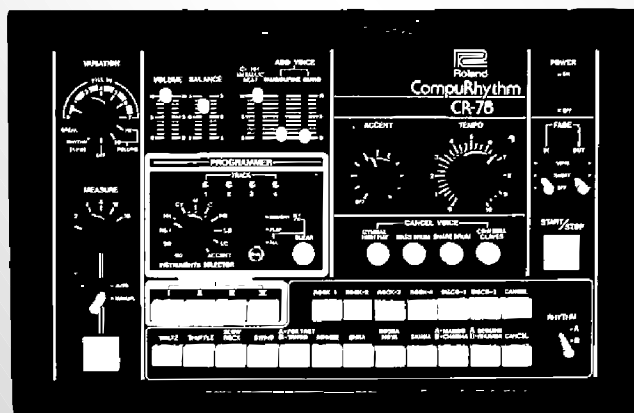# CRX8-M

MIDI Interface for Roland CR-68 / CR-78

Model 8-449
ver. 1.0

SYSTEM EXCLUSIVE COMMUNICATION

Content

1. SYSTEM EXCLUSIVE COMMUNICATION

CRX8-M interface disposes of a system of System Exclusive communication which enables to receive and to transmit a SysEx Messages for the changes of all parameters in memory banks of the interface. Also, it is possible to list an actual memory banks content. These data can be archived in PC for example. Some system and testing functions can be launched via SysEx commands too. Block diagram on pic. 1 shows system of SysEx communication.

Pic. 1 – Block diagram of SysEx communication
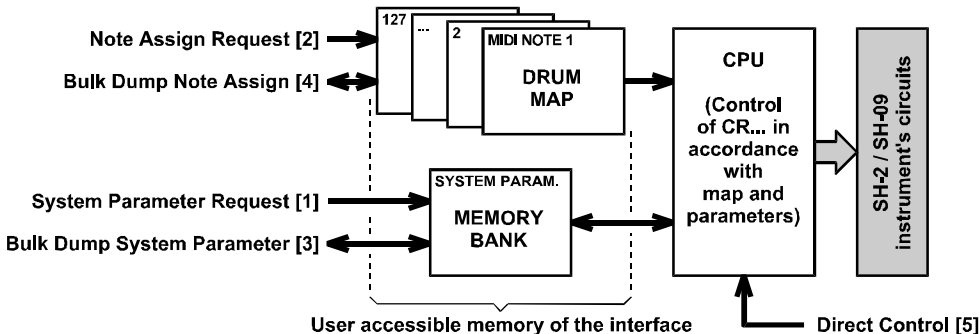
2. SYSEX MESSAGES STRUCTURE

CRX8-M receives and transmits own specific messages with this structure:

F0h	Start SysEx
00h 20h 21h	Manufacturer ID
ii	Device ID
65h	Model ID
cc	Command
aa	Address
dd…dd	Datablock
xx	Checksum
F7h	End SysEx

2.1. "DEVICE ID" BYTE

The device ID byte identifies the MIDI device exactly. It is equal to a number of just an active MIDI channel (00h for channel Nr. 1, 01h for channel Nr. 2 etc. up to 0Fh for channel Nr. 16). Next possible value of the Device ID is 7Fh – this is the universal Device ID always accepted by the interface regardless on which MIDI channel number is active.

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

2.2. "COMMAND" BYTE

The command "**cc**" specifies the interface function type after the whole SysEx Msg is received.

2.3. "ADDRESS" BYTE

The address "**aa**" specifies the memory area or selects the system function in dependence on a type of the SysEx Msg – see below.

2.4. "DATA" BYTE(S)

The block of databytes "**dd…dd**" contains the values of a parameters or functions given by the SysEx Msg. The number of bytes in datablock "**dd…dd**" is a different in dependence on a type of the SysEx message. There can be from 1 to 3 databytes. Some SysEx messages don't use the databytes at all.

2.5. "CHECKSUM" BYTE

The checksum byte "**xx**" confirms the validity of System Exclusive message. It must be calculated so that seven-bit sum of bytes from the "Model ID" to "Checksum" is equal to zero. If the checksum byte is invalid the whole message is invalid too and it is ignored by the interface.

3. COMMAND TYPES OVERVIEW

The command "**cc**" gives the interface activity after the whole message is received. The valid values of "**cc**" are:

"**cc**"=10h - **System Parameter Request** - inquiry for value of a system parameter stored in the interface's memory bank (see [1] on pic. 1).
"**cc**"=20h - **Note Assignment Request** - inquiry for values of a MIDI note definition (assigning to sound generators) stored in the interface's drum map (see [2] on pic. 1).
"**cc**"=30h - **System Parameter Load** - response to the "System Parameter Request" or the change of a content of memory bank in the interface's memory (see [3] on pic. 1).
"**cc**"=40h - **Note Assignment Load** - response to the "Note Assign Request" or the change of a content of drum map in the interface's memory (see [4] on pic. 1).
"**cc**"=50h - **Direct Control** – system functions for direct control of the interface's hardware, exact procedure is specified by the values of the "Address" and "Data" bytes (see [5] on pic. 1).
"**cc**"=60h - **Test** – testing functions for the interface's hardware checking.

3.1. "SYSTEM PARAMETER REQUEST" COMMAND

Form : **F0 00 20 21 ii 65 10 aa xx F7 [hex]**
(where "**ii**" = Device ID, "**aa**" = Address, "**xx**" = Checksum)

With a help of the "System Parameter Request" command, it is possible to ascertain values of system parameters stored in the system memory bank of user memory (see [1] on pic. 1).

Valid values of Address "**aa**" byte are from 00h to 07h. Address "**aa**" then defines requested parameter:

- 00h = MIDI Channel
- 01h = Velocity Priority
- 02h = Velocity Curve
- 03h = Noise Gate Duration
- 04h = Guiro Short Duration
- 05h = Guiro Long Duration
- 06h = Volume Control Mode
- 07h = MIDI Tempo Multiplier

For the address "**aa**" in range from 08h to 7Fh, the message is evaluated as invalid and it is ignored by the interface.

Databyte "**dd**" is not used in this type of the SysEx message at all.

Immediately after the "System Parameter Request" message receiving, the interface sends the "System Parameter Load" message as reply. Value of requested parameter stored in memory bank is included in that message.

3.2. "NOTE ASSIGNMENT REQUEST" COMMAND

Form : **F0 00 20 21 ii 65 20 aa xx F7 [hex]**
(where "**ii**" = Device ID, "**aa**" = Address, "**xx**" = Checksum)

With a help of the "System Parameter Request" command, it is possible to ascertain values from drum map in user memory which define assigning of a MIDI note to instrument's sound generators (see [2] on pic. 1).

Valid values of Address "**aa**" byte are from 00h to 7Fh. Address "**aa**" defines number of requested MIDI note.

Databyte "**dd**" is not used in this type of the SysEx message at all.

Immediately after the "Note Assignment Request" message receiving, the interface sends the "Note Assignment Load" message as reply. Values for requested MIDI note definition stored in memory bank are included in that message.

3.3. "SYSTEM PARAMETER LOAD" COMMAND

Form : **F0 00 20 21 ii 65 30 aa dd xx F7 [hex]**
(where "**ii**" = Device ID, "**aa**" = Address, "**dd**" = databyte, "**xx**" = checksum)

By transmitting of the "Bulk Dump System Parameter" message to the interface, it is allowed to change default values of system parameters stored in the system memory bank of user memory. Also, this SysEx Msg message is sent from the interface as reply to the "System Parameter Request" message (see [3] on pic. 1).

Valid values of Address "**aa**" byte are from 00h to 07h. Address "**aa**" then defines requested parameter:

- 00h = MIDI Channel
- 01h = Velocity Priority
- 02h = Velocity Curve
- 03h = Noise Gate Delay
- 04h = Guiro Short Duration
- 05h = Guiro Long Duration
- 06h = Volume Control Mode
- 07h = MIDI Tempo Multiplier

For the address "**aa**" in range from 08h to 7Fh, the message is evaluated as invalid and it is ignored by the interface.

The only one databyte "**dd**" contents value of requested parameter which will be stored in the interface's memory (or value read from memory bank if the message is transmitted from the interface as response to "System Parameter Request"). Value of the databyte "**dd**" must be inside valid range – valid range for individual parameters are:

- MIDI Channel
  00h to 0Fh : corresponds to numbers of MIDI channels 1 to 16
- Velocity Priority
  00h to 03h : 00h for "Average", 01h fof "Highest", 02h for "Lowest", 03h for "Last"
- Velocity Curve
  00h to 04h : 00h for "Lin", 01h for "Exp1", 02h for "Exp2", 03h for "Log1", 04h for "Log2"
- Noise Gate Delay
  00h to 3Fh : corresponds to period of noise gate opening from 100 to 1360 ms, step is 20 ms
- Guiro Short Duration
  00h to 3Fh : corresponds to sounding of Short Guiro instrument from 20 to 398 ms, step is 6 ms
- Guiro Long Duration
  00h to 3Fh : corresponds to sounding of Long Guiro instrument from 50 to 1310 ms, step is 20 ms

5

- Volume Control Mode
  00h to 03h : 00h for "None", 01h for "Volume", 02h for "Expression", 03h for "Both"
- MIDI Tempo Multiplier
  00h to 01h : 00h for "x1", 01h for "x2"

  Detailed description of the meaning of system parameters is in the user manual of the device.

3.4. "NOTE ASSIGNMENT LOAD" COMMAND

Form : **F0 00 20 21 ii 65 40 aa d1 d2 d3 xx F7 [hex]**
(where "**ii**" = Device ID, "**aa**" = Address, "**dx**" = databyte, "**xx**" = checksum)

By transmitting of the "Bulk Dump Note Assign" message to the interface, it is allowed to define assigning of any MIDI note to sound generators of the instrument. By other words, it is allowed to create own map of drum instrument in the user interface's memory. Also, this SysEx Msg message is sent from the interface as reply to the "Note Assign Request" message.

Valid values of Address "**aa**" byte are from 00h to 7Fh. Address "**aa**" defines number of requested MIDI note.

Datablock always includes three databytes "**d1 d2 d3**". Their meaning is as below:

- **d1**: Defines sound generator of the instrument which will be launched by given MIDI note. Allowed values of the databyte are from 00h to 14h – it corresponds to these sound generators:

  - 00h = None Generator (Silence) – no generator is assigned to MIDI note
  - 01h = Bass Drum (BD)
  - 02h = Snare Drum (SD)
  - 03h = Rim Shot (RS)
  - 04h = High Bonga (HB)
  - 05h = Low Bonga (LB)
  - 06h = Low Conga (LC)
  - 07h = Cow Bell (CB)
  - 08h = Maracas (MA)
  - 09h = High Hat (HH)
  - 0Ah = Cymbal (CY)
  - 0Bh = Claves (CL)
  - 0Ch = Tambourine Short (TS), for CR-78 instrument only
  - 0Dh = Tambourine Long (TL), for CR-78 instrument only
  - 0Eh = Guiro Short (GS), for CR-78 instrument only
  - 0Fh = Guiro Long (GL), for CR-78 instrument only
  - 10h = Metallic Beat (MB), for CR-78 instrument only
  - 11h = Snare Drum + Rim Shot (SD+RS) – simultaneous launching of SD and RS
  - 12h = High Hat + Metallic Beat (HH+MB) – simultaneous launching of HH and MB, CR-78 only
  - 13h = Cymbal + Metallic Beat (CY+MB) – simultaneous launching of CY and MB, CR-78 only
  - 14h = Cow Bell + Metallic Beat (CB+MB) – simultaneous launching of CB and MB, CR-78 only

  If value of first databyte "**d1**" is out of valid range (i.e. 15h to 7Fh), the whole message is evaluated as invalid and it is ignored by the interface.

- **d2**: Defines minimum of enabled dynamic range of assigned sound generator, i.e. if velocity of the MIDI note is equal to 1 (see pic. 2). Higher minimal level of acoustic signal corresponds to higher value of the databyte. Value of "**d2**" databyte can be from 00h to 7Fh.

- **d3**: Defines maximum of enabled dynamic range of assigned sound generator, i.e. if velocity of the MIDI note is equal to 127 (see pic. 2). Higher maximal level of acoustic signal corresponds to higher value of the databyte. Value of "**d3**" databyte can be from 00h to 7Fh. However, there is condition that value of "**d3**" databyte (maximal level of acoustic signal) must not be lower than value of "**d2**" databyte (minimal level of acoustic signal). If the interface receives SysEx Msg with

6

disallowed value of the third databyte "**d3**" (i.e. if "**d3**" < "**d2**"), the whole message is evaluated as invalid and it is ignored by the interface.

Each of SysEx Msg defines just one MIDI note. Definitions of all other notes stay without changes. If there is a request for rewriting of more notes definition, corresponding number of SysEx messages must be transmitted to the interface. But these messages **must not** be transmitted consecutive. **Time delay 50 ms at least** must be inserted before every particular message in order to store previous value into interface's memory safely. If a lot of SysEx messages are transmitted to the interface consecutive, input buffer overflows and an error can occur. The error then disallows correct operating of the interface (see appendix B. in user manual).

Pic. 2 – Definition of dynamic range of sound generator
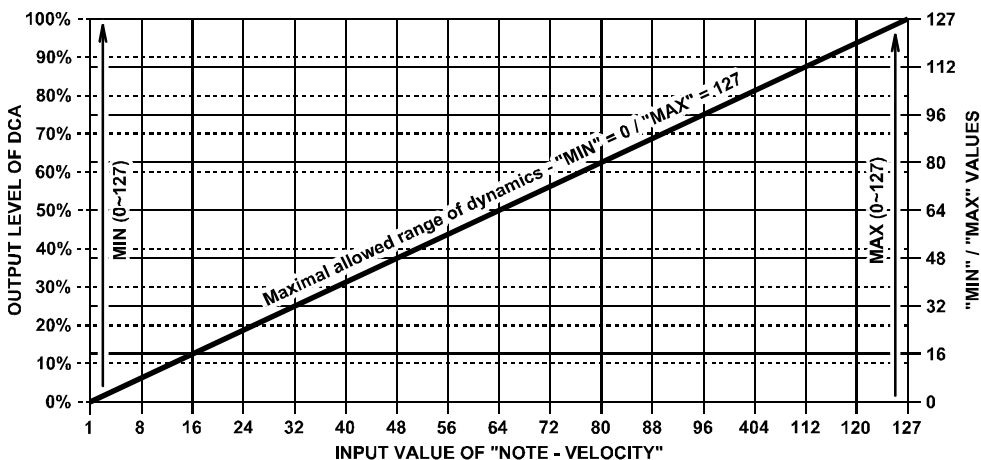
3.5. "SYSTEM FUNCTION" COMMANDS

Form : **F0 00 20 21 ii 65 50 aa dd xx F7 [hex]**
(where "**ii**" = Device ID, "**aa**" = Address, "**dd**" = databyte, "**xx**" = checksum)

The commands "System Function" ("**cc**"=50h) are direct commands for the execution of a system function which controls the interface's hardware directly. The type of function is given by the address "**aa**":

- 00h = Save Edit Buffer – saving of parameters values from edit buffer to memory bank.
- 01h = Set DCA – setting of output level of acoustic signal.
- 02h = LED-Indicator – direct control of indication LED on instrument's front panel.
- 03h = LED-OnBoard – direct control of indication LED on interface's board.
- 04h = SW Version Request – request for version of installed software.
- 05h = Memory Test – execution of functionality test of interface's user memory.
- 06h = Reset – hardware reset or factory reset of the interface.

The only one databyte "**dd**" specifies an activity of a system function exactly.

3.5.1. "SAVE EDIT BUFFER"

Form : **F0 00 20 21 ii 65 50 00 7F 4C F7 [hex]**

If the address "**aa**" is equal to 00h, the SysEx Msg is the command for the saving of temporary data from edit buffer to memory bank of the interface.

Value of databyte "**dd**" must be always 7Fh. For all other values, the message is evaluated as invalid and it is ignored by the interface.

3.5.2. "SET DCA"

Form : **F0 00 20 21 ii 65 50 01 dd xx F7 [hex]**

If the address "**aa**" is equal to 01h, the SysEx Msg is the command for direct setting of the interface's DCA.

Value of databyte "**dd**" can be from 00h to 7Fh. The value gives gain of the DCA – for value equal to 00h the gain is -∞ dB (zero level of acoustic signal), for value equal to 7Fh the gain is 0 dB (maximal level of acoustic signal).

3.5.3. "LED - INDICATOR"

Form : **F0 00 20 21 ii 65 50 02 dd xx F7 [hex]**

If the address "**aa**" is equal to 02h, the SysEx Msg is the command for direct setting of indication LED on instrument's panel.

Value of databyte "**dd**" can be only 00h (for LED off) or 01h (for LED on). For all other values, the message is evaluated as invalid and it is ignored by the interface.

3.5.4. "LED - ONBOARD"

Form : **F0 00 20 21 ii 65 50 03 dd xx F7 [hex]**

If the address "**aa**" is equal to 02h, the SysEx Msg is the command for direct setting of red LED on the interface's board.

Value of databyte "**dd**" can be only 00h (for LED off) or 01h (for LED on). For all other values, the message is evaluated as invalid and it is ignored by the interface.

3.5.5. "SW VERSION REQUEST"

Form : **F0 00 20 21 ii 65 50 04 7F 48 F7 [hex]**

If the address "**aa**" is equal to 04h, the SysEx Msg is inquiry for the number of version of installed operational system. The value of databyte "**dd**" can be only 7Fh. For all other values the message is evaluated as invalid and it is ignored by the interface.

Immediately after the inquiry receiving the interface transmits the SysEx message:

F0 00 20 21 ii 65 50 04 dd xx F7 [hex],

where particular nibbles of databyte "**dd**" give number of installed software version. For example, byte 10h denotes the version 1.0.

3.5.6. "MEMORY TEST"

Form : **F0 00 20 21 ii 65 50 05 7F 47 F7 [hex]**

If the address "**aa**" is equal to 05h, the SysEx Msg is the command for execution of complete test of internal interface's memory.

The value of databyte "**dd**" can be only 7Fh. For all other values the message is evaluated as invalid and it is ignored by the interface.

Test of memory is executed immediately after the message receiving. The test takes about 3 seconds. After end of the test, the interface transmits SysEx message with info about result of the test:

- **F0 00 20 21 ii 65 50 05 00 46 F7 [hex]** – if the memory works correctly
- **F0 00 20 21 ii 65 50 05 01 45 F7 [hex]** – if any malfunction was detected during the test

3.5.7. "RESET"

Form : **F0 00 20 21 ii 65 50 06 dd xx F7 [hex]**

If the address "**aa**" is equal to 06h, the SysEx Msg is the command for execution of reset of the interface. The value of databyte "**dd**" can be only 00h or 7Fh. For all other values the message is evaluated as invalid and it is ignored by the interface.

If the value of databyte "**dd**" is 00h, the hardware reset of the interface is executed. The interface is set to the same status as after the instrument is switched on.

"Factory Reset" drum instrument map											
Note Nr.		Assigned sound generator				Note Nr.		Assigned sound generator			
dec	hex	Nr.	Instrument name	Dynamics		dec	hex	Nr.	Instrument name	Dynamics	
				Min.	Max.					Min.	Max.
32	20	11	Claves	0	127	55	37	18	Cymbal + Metallic B.	0	127
33	21	1	Bass Drum	0	127	56	38	7	Cow Bell	0	127
34	22	3	Rim Shot	0	127	57	39	20	Cow Bell + Metallic B.	0	127
35	23	1	Bass Drum	0	127	59	3B	18	Cymbal + Metallic B.	0	127
36	24	1	Bass Drum	0	127	60	3C	4	High Bonga	0	127
37	25	3	Rim Shot	0	127	61	3D	5	Low Bonga	0	127
38	26	2	Snare Drum	0	127	62	3E	6	Low Conga	0	127
40	28	17	Snare Drum + Rim S.	0	127	63	3F	5	Low Bonga	0	127
42	2A	9	High Hat	0	127	64	40	5	Low Bonga	0	127
44	2C	9	High Hat	0	127	69	45	8	Maracas	0	127
46	2E	18	High Hat + Metallic B.	0	127	70	46	8	Maracas	0	127
49	31	10	Cymbal	0	127	73	49	14	Guiro Short	0	127
51	33	10	Cymbal	0	127	74	4A	15	Guiro Long	0	127
52	34	10	Cymbal	0	127	75	4B	11	Claves	0	127
53	35	12	Tambourine Short	0	127	80	50	16	Metallic Beat	0	127
54	36	13	Tambourine Long	0	127	82	52	8	Maracas	0	127
Remark: "None Generator" (instrument Nr. 0) is set for al other MIDI notes.											

"Factory Reset" values of system parameters			
Nr.	Parameter name	Setting to	Parameter value [hex]
00h	MIDI Channel	MIDI channel Nr. 10	09h
01h	Velocity Priority	Average	00h
02h	Velocity Curve	Linear	00h
03h	Noise Gate Delay	Delay 400 ms	0Fh
04h	Guiro Short Duration	Duration 92 ms	0Ch
05h	Guiro Long Duration	Duration 410 ms	12h
06h	Volume Control Mode	CC7 – Volume	01h
07h	MIDI Tempo Multiplier	x1	00h

If the value of databyte "**dd**" is 7Fh, the complete "Factory Reset" is executed. The interface is initialized to the factory status (all parameters are set to the default values). **WARNING - All data stored by user in the internal memory are fully erased during execution of the command.** It is recommended to make a back up of all user data before the "Factory Reset" execution.

After successful execution of "Factory Reset", red LED in interface's board blinks 5 times with period about 0,4 sec.

3.6. "TEST" COMMANDS

Form : **F0 00 20 21 ii 65 60 aa dd xx F7 [hex]**
(where "**ii**" = Device ID, "**aa**" = Address, "**dd**" = databyte, "**xx**" = Checksum)

The "Test" commands ("**cc**"=60h) are assigned only for the debugging in production and for the service procedures of the interface. Their usage is not assumed in normal usage.

4. EXAMPLE OF SYSEX MESSAGE CREATION

Task1:

Set MIDI channel to channel r. 16.

Solution:

- We don't know what number of the MIDI channel is just an active so the universal Device ID (7Fh) will be used.
- The necessary command is the "**Bulk Dump System Parameter** ", the command byte will be **30h**
- The value will be stored to the first position of memory bank so address byte is **00h**
- MIDI channel Nr. 16 corresponds to parameter value 15 so databyte is **0Fh**

Start of SysEx Msg will be then :

Start SysEx :	**F0h**
Mfr ID :	**00h 20h 21h**
Device ID :	**7Fh**
Model ID :	**65h**
Command :	**30h**
Address :	**00h**
Data :	**0Fh**

- Now, the checksum must be calculated as 7-bit complement of the sum of bytes from 'Model ID' to 'Data', i.e.:

  00h – (65h + 30h + 00h + 0Fh) = 5Ch

- The form of the whole required System Exclusive message is after the checksum refilling:

  F0h 00h 20h 21h 7Fh 65h 30h 00h 0Fh 5Ch F7h

Immediately after that message is received, default MIDI channel will be changed in both memory bank and edit buffer and the interface will start the working with newly stored data.

Similar method can be used for the creation of the SysEx Msg for drum map definition.

Task2:

Assign sound generator Cymbal with dynamic range from 32 to 96 to MIDI note Nr. 0.

Solution:

- We don't know what number of the MIDI channel is just an active so the universal Device ID (7Fh) will be used.
- The necessary command is the "**Bulk Dump Note Assign**", Command byte will be **40h**
- Values for MIDI note Nr. 0 are stored so Address byte will be **00h**
- Requested sound generator is Cymbal so databyte **d1** will be **0Ah**
- Requested minimum of dynamic range is 32 so databyte **d2** will be **20h**
- Requested maximum of dynamic range is 96 so databyte **d3** will be **60h**

Start of SysEx Msg will be then :

Start SysEx :	**F0h**
Mfr ID :	**00h 20h 21h**
Device ID :	**7Fh**
Model ID :	**65h**
Command :	**40h**
Address :	**00h**
Data :	**0Ah 20h 60h**

- Now, the checksum must be calculated as 7-bit complement of the sum of bytes from 'Model ID' to 'Data', i.e.:

  00h – (65h + 40h + 00h + 0Ah + 20h + 60h) = 51h

- The form of the whole required System Exclusive message is after the checksum refilling:

  F0h 00h 20h 21h 7Fh 65h 40h 00h 0Ah 20h 60h 51h F7h

    Immediately after that message is received, definition of MIDI note Nr. 0 is changed in drum map memory and since this time, the interface will use Cymbal drum instrument with dynamic range 32 to 96 for MIDI note Nr. 0.

- - - - -

    For easier calculation of the checksum, a special software Checksum Calculator can be used. The calculator is based on Java script. The calculator is available on the supplemental CD-ROM.

- - - - -
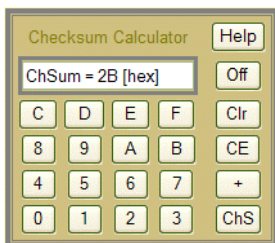
    Special software generator for the simple creation of the SysEx messages necessary for control and programming of CRX8-M interface is available on the supplemental CD-ROM. The generator is based on Java scripts. Any message for the interface setting described above can be created with a help of this generator.