ 07h → the function type)
dd	0ddddddd	Data block (1 data byte)
xx	0xxxxxxx	Checksum
F7	11110111	End SysEx

The type of the function is defined by the "**aa**" address:

Table 6 – System Functions		
"aa" Byte	Function Name	Meaning
00h	Patch No. Request	Request for the number of actual (active) interface patch number.
01h	Patch Change	Command for the actual interface patch change.
02h	Patch Save	Command for saving of the data from the edit buffer to an interface patch memory
03h	Instrument Program Change	Command for a change of the instrument program (sound).
04h	Remote Control	Remote control of some instrument's panel buttons.
05h	Memory Status Request	Request for status of the interface memory protection or request for complete content of the interface memory.
06h	Firmware Version Request	Request for installed version of the interface firmware.
07h	Interface Reset	Command for execution of warm HW reset or total factory reset of the interface.

The datablock "**dd**" always contains only one databyte which specifies an activity of selected system function.

Immediately after the "Bulk Dump Prest Data Request" message reception, the interface executes the requested function.

3.7.1 PATCH NO. REQUEST

If the "**aa**" address is equal to 00h, the SysEx Msg is a request for the number of actual (active) interface patch number. The value of "**dd**" databyte can be any number from 0 to 127 (00h to 7Fh).

Immediately after the request is received, the interface transmits the information about the actual (active) interface patch number to the MIDI output. The form of transmitted SysEx message is following:

F0 00 20 21 ii 45 70 00 dd xx F7 [hex],

where the "**dd**" databyte is the number from 64 to 127 (40h to 7Fh) which defines the number of the actual (active) patch number (1 to 64). If the request is received immediately after the interface reset when none patch is still active, the "**dd**" databyte is 0 (00h).

3.7.2 PATCH CHANGE

If the "**aa**" address is equal to 01h, the SysEx message is a command for the actual interface patch change[14] (see ⑦ on fig. 10). The value of "**dd**" databyte is from 64 to 127 (40h to 7Fh) - it specifies the number of newly selected interface patch (1 to 64). For all other values, the message is considered to be invalid and it is ignored by the interface.

3.7.3 PATCH SAVE

If the "**aa**" address is equal to 02h, the SysEx Msg is a command for saving of the data from the edit buffer to an interface patch memory[15] (see ⑧ on fig. 10). The value of databyte "**dd**" can be from 64 to 127 (40h to 7Fh) - it specifies the number of a patch in the memory (1 to 64) to which the data will be stored. For all other values, the message is considered to be invalid and it is ignored by the interface.

3.7.4 INSTRUMENT PROGRAM CHANGE

If the "**aa**" address is equal to 03h, the SysEx Msg is a command for a change of instrument program (sound)[16]. The value of databyte "**dd**" can be from 0 to 32 (00h to 20h). Value 0 selects MANUAL mode. Values 1 to 32 select programs A-1 to D-8. For all other values, the message is considered to be invalid and it is ignored by the interface.

3.7.5 REMOTE CONTROL

If the "**aa**" address is equal to 04h, the SysEx Msg is a command for remote control of a button on instrument panel. The value of databyte "**dd**" can be from 0 to 4 (00h to 04h). For all other values, the message is considered to be invalid and it is ignored by the interface. The "**dd**" value specifies the controlled button: 00h for ARPEGGIO, 01h for POLY, 02h for UNISON, 03h for CHORD MEMORY and 04h for HOLD.

3.7.6 MEMORY STATUS REQUEST

If the "**aa**" address is equal to 05h, the SysEx Msg is a request for status of the interface memory protection or request for complete content of the interface memory. The value of databyte "**dd**" can only be 00h or 7Fh. For all other values, the message is considered to be invalid and it is ignored by the interface.

If the value of databyte "**dd**" is 00h, the message is request for memory protection status. Immediately after the request is received, the interface transmits the information about its status to the MIDI output. The form of transmitted SysEx message is following:

F0 00 20 21 ii 45 70 05 dd xx F7 [hex],

[14] This message is an equivalent for the "Program Change" standard MIDI command with values from 64 to 127.
[15] This message is similar to the "CC#119" standard MIDI command . The difference is that the CC#119 saves the data to just active patch number but the SysEx message can save the data to any of patches 1 to 64.
[16] This message is an equivalent for the "Program Change" standard MIDI command with values from 0 to 32 – see owner's manual.

where the "**dd**" databyte is 00h for unprotected memory and 7Fh for protected memory (by the "MEM" jumper plugged on interface board – see Owner's Manual).

If the value of databyte "**dd**" is 7Fh, the message is request for content of the memory. Immediately after the request is received, the interface transmits a series of 65 "**Bulk Dump – Load**" type SysEx messages with complete content of the interface memory (see table 7). These messages can be backed up / archived in your computer for further use. For example - the complete memory content can be subsequently processed by the "Patch Memory Organizer" support utility available at our web site.

Table 7 – Order of SysEx Messages

Order	Message Type	SysEx form
1	Bulk Dump – Patch Data Load / Patch No. 1	F0 00 20 21 ii 45 40 00 d1 … d30 xx F7 [hex]
2	Bulk Dump – Patch Data Load / Patch No. 2	F0 00 20 21 ii 45 40 01 d1 … d30 xx F7 [hex]
3 ~ 63	Bulk Dump – Patch Data Load / Patches No. 3 to 63	. . .
64	Bulk Dump – Patch Data Load / Patch No. 64	F0 00 20 21 ii 45 40 3F d1 … d30 xx F7 [hex]
65	Bulk Dump – Global Data Load	F0 00 20 21 ii 45 20 00 d1 … d4 xx F7 [hex]

"ii" → Device ID byte
"dx" → Data bytes (data block of 30 or 4 bytes)
"xx" → Checksum byte

3.7.7 FIRMWARE VERSION REQUEST

If the address "**aa**" is equal to 06h, the SysEx Msg is a request for the version number of the installed firmware. The value of databyte "**dd**" can only be 00h. For all other values the message is considered to be invalid and it is ignored by the interface.

Immediately after the request is received, the interface transmits the information about the firmware version to the MIDI output. The form of transmitted SysEx message is following:

F0 00 20 21 ii 45 70 06 dd xx F7 [hex],

where nibbles of the "**dd**" databyte are identifying installed firmware version. For example: Byte 10h (i.e. nibbles 01h, 00h) means the version "1.0".

3.7.8 INTERFACE RESET

If the address "**aa**" is equal to 07h, the SysEx Msg is the command for execution of the interface reset. The value of databyte "**dd**" can only be 00h or 7Fh. For all other values, the message is considered to be invalid and it is ignored by the interface.

If the value of databyte "**dd**" is 00h, the "Warm Hardware Reset" of the interface is executed. The interface is set to the same status (Stand-by) as when the instrument is switched on.

If the value of databyte "**dd**" is 7Fh, the complete "Factory Reset" is executed. The interface is initialized to the factory status (all parameters in both global and patch memory banks are set to the default values).

WARNING - All data stored by user in the internal memory are fully erased after the execution of the "Factory Reset" command. It is recommended to make a back up of all user data before the "Factory Reset" execution (see chapter 3.7.6) so that you don't lose them.

3.8 SERVICE FUNCTIONS

The "Service" command ("**cc**"= 00h) is used only for the debugging of the firmware and for the special service and test procedures of the interface hardware. The service functions are not expected to be used during regular operation so they are ignored in standard working modes of the interface.

4 EXAMPLES OF SYSEX MESSAGES

4.1 EXAMPLE 1 – CHANGE GLOBAL PARAMETERS

Task:

Set the global data of the interface to the following values:

- MIDI Channel:	0Fh (i.e. channel Nr. 16)
- Auto Reset:	01h (i.e. On)
- MIDI Clock Indicator:	00h (i.e. Off)
- SysEx Device ID:	11h (i.e. Universal ID 7Fh)

Solution:

- I don't know what number of the SysEx ID is actually active so the universal Device ID (**7Fh**) will be used.
- The command to be used is the "**Bulk Dump – Global Data Load**", the command byte will be **20h**.
- The address byte is always **00h** for this message type.
- Start of the SysEx Msg will be then:
  Start SysEx : **F0h**
  Mfr ID : **00h 20h 21h**
  Device ID : **7Fh**
  Model ID : **45h**
  Command : **20h**
  Address : **00h**
- All four parameter bytes of the datablock are given by the task. Whole datablock will be:
  0Fh 01h 00h 11h
- Now, the checksum must be calculated as 7-bit complement of the sum of bytes from 'Model ID' to 'Data':
  00h – (45h + 20h + 00h + 0Fh + 01h + 00h + 11h) = 7Ah
- The form of the whole required System Exclusive message is after the checksum and End SysEx byte refilling:
  F0h 00h 20h 21h 7Fh 45h 20h 00h 0Fh 01h 00h 11h 7Ah F7h

When the message is received, the values of all global parameters in the global memory bank will be set accordingly to the SysEx message data. Note that to use the newly stored data, the interface needs to be reset.

4.2 EXAMPLE 2 – CHANGE PATCH PARAMETERS

Task:

Set the patch data of the patch memory Nr. 1 to the following values:

- MIDI Notes Shift:	18h (i.e. from MIDI Note Nr. 24)
- Pitch Bend Range:	02h (i.e. ± 2 semitones)
- VCF Cutoff Modulation:	40h (i.e. Off)
- VCF Velocity Mode:	02h (i.e. Last Positive)
- VCF Velocity Amount:	40h (i.e. middle position)
- VCF Chnl Aftertouch Amount:	40h (i.e. Off)
- VCF LFO Waveform:	25h (i.e. Sine)
- VCF LFO Sync:	01h (i.e. MIDI)
- VCF LFO Rate:	68h (i.e. 4th Note)
- VCF LFO Delay:	00h (i.e. None)
- VCF LFO Amount:	20h (i.e. 25 %)
- VCF LFO Modulation Wheel:	40h (i.e. 50%)
- VCF LFO Chnl Aftertouch:	00h (i.e. Off)
- ARPG Clock Source:	01h (i.e. Fixed)
- ARPG Clock Rate:	5Ah (i.e. 8 Hz)
- Indicator Mode:	03h (i.e. MIDI Event)
- Patch Name:	50h 41h 54h 43h 48h 2Dh 31h 32h 33h 34h (i.e. "PATCH-1234")

Solution:

- I don't know what number of the SysEx ID is actually active so the universal Device ID (**7Fh**) will be used.
- The command to be used is the "**Bulk Dump – Patch Data Load**", the command byte will be **40h**.
- Data will be stored to preset memory Nr. 1, the address byte will be **00h**.
- Start of the SysEx Msg will be then:
  Start SysEx : **F0h**
  Mfr ID : **00h 20h 21h**
  Device ID : **7Fh**
  Model ID : **45h**
  Command : **40h**
  Address : **00h**
- All 16 parameter bytes and 10 name bytes of the datablock are given by the task. Whole datablock will be: **18h 02h 40h 02h 40h 40h 25h 01h 68h 00h 20h 40h 00h 01h 5Ah 03h 00h 00h 00h 00h 50h 41h 54h 43h 48h 2Dh 31h 32h 33h 34h** (17th, 18th, 19th and 20th bytes are reserve – value is always 00h)
- Now, the checksum must be calculated as 7-bit complement of the sum of bytes from 'Model ID' to 'Data': **00h – (45h + 40h + 00h + 18h + 02h + 40h + 02h + 40h + 40h + 25h + 01h + 68h + 00h + 20h + 40h + 00h + 01h + 5Ah + 03h + 00h + 00h + 00h + 00h + 50h + 41h + 54h + 43h + 48h + 2Dh + 31h + 32h + 33h + 34h) = 6Ch**
- The form of the whole required System Exclusive message is after the checksum and End SysEx byte refilling: **F0h 00h 20h 21h 7Fh 45h 40h 00h 18h 02h 40h 02h 40h 40h 25h 01h 68h 00h 20h 40h 00h 01h 5Ah 03h 00h 00h 00h 00h 50h 41h 54h 43h 48h 2Dh 31h 32h 33h 34h 6Ch F7h**

When the message is received, the values of all preset parameters and patch name in permanent patch memory Nr. 1 will be set accordingly to the SysEx message data. If the patch Nr. 1 is just set as active, it must be recalled again so that the interface will start to work with the newly stored data.

4.3 EXAMPLE 3 – CHANGE EDIT BUFFER

Task:

Change temporarily value of the "VCF LFO Waveform" parameter in edit buffer to "Saw - Rise".

Solution:

- I don't know what number of the SysEx ID is actually active so the universal Device ID (**7Fh**) will be used.
- The command to be used is the "**Bulk Dump – Edit Buffer Parameter**", the command byte will be **60h**.
- The "VCF LFO Waveform" parameter has address **06h** as shown in table 5.
- Start of SysEx Msg will be then:
  Start SysEx : **F0h**
  Mfr ID : **00h 20h 21h**
  Device ID : **7Fh**
  Model ID : **45h**
  Command : **60h**
  Address : **06h**
- Value for the "Saw - Rise" waveform is 8 (see table 3 in Owner's Manual). Since the datablock has only one byte, whole datablock will be then **08h**
- Now, the checksum must be calculated as 7-bit complement of the sum of bytes from 'Model ID' to 'Data': **00h – (45h + 60h + 06h + 08h) = 4Dh**
- The form of the whole required System Exclusive message is after the checksum and End SysEx byte refilling: **F0h 00h 20h 21h 7Fh 45h 60h 06h 08h 4Dh F7h**

Immediately when the message is received, the actual value of the parameter in the edit buffer will be changed.

4.4 EXAMPLE 4 – SAVE EDIT BUFFER

Task:

Save the data from edit buffer (e.g. set previously by CCs) to patch memory Nr. 64.

Solution:

- I don't know what number of the SysEx ID is actually active so the universal Device ID (**7Fh**) will be used.
- The command to be used is the "**System Functions**", the command byte will be **70h**.
- The function is "**Patch Save**" - the address byte will be **02h** as shown in table 6.
- Start of SysEx Msg will be then:
  Start SysEx : **F0h**
  Mfr ID : **00h 20h 21h**
  Device ID : **7Fh**
  Model ID : **45h**
  Command : **70h**
  Address : **02h**
- Databyte depends on number of required patch memory; for patch Nr. 64 it is **7Fh** (see chapter 3.7.3). Since the datablock has only one byte, whole datablock will be then **7Fh**
- Now, the checksum must be calculated as 7-bit complement of the sum of bytes from 'Model ID' to 'Data': **00h – (45h + 70h + 02h + 7Fh) = 4Ah**
- The form of the whole required System Exclusive message is after the checksum and End SysEx byte refilling: **F0h 00h 20h 21h 7Fh 45h 70h 02h 7Fh 4A F7h**

When the message is received, the actual values of all parameters in the edit buffer will be copied to permanent patch memory Nr. 64.

4.5 EXAMPLE 5 – FACTORY RESET

Task:

Make factory reset of the interface.

Solution:

- I don't know what number of the SysEx ID is actually active so the universal Device ID (**7Fh**) will be used.
- The command to be used is the "**System Functions**", the command byte will be **70h**
- The function is "**Interface Reset**", the address byte will be **07h**
- Start of SysEx Msg will be then:
  Start SysEx : **F0h**
  Mfr ID : **00h 20h 21h**
  Device ID : **7Fh**
  Model ID : **45h**
  Command : **70h**
  Address : **07h**
- Databyte for complete factory reset is **7Fh** (see chapter 3.7.8). Since the datablock has only one byte, whole datablock will be then **7Fh**
- Now, the checksum must be calculated as 7-bit complement of the sum of bytes from 'Model ID' to 'Data': **00h – (45h + 70h + 07h + 7Fh) = 45h**
- The form of the whole required System Exclusive message is after the checksum and End SysEx byte refilling: **F0h 00h 20h 21h 7Fh 45h 70h 07h 7Fh 45h F7h**

When the message is received, the interface is set to factory status.

4.6 EXAMPLE 6 – SEND COMPLETE MEMORY CONTENT

Task:

Send request for transmitting of complete content of the interface memory.

Solution:

- I don't know what number of the SysEx ID is actually active so the universal Device ID (**7Fh**) will be used.
- The command to be used is the "**System Functions**", the command byte will be **70h**.
- The function is "**Memory Status Request**" - the address byte will be **05h** as shown in table 6.
- Start of SysEx Msg will be then:
  Start SysEx : **F0h**
  Mfr ID : **00h 20h 21h**
  Device ID : **7Fh**
  Model ID : **45h**
  Command : **70h**
  Address : **05h**
- Databyte for request for complete memory content is **7Fh** (see chapter 3.7.6). Since the datablock has only one byte, whole datablock will be then **7Fh**
- Now, the checksum must be calculated as 7-bit complement of the sum of bytes from 'Model ID' to 'Data': **00h – (45h + 70h + 05h + 7Fh) = 47h**
- The form of the whole required System Exclusive message is after the checksum and End SysEx byte refilling: **F0h 00h 20h 21h 7Fh 45h 70h 05h 7Fh 47 F7h**

When the message is received, the complete content of the interface memory is send as stream of a SysEx messages (see table 7). These messages can be archived or processed by the "Patch Memory Organizer" support utility.

All documents and support software for the P6-M interface are available at manufacturer's web pages.

Korg Polysix MIDI Interface
Model P6-M, Nr. 8-427, ver. 1.00
Document: 842710_syxcom, rev. 2

Manufacturer: CHD Elektroservis, Czech Republic

www.chd-el.cz info@chd-el.cz