

# MSQ-CC-BCR

MotionSeQuencer for ControlChanges for BCR2000 by wiring it to MBHP Synth-Patch-Editor & Motion-Sequencer 4 ControlChange (= CC-Automation)





## Introduction

i have the need to control and automate my Nord Drum2 [NORD DRUM 2](#)

So i connect a BlackBox between Sequencer & Synthesizers...

This Box is called MSQ\_CC\_BCR: **M**otion **S**equencer for **M**idi**C**ontrol**C**hange controlled via a **BCR2000** Midicontroller

### It acts as:

- **Midi Merger** NTE,CLK,PC merge with CC... & CCinput is a thing between MSQ\_CC\_BCR and BCR only since we have intelligent UI with Pages..

- **Patch Manager** it replaces the Synths internal Patch Storage, to even get more, because, each PC event from your Sequencer is multiplied by 4... in each Program you have 4 Variants, and you can morph between them, to get smooth transition between them

- **Motion Sequencer** Record your Controller Movements in a Sequence in 32th Resultion @ maximal 128 Steps length

64th is possible, just a Factor in the code, but it will also reduce the max.Step length to 64 steps, also the midi traffic will go HI! imagine you automate  $8 \times 32 = 256$ CCs, and dump that @64th into your synth, over midi... ;)... but in cases like: drumcomputers, where a view things must be static to provide the percussive punching sound - there are normally not that much automations... so maybe for a drummachine 64th is a good voice... how ever i get lost in detail...

## Features

- **Remote your Synths** by: 8x Midichannels with up to 32x Control Change (CC)

For the BCR i only can provide  $8 \times 29$ , because i need some controlls to control the MB Program itself...

- **Save the Patches** - and dump it the Synth

- **Save 4xSnapshots per PROGRAM (=Song) && Morph between them** (exclusive MotionSequence!)

Morph between Snapshots: when Morph is activated, you are somewhere in between 2 Patches... so you have to save the patch to one of the other (2others, when morph between 2 = 4) in order to store the current "cool state" »> whole thing has to implemented... searching for a faster code

-  **Fix Me!** want help for a FAST Code: Morph between ACTUAL & NEXT PATCH (via Morph Rotary)

- **Load Patches via received Program Change**

- **Record CC-Motion-Sequences** use a footpedal connected to FSW1 on the backside of the BCR, to ARM/Disarm it... so you can tweedle 2 ore more CC @ once... but you dont have to, BCR-onboard is also a Button for it
- **PLAY Motions-Sequences**, up to 256 steps @ 32th rate, which is 128Steps @ normal 16th, this rate can changed in code only, and 256 is maximum... 512 make a RAM-Issue (8x32x512 + the rest of all variables...)!...\\so if you want 64th CC-Smoothness your Motionsequencer has only 64steps length then, this steps of course are only right when you have choosen 4/4 in your Patch... elsewhere of course will differ a bit...but 256 is max.
- **VELOCITY MORPH** Add Velocity-Ammount to CCs
- **MERGE incoming Midi-Notes/Clock/Pitchbend with Automated CCs.**
- **Set Sequencer Beatstructure** » how to interpret Clock-ticks (4/4, 5/4, 6/4, 7/4...)



- Need help for a FAST Code: **scale min max values for CCs** (synths have CCs which value are between 0-3 instead of 0-127)
- A **Global Page**: for example you use 8 similar Drum-Voices, with the Global you have 8 channel strips with dedicated Controls, for example: 8xVolume, 8xTone/Noise-Mix, 8xDistortion, 8xClick if you have one Synth over 2 MSQ\_CC\_BCR Tracks(booth set to MidiChannel 0, to get 64CCs instead of 32), then the Global Page: have the ability to show/edit a parameter from Track1Voice on Track1Global, and from Track2Voice on Track2Global... it depends how you set the Midichannel in the Systemsettings (which are currently in the CODE itself (recompile
- Many of this features, especially the **System Settings would need a UI**, but that would it make bigger, more expensiv, and maybe more complex to use... in **this case is UI-less** - it is set once, for one multipart-synth+bcr2000, MSQ\_CC\_BCR do all the Preset Store, and Automations, so it is one Unit > to use the Unit in a other way would make all the Patches (1000Patches+3000 Variants) useless, so once done, it is a black box loadet via Programchange!

## Hardware Requirements

### External Requirement:(for example)

- Melody/Clock Source with ProgramChange-Output: [midibox\\_seq\\_v4l](#) oops that dont do PC...
- Melody/Clock Destination: Waldorf Pulse, NordDrum 2, anything which can handle CC
- Midicontroller: 1x BCR2000, or any other with LEDRing & Encoder.

### Midibox:

- [core32](#)
- [1xMidi IO](#) connect 1 midicontroller and 1 Note/Clock-Source/Destination
- SD-Card, formated with FAT32, and the file "bcr1.syx" on it
- Soldering Iron, Wires, PCB....
- USB Power Supply... I tried to use the Midi-BUS-Power from BCR2000 but it is too weak!

### Visual Feedback directly from MBHP

- a Momentary Switch Connected to J5A Pin0
- 7 LEDs in serial with 220Ohm each to GND connected to J5A Pin1-3 and J5B Pin0-4

The LEDs show via Gestic (Patterns) if something is wrong, done, busy, & show the Rythm structure: The Switch switches as Radio-Button thru the Rythm Structures (4/4, 5/4...), the LED-Indicating this.

By Holding the Switch and Powering the Core, it will Dump Out a Sysex Template to your BCR.

Be Careful dont short the Input Pins!



### Why BCR2000

because I have 3 of them but they are to old dirty, damaged... i cant get a good price for it, so better hold it and make something with it.

### Setting up a BCR2000

#### Cabeling

MidilIO PortA Out >> BCR Midi IN  
AFTER Uploading the Sysex, and restarting the BCR connect:  
MidilIO PortA In >> BCR Midi OUT A

#### Upload the Sysex-Template

1. unpack [bcr1.syx.zip](#) and put "bcr1.syx" on a SD-Card (root level)
  2. Put SD-Card into CORE32
  3. bridge J5A Pin0 to ground, or connect a switch to it, that you will need if you want to sequence other song structures then 4/4 (which is default)!
  4. Power the core up.
- ...if the filestructure (patches) are already existent...then it takes less then half a minute to dump the BCR-2000 Layout Data...
- You dont have to save the preset, it will make it automatic

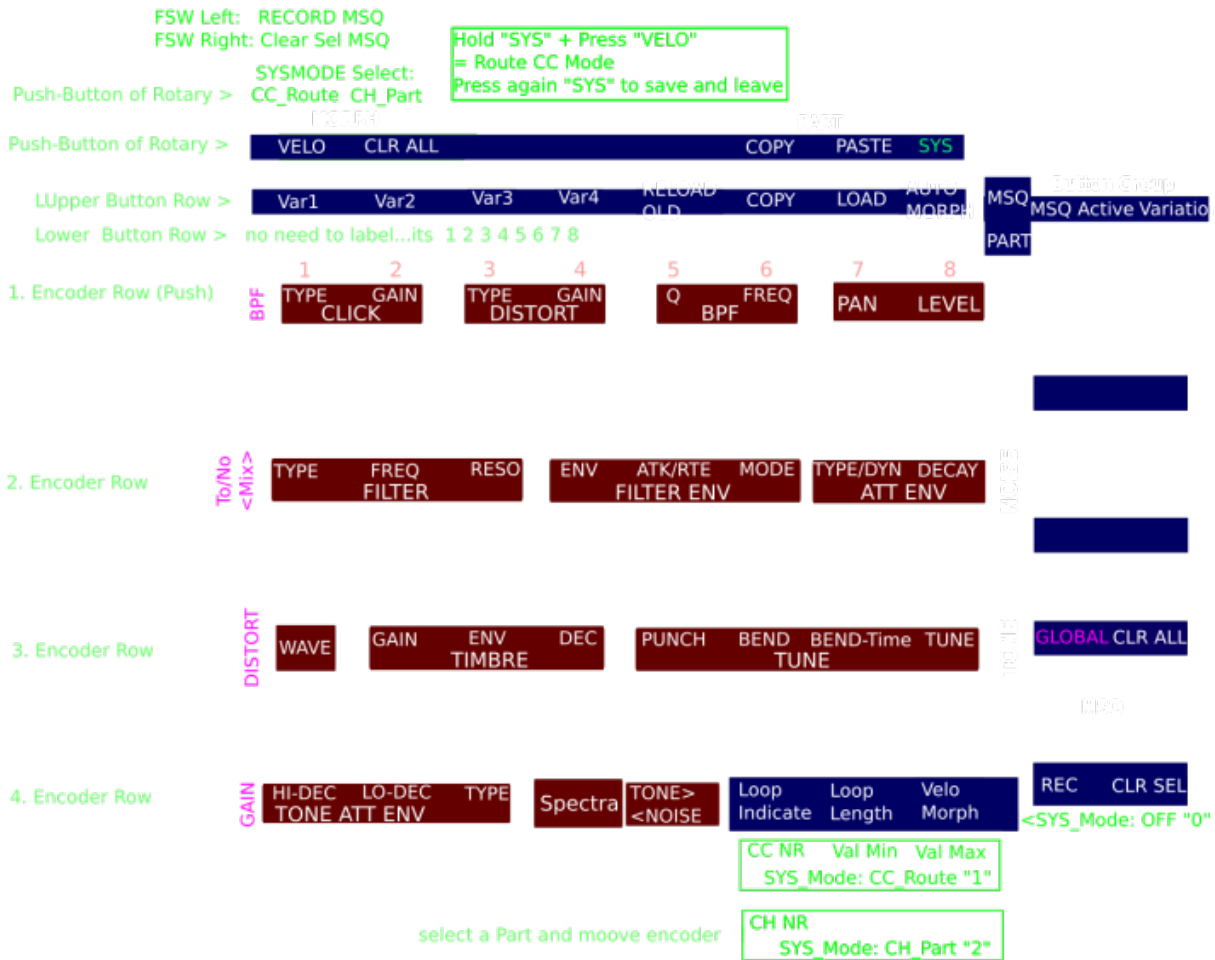
...when no filestructure... then it will take about 16minutes... the core has to make 4000Patches\*32KB=139MB!!!!... so better:

\* Faking a filestructure: make a empty folder "mq" and put it on SD-Card, make the syx.dump, make your first simple standart patch, the sound you will start the next 1000 Patches with ;) so choose carefully... then remove the Card, earse the "mq" folder on the card, and put it into the core again, it now will copy your "standart patch" to 4000 others

## Frontpanels

### BCR2000 Stickers

The Blue Elements are the MBHP Remotes... the Rest is for the Synth



UNTESTET, NOT SCALED!!!!

## MBHP

# Software

# CC Routing to Synths

MSQ\_CC\_BCR internal i have 8x32 CCs, they are always identical. but with a simple input output matrix i can decide which CC it gets in real world. each of the 8 Part can have midichannle 0-15... So we talking about Mapping... in the moment it is made in the source code with a simple array. this array could be saved and loadet from SD-Card aka "SYS settings", and this array could be editet by a



simple editor... i dont have a glue about this... but the format of this setting is very simple, the file starts with (converted from hex) mq04 and then the Routing array starts [32][127] for those how know how to program a simple interface for it?

## To Do

getting Access to the 7segment Display on the BCR2000 needet for to see the actual Preset (0-999), the Machine is built to PC via Midi, but what when you want to copy a whole patch! to a other

scale min max values for CCs (for example different synths have only 0-3value instead of 0-127, by different functions like WAVEFORM...) - should also a part of the editor

## Resources

- [BCR-Manual](#)
- [BCR-SYSEX-GUIDE](#)
- [TOKEN-Reference](#)
- [BC-Convert](#) Convert SYX into Textfile to Edit and reverse...

# Community users working on it

- [Phatline](#) = Programming, Documentation...

Just let a Private message on the forum to user already involved

From: <https://www.midibox.org/dokuwiki/> - **MIDIbox**

Permanent link: <https://www.midibox.org/dokuwiki/doku.php?id=msq-cc-bcr&rev=1476809002>

Last update: **2016/10/18 16:43**

