TOTAL REMOTE MIXING

David Kent

(Part1)

I am sure there are a number of engineers like myself, with minimal knowledge of Midi, that have already solved all of the numerous problems that present themselves when trying to remote control any digital desk. Still, I could not work it out so I bought a Yamaha 01v96 and a Behringer BCF2000 and a Belkin USB Network Hub so that I might verify any possible outcomes before dying on a live show!

Thanks to Apple (Macintosh) for driving me to finish my project, after reading their patent application for remote mixing. They really want to own the World! – and my world too!!

My requirements –

Control of the desk from my PC.

Playback of audio from my PC to the desk.

Headphone monitoring on my PC of the desk solo output.

Stereo recording of the desk control room output on my PC.

Musicians direct control their own in ear monitor mixes.

Real linear fader control of the main mix from my PC.

All stable, and as failsafe as possible.

All of the above via one dedicated Ethernet cable.

I started my exploration of Midi before purchasing the mixer. I found that all of the Studio Manager emulators (apart from the PM5) will send system exclusive (sysex) Midi messages when their controls are moved, even when not connected to a mixer. These messages are desk specific, as the first part of the message says which piece of Yamaha equipment it is for. This could be useful if it is necessary to control more than one desk remotely.

..................................................................................................................................................................

***YOU DO NOT NEED TO UNDERSTAND THE NEXT PART IN DETAIL, BUT IT WILL EXPLAIN ANY PROBLEMS THAT YOU MIGHT HAVE IF YOU TRY TO USE THE “LEARN” MODE ON THE BCF***

Their format is: $F0 $43 $10 $3E $7F $01 $1C $00 $00 $00 $00 val7.13 val $F7

The “val” bits you can investigate, along with some other great leaps of knowledge, by reading (and understanding??) “Behringer’s Secrets”, a document that results from some pioneering work and that can still be found at: <http://shoshin.110mb.com/genmce/bcr2000/files/SecretBC.pdf>

The $ denotes that what follows is a Hex value. The “val” parts are the variables that change as, for example, a fader is moved. If you look at the sysex messages in MidiOx you will see 14 Hex values. You will not see “val” nor “val7.13”. These two items will appear as $(Hex that changes as you move the control)

“val” is bits 0 to 6, the Most Significant Bits (MSB) of the 14-bit number that represents the control’s position.

“val7.13” is bits 7 to 13, the Least Significant Bits (LSB) of the same number.

The whole message, which is repeatedly sent as a control is moved, reads like this:

$F0 (new sysex message)

$43 (Manufacturer’s ID number - Yamaha)

$10 (device no. and Midi channel (0=1 for midi))

$3E (MODEL ID – Digital Mixer)

$7F (specific Model – apparently the 01v96 should be 0D, but it accepts 7F – ERR????)

$01 $1C $00 $00 (address)

$00 $00 (input ch1 fader)

val7.13 (LSB)

val (MSB)

$F7 (end of message)

Time to set about using this knowledge to set up a BCF2000, and to use the faders to control Studio Manager and the desk. I found that I was only really interested in the faders on the BCF in order to control the faders and the aux sends, and maybe the buttons for On/Off and Solo. Studio Manager was fine for controlling everything else.

I also found later that I was not really going to want the controls on the BCF to continuously respond to incoming Midi. At first this was frustrating, and as I was still tearing out my hair trying to understand NRPN (non-registered parameter numbers) Midi messages, I started to despair. So now I could make the Desk and SM (Studio Manager) respond to my BCF, but the BCF would not follow the other controls as it is set to send Sysex, but cannot recognise those messages.

HOW? Using ..............................

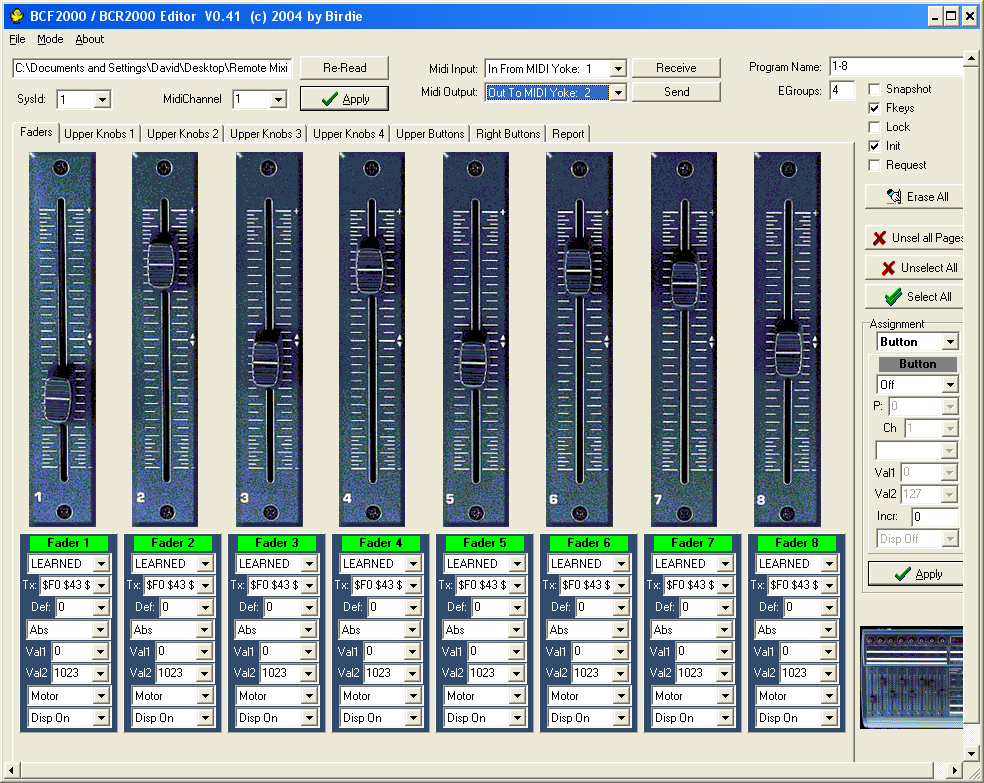


Birdie link: <http://rapidshare.com/files/115649724/bcfredit_V0.41.zip.html>

Like this....

.........

...........................



Into the Tx box I typed (using a spreadsheet and cut and paste is easier) the following for the first four input faders of my new 01v96.

$F0 $43 $10 $3E $7F $01 $1C $00 $00 $00 $00 val7.13 val $F7

$F0 $43 $10 $3E $7F $01 $1C $00 $01 $00 $00 val7.13 val $F7

$F0 $43 $10 $3E $7F $01 $1C $00 $02 $00 $00 val7.13 val $F7

$F0 $43 $10 $3E $7F $01 $1C $00 $03 $00 $00 val7.13 val $F7

Etc.

Note that for channel 11 and 12 it will be:

$F0 $43 $10 $3E $7F $01 $1C $00 $0A $00 $00 val7.13 val $F7

$F0 $43 $10 $3E $7F $01 $1C $00 $0B $00 $00 val7.13 val $F7

You will need to love Hex!

You will note that I decided to send with a resolution of 1023, which I thought would usually be enough. Later I changed the “Abs” to “Abs 14” and typed 16383 into “val2” to use the same 14 bit resolution as SM.

So now I am the proud owner of an 01v96 and I can’t wait to connect it all up. To get where I want to go I need to buy more BCF’s, another Belkin Network Usb Hub (NUH) and another sound card, but for the time being all is well.

..................................................................................................................................................................

***BUT THIS BIT YOU WILL NEED!***

Thanks to Andy for “Sends on Faders” (SOF) and “Change Recorder” (CR). I am not really using SOF as I have given control of the monitors (in-ear) over to the band. This allows me to do other things during rehearsals. I have incorporated it into my setup however as it is a great piece of software that others will certainly find useful.

Change Recorder is crucial though. I found that the desk would spit out NRPN Midi messages on the Midi Out 5-pin din when it received Sysex messages. These two types of message are interleaved in the Midi message flow. They don’t interfere with each other, as all of the equipment understands the differences, and I have found that useful.

(Part2)

If I feed the NRPN back to the BCF then its faders will not move as they are sending sysex ((see the bit you jumped!)) and don’t recognise NRPN. Great? But this feature is not necessary, as my chosen faders are only being moved by the BCF (FOH by me, monitors by the Band), and as I change pages on the BCF (ch1-ch8 >>> ch9-ch16>>>> and back again) the faders will move to their new positions each time anyway as they are held in the BCF’s temporary buffer.

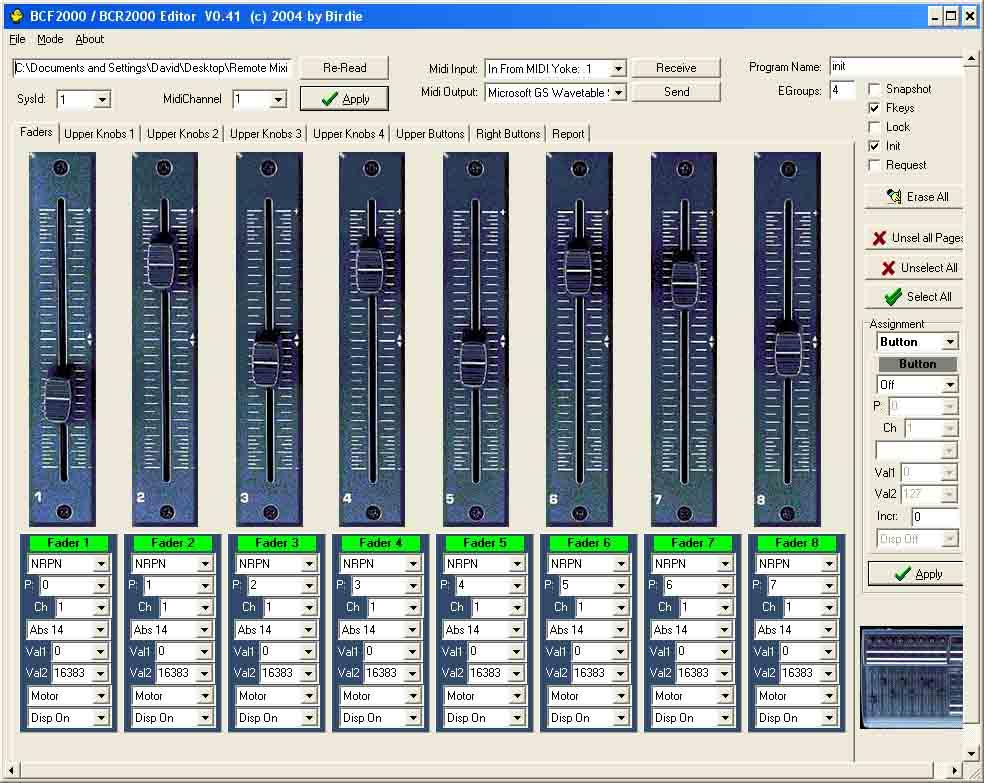
The problem rears up if there is a power failure. I am looking of course for a stable system. The BCF’s will then not remember their last fader position unless I have the presence of mind (and the time to waste) saving every move that I make. On power up any fader movement will write over the last value of the desk, and as the faders are at minus infinity when I change pages (on the BCF’s) the mix will be destroyed by immediate mutes.

Change Recorder to the rescue. It only talks NRPN and remembers always the last values that it has received for all parameters. As the desk is spitting out the NRPN equivalent of the Sysex messages that it receives from SM and the BCF’s, CR will hold the latest values for all parameters at the moment that the power is turned off by that idiot who won’t own up! Stop CR Record (I am using a laptop, with a good battery!), and once the power is back up play back the NRPN. The BCF’s will receive the NRPN and the motorised faders will set to the value received. NO THEY WILL NOT – because they are sending sysex they will not recognise the NRPN values.

Back to the drawing board.

Now is the time to nearly forget about Sysex, although I hope that what you have seen above will help your understanding of the whole caboodle.

NRPN turns out to be easy in Birdie:



This is a screenshot of the setup for the first (1-8) faders. I believe that it will work for any Yamaha digital desk, but that I have not verified. All of the settings can be programmed directly with very little typing required (*READ BIRDIES INSTRUCTIONS ON HOW TO USE THE PROGRAM*!).

Now my BCF’s will receive NRPN and the motorised faders will move. However, when they move they will send their new position (required, that is their purpose after all). This NRPN will be sent back from the desk and will be received by the BCF. This sets up an indirect feedback loop and is disastrous.

So – I do not want to connect the BCF’s to the midi out of the desk. I want to record the midi out from the desk in CR, and then play back CR after a power cut, repeating the playback for each preset on each BCF. Now I want to be able to automate this. How? With sysex program change messages I expect. I am going away to look at that right now.

Today, 10 March 2010, that is as far as I have advanced. Andy is I hope going to modify CR (see my last couple of posts), and he will give me a recoverable set-up on a silver platter!

This has become a bit of a blog (my first ever!). I will tidy it up into a tutorial maybe, but I believe that understanding is a vital accessory to information, so I might well leave it as is.

David Kent.

............................................................24 March 2010..........................................................

Thank you Andy, that is brilliant. CR now recognises all NRPN and does not care about message order. I was having trouble because the BCF’s chuck out NRPN as:

MSB > LSB > MSB data > LSB data and the desk sends:

LSB > MSB > MSB data > LSB data

CR now records all of the NRPN and when recording is stopped and the data is played back the BCF’s update just fine! No they do not!!! They only update their current preset. Grr!

The BCF can be sent SYSEX preset info. It can be asked to send it back ( you do not have to press EDIT + PRESET> on the BCF if you send the correct SYSEX ). And you can tell it to change to another preset.

Here is the sysex that I have so far:

*GO TO PRESET 20* $F0 $rev F1 $recall 20 $end

Translates as: F0 00 20 32 00 14 22 13 F7 (paste this into the SyxEx command window and send it from MidiOx)

So the “22” means go to (load) preset and the “13” (HEX) means 19 (decimal) which is preset 20 in midi-speak. Ugh!

*SEND ME BACK PRESET 18 (single preset dump)* F0 00 20 32 00 14 40 11 F7

The “40” means “send me the sysex file for” and the “11” means 17 (decimal) which is 18 in midi-speak. This command also means “and do not require local confirmation”. It also DOES NOT change the live preset on the BCF..... Handy!

Want more?

For a BCF unit set to ID2

*GO TO PRESET 1 unit ID2* F0 00 20 32 01 14 22 00 F7

Obviously the “01” means “this message is for the unit set to ID2”. You will be able to draw your own conclusions about how the rest will pan out. Do not forget that all of the values are Hex.

*SEND ME BACK EVERYTHING (id1 DUMP)* F0 00 20 32 00 14 40 7E F7

I am interested to know which is the current preset, as I want to return to it when I have finished restoring the data in the BCF temporary buffers. Otherwise I think that the musicians, and especially the drummer, could become confused. I need a message “send me back *current* preset”. I hope that it exists.

And I would like to script the whole process, as time might be of the essence and errors due to panic are to be avoided.

And I will try to do all of the IP talk over a mains Ethernet link.

Back later.

(Part3)

I have ordered some powerline ethernet CPL adapters.

It took me a while to remember my setup for testing for BCF Sysex commands. It was pretty simple. Just put the BCF in Mode 4 and connect output “A” to another midi device and then you can record just the sysex that the BCF is receiving.

So the command for

*SEND ME THE CURRENT PRESET (id1) and don’t even think about it* is:

F0 00 20 32 00 14 40 7F F7

The trouble is that the information returned does not say what the preset number is.

I start to wish that I never had!

But it does return a “name”. This name’s default is ‘init’, but it can be changed in all of the BCF editor programs. It will not show up on the BCF’s display. Behringer say that the firmware will not support it (true?), but when you save your Preset setup to the BCF the name goes with it, and when you retrieve that file from the BCF it comes back. So now I can identify the current preset when the power comes back. Yes, the BCF starts up where it left off, just like your car, it just forgets who was along for the ride.

The "Name" will be bytes 34,35,36,37, ..... of the returned file, and of course as this is Hex and the first byte is numbered "0" we are looking at the 53rd to the 56th.......... bytes in the list.

And then, while I was looking for something else, I found

*SEND ME BACK THE CURRENT PRESET NAME*

F0 00 20 32 00 14 42 7F F7

And it does just that! It returned: F0 00 20 32 00 14 21 00 7F 41 6C 65 78 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 F7

In this case the preset name was “Alex” – 41 6c 65 78 in hex, and 24 “name” bytes have been returned . I don’t know as yet what all of the part F0.....7F means. F7 means “end of message” as usual, and some of the rest I can guess at. The 21 is 33 in decimal. Maybe that means that the current preset is held temporarily as preset 33 while it is being used?

So now a bit of sysex “request” message analysis:

$F0 System exclusive message header

$00 $20 $32 Behringer

$00 Device ID = $00.....$0F (=ID on actual device) or $7F(=any)

$14 Model = $14(=BCF2000) or $15(=BCR2000) or $7F(=any) or $xx?(=BCD2000)

$42 Command = well, we now know some of them!

$7F Preset $7F(= current) $7E (= all or any) $00....$1F (preset 1...32)

$F7 End Of Exclusive message

I want to be able to tabulate the data that is returned and use it for recovery automation, so I am now off to learn some VBS script, and to mow the lawn (BLOG).

A+

Hopefully I am pretty much finished with Sysex, NRPN, and Midi at this point.

If I keep it simple and just name my preset Sysex files 1.....32 then I will know what it means in my real world. Although this could become complicated with the Hex, I am trying to automate a process, so I think that I will stick with real world for the bits that I will have to deal with manually, like making new preset files.

I would like two files, one that I can run at Boot – before a soundcheck, and one that I can run if the power goes down.

***BOOT:***

Start

Ask for confirmation that all components are plugged

Open Belkin Control Center

Close (minimise) Belkin Control Center

Start MidiOx

Load MidiOx Profile

Minimise MidiOx

Open SOF

Open Studio Manager

Open Console Manager

Establish Console Manager Midi links

Synchronise Console >> Pc in Console Manager

Open Change Recorder

Load Last Change recorder XML file

Confirm, and show and ask for confirmation of/offer to change Last BCF current Preset sync XML file data

Open BCF Current Preset sync file

Ask BCF (idx) Current Preset and parse data

Insert Data to sync file

Repeat (governed by sync file - file must be set up with data previously listing the presets used)

Set all BCF to Preset 1

Play Change Recorder

Repeat (governed by sync file)

Set all BCF to last current preset parsing data from Preset Sync file

Start Change Recorder Record

Confirm

End

***IF POWER FAILS:***

Start

Stop Change Recorder Record

Save Change Recorder Recorded XML file

Close Console Manager

Close Studio Manager

Close SOF

Close MidiOx

Exit Belkin Control Center

Ask for confirmation that Power is restored

Open Belkin Control Center

Close (minimise) Belkin Control Center

Start MidiOx

Load MidiOx Profile

Minimise MidiOx

Open SOF

Open Studio Manager

Open Console Manager

Establish Console Manager Midi links

Synchronise Console >> Pc in Console Manager

Open BCF Current Preset sync file

Ask BCF (idx) Current Preset

Insert Data to sync file

Repeat (governed by sync file - file must be set up with data previously listing the presets used)

Set all BCF to Preset 1

Play Change Recorder

Repeat (governed by sync file)

Set all BCF to last current preset parsing data from Preset Sync file

Start Change Recorder Record

Confirm

End

The two scripts will be very similar, but as I have done almost no programming since Fortran was the leading language (Cobal was just starting for this sort of application) this bit is going to take a while. Don’t hold your breath. Instead, you could write some of it for me if you like?

If you are holding your breath.... STOP.

Here is something else.

You will remember, if you have read the start of this Blog, that I wanted to be able to monitor the 01v96v2 (or other desk) SOLO at my computer.

First I must explain my setup. My 01V96 is on stage and connected (along with my USB soundcard and BCF's for the band) to a Belkin 5-port USB Hub which is connected via ethernet (wireless works too, but at short range) to my laptop. I have another BCF connected to my laptop for mixing and other control (such as fine aux tuning).

Trying to monitor at my laptop I had no luck with "Jack Control". XP does not allow connection of one soundcard input to another soundcard output. But the problem is solved with "Audacity" –

<http://audacity.sourceforge.net/download/>

Preferences set to "software playthrough" ticked, and then the program set to input from my soundcard and output to my laptop soundcard and then put in paused record and minimalised.

I had tried the separate program called "*software playthrough*" but it Hung repeatedly.

This works with a slight delay when monitoring, which is actually helpful when trying to identify hums and buzzes and distortion.

I can play audio from my computer to my 01V96 at the same time. Playback is with micro-mini-tiny latency. The soundcard is connected analogue (for simplicity) to the 01V96 headphone socket and a couple of inputs.

This setup in XP Pro uses 18% of CPU and less than 5% of the ethernet connection when maxed, and is stable unless I pull the ethernet plug repeatedly. I can access the internet via wifi, and play videos to screen with the audio to the 01v96 without problems with the audio or the midi or the Video, but I would not recommend surfing as the audio will always go to the mixer and you might forget to mute the inputs.

In OSX the same setup works with Midipipe, but the Belkin Hub cannot access my Terratec Phase26 soundcard, so no monitoring or playback. I have just bought a Maya44 which might work, and which should anyway allow me to record the desk output at the same time.

The graphic representation of SM2 is not as pretty in OSX, and OSX has some problems mixing sample rates that XP handles easily (by re-sampling everything at 48K - YUK), but the setup is extremely stable, not caring one Hoot if the ethernet connection is repeatedly broken.

I am also working on foam pads for the BCF faders, as in quiet numbers or conference situations I do not dare to use the BCF’s unless I remember never to return the faders completely to Zero.

I received a couple of Devolo CPL's - CPL Devolo MicroLink dLan 200 AVdesk. They plug to the mains with a cable, which I prefer as I find that "on tour" anything with a plug built in gets broken. They tell me that they are capable of about 50mb/s when they are about 20M apart and on different phases (my house has 3phase power) and I have had them running my system for a day now without any dropout. I had put every lighting dimmer in the house at half power, and turned on all the fluorescents in the basement as well. I am now going to see whether I can access my setup from my nearest neighbor, who lives about 500M away. I will be pleasantly amazed if it works!

05/June/2010

The CPL’s don’t work 500Metres away. I am not really surprised. They do work 75M away on a different phase, and that will do just fine.

Most problems because the 01v96 USB connection and SM will not talk NRPN.

I was about to give a final setup plan, and as I was about to start typing I thought of a better (simpler) way of doing things – uh-oh. No- the NRPN cannot go back to the BCF’s directly (feedback) so have to connect via usb so that the NRPN returned by the desk is stopped by midiox.

If desk with me at control – use Terratec to feed NRPN from onstage BCF’s to the 01v96. Probably easiest to connect Main on-stage BCF in USB-4 mode, and chain additional units (in Stand-alone Mode 1) through “IN” and “OUTA” which are effectively in parallel with the USB.

A Problem when recording digitally and using two sound cards. Need to keep all of the sample rates the same – 48kHz.

A scene recall needs to be followed by a resend of CR data? No. I need to use “recall safe” abundantly for scene memories.

I have gone back to SysEx at 1023 resolution (plenty?) for my FOH BCF, as when continuously mixing on 8 faders there is too much delay using NRPN. It is fine for the onstage BCF’s at 16383 resolution even though the messages are being returned to my laptop via Ethernet and then sent back to the desk, as generally only one or two faders are being moved at any one time. NRPN is generating 64x as many messages as Sysex. I will have a go with Sysex at 16383 resolution (16x as many) and see how it behaves. Need to work a bit harder on buffer sizes.

David Kent

Whale AudioVisual