MSB REV 2

MSB Plus Rev 2

Owners Manual
Seventh Edition

142 Arena Street, El Segundo, CA 90245 U.S.A.

JLCOOPER ELECTRONICS

INTRODUCTION

Thank you for purchasing the MSB Plus Rev 2. Most manufacturers use this space to pat you on the back for laying down cold, hard cash to buy their product. We hope, however, that you'll find the MSB Plus Rev 2 rewarding by itself, without all the hype. We created the MSB Plus Rev 2 as a direct response to musician requests for a fully programmable MIDI routing box with internal data merging and processing. The "Rev 2" means it's the 2nd Revision, which features Program Change Command ("patch") mapping. State-of-the-art circuitry has been combined with sophisticated software techniques to implement many features never before found in a single, integrated package. We feel that the MSB Plus Rev 2 is the most useful MIDI accessory in the Universe.

JLCooper Electronics also makes an Editor/Librarian software package for the MSB Plus. It's called MSB+ Remote, and it is available for the Macintosh, PC and Atari computers. This program allows you to edit and store all of the MSB Plus internal memory to a computer. MSB+ Remote software operates as a "desk accessory", so that it can run along with another program such as a MIDI sequencer. Se enclosed catalogue/Order Form for more details.

Please take the time to study this owner's manual to get the fullest usage of this multi-faceted product.

Special Note for Advanced Users:
Once you've become familiar with the details of operating the MSB Plus Rev 2, we encourage you to use the Advanced User Quick Operation Reference at the end of the manual.

TABLE OF CONTENTS

CHAPTER 2 GENERAL DESCRIPTION

The MSB Plus Rev 2 is an eight input, eight output programmable MIDI routing system. In addition to behaving like a smart "thru box" and MIDI splitter, the MSB Plus Rev 2 includes a MIDI merger; two data processors, and the JLCooper Panic Button. The MSB Plus Rev 2 also has the ability to send out up to eight MIDI Program Changes at a time set up all your sound modules and signal processors.

Any MIDI input can be routed to any or all of the MSB outputs.

Any input can also be routed to either of the two data processors. The data processor's outputs can be routed to any MSB Plus Rev 2 output.

The data processors selectively filter out any class of MIDI Commands. The processors also include Transpose and Channel Bump functions. The Transpose function will transpose incoming MIDI Notes over a 5 Octave in semi-tone steps. The Channel Bump will change the incoming MIDI Channel data by any number of Channels.

The two data processors can be merged and the result sent to any MSB Plus Rev 2 outputs.

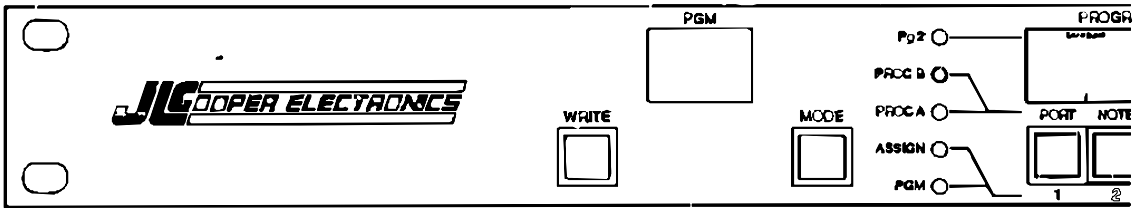
The Panic Button feature eliminates "stuck notes" and features a foot switch input for remote operation. The foot switch may also be assigned to step through the MSB's programs. Each time a new program is selected, the MSB may send out a burst of up to eight Program changes.

All MSB Plus Rev 2 parameters (routing, filtering, channel bumping, transposing, merging, program change commands and remote control input/channel assignments) can be stored as a “program”. There are 64 battery-backed programs.

Programs may be selected remotely via MIDI Program Change commands. As mentioned, both MIDI receive input and channel are programmable.

In addition, the MSB Plus Rev 2 will dump its internal memory via MIDI system Exclusive, to an appropriate storage device. The dump may be initiated from the front panel.

CHAPTER 3   FINDING YOUR WAY AROUND THE FRONT PANEL



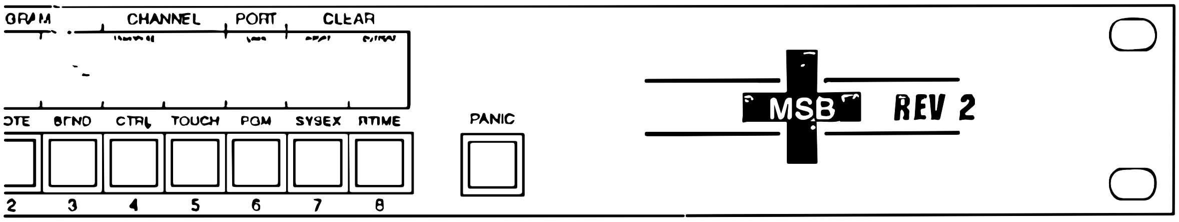
Refer to the picture above. We'll go from left to right.

The WRITE button is used for Writing programs to memory. Also, pressing the WRITE button once will return you to the PROGRAM mode. That is the mode used for recalling programs from memory.

(The WRITE button, when pressed while holding down buttons 1 and 8, is also used to initiate a bulk data dump. See Chapter 13 for specifics.)

The two digit **PROGRAM DISPLAY** shows what program number you're on. The display range is from 11 to 88.

The MSB Plus Rev 2 is a "smart" MIDI processing device, but in order to get it to do what you want, you must be sure that it is in the right MODE.



The **MODE** select button and its five LEDs are used to select just what kind of information you intend to enter into the MSB Plus Rev 2. The seven modes are listed here, along with the Owners Manual chapter number that explains each mode:

Program Mode	(Chapter 4)
Assign Mode	(Chapter 4)
Processor A	(Chapter 7)
Processor B	(Chapter 7)
Processor A, Page 2	(Chapter 7)
Processor B, Page 2	(Chapter 7)
Page 2, Program Change Command Mode	(Chapter 6)

The eight push buttons under the display are used to enter information, and the eight-digit display tells you what you've entered or what you've got stored in memory.

The **Panic Button** eliminates stuck notes. (Chapter 9)

Feel free to push some buttons to see what happens.

CHAPTER 4 BASIC SWITCH BOX OPERATION

This section describes the use of the MSB Plus Rev 2 as a simple MIDI routing device.

The first thing to do is to hookup your MIDI cables. Any device with MIDI Inputs, be it a synthesizer, computer, sequencer or drum machine, can be hooked up to the MSB Plus Rev 2 MIDI Outputs. Conversely, any of these devices with a MIDI Output can be hooked up to the MSB Plus Rev 2 MIDI Inputs.

Go into the ASSIGN MODE by pressing the MODE button once until the ASSIGN LED goes on.

The eight buttons below the display are used for routing assignments in the following manner:

Each button corresponds to an MSB Plus MIDI output (i.e. to a destination device). There are eight buttons, one for each output. The number in the display above the button indicates which MSB Plus MIDI input (source device) is sending MIDI to that destination.

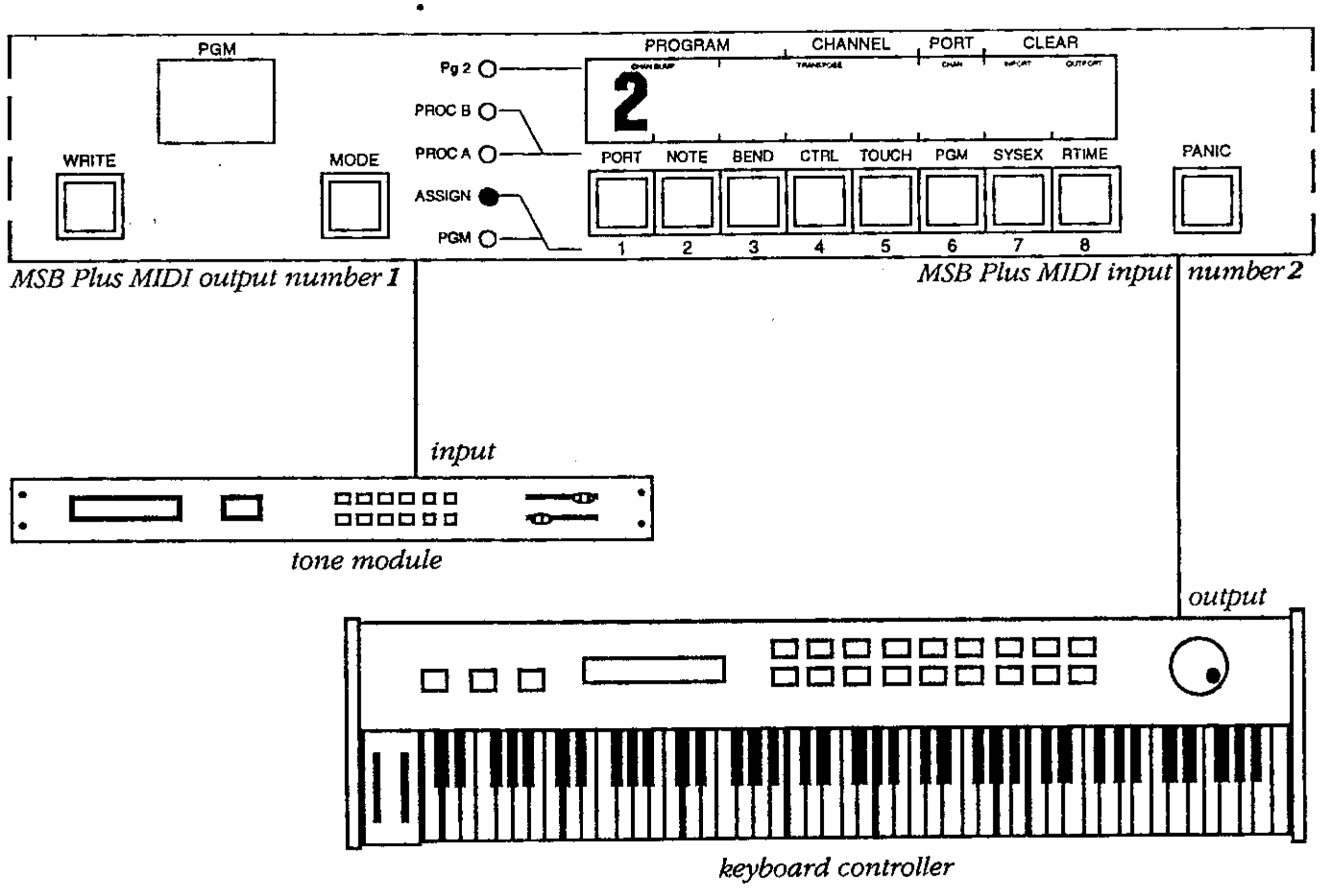
Just Remember:
Buttons 1 through 8 are the outputs (MIDI Destinations). That is, where the MIDI is going to.
The Lit-Up Numbers are the inputs (MIDI Sources). That is, where the MIDI is coming from.

Example: Plug a cable from the MIDI OUT #1 of the MSB Plus Rev 2 to the MIDI Input of a synthesizer or tone module of some kind. Go into Assign Mode by pressing the MODE button once. Tap button #1. The LED display above the button will count up 0,1, 2,3,4,5,6, 7, 8, A, B, C, as you tap the button. (If it doesn't count up, you are not in the ASSIGN Mode!)

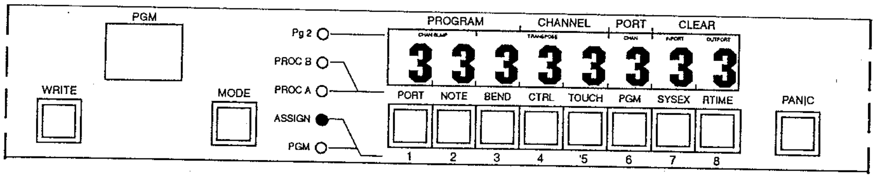
The A, B, C indicates assignment to the internal Processors, which we will cover in later chapters. After the C, the display indicates "0", which corresponds to the output turned Off, and the synthesizer attached to Output #1 will receive nothing.

The button you are pressing, in this case Button #1, is selecting *which of the* 8 *MSB Plus inputs are routed to the instrument that you have connected to output one.*

For example, if the display above button #1 shows the number "2", it means that the synthesizer or tone module will receive MIDI data from the controller that is connected to Input #2 of the MSB Plus Rev 2. Try it and see. Hook up a keyboard controller, MIDI Out, to the MSB Plus, MIDI Input number 2.

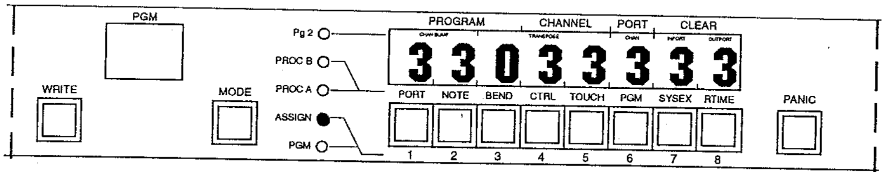


Example: Plug a cable from the MIDI Out of a keyboard to, say, Input #3 of the MSB Plus Rev 2. Now tap each output assignment button until all the displays show only "3"s. Now MIDI data from the synthesizer will be sent out of all eight outputs.



You've just created a one-input, eight output, "thru box", or "splitter".

Let's say that you hooked up both MIDI in and out of this synthesizer. And, for no other reason than simplicity, you hooked up the MSB output 3 to the input of the keyboard. In this case your "splitter" assignment should look like this:



Now you are sending data from input 3 to every output, except output 3. This is so you don't route an instrument back to itself. (This will not hurt anything; it has the same effect as looping a MIDI cable from your input right to your output. This layering would rob half of your instruments voices.)

IMPORTANT: Try not to route an instrument back to itself. Unless you are intentionally creating some special effect, you will likely only create confusion.

It is much easier to keep track of things if you run a thin strip of masking tape below the push buttons and label each button with the name of the slave devices, that is, the name of the device that each MSB Plus Rev 2 Output goes to.

After you've entered a "routing assignment", you may write it into one of 64 memory locations.

To write an assignment into memory, press and hold the WRITE button. While holding the write button, enter a pair of numbers, 11 through 88, to select the location of the program in memory.

Example: Say you want to enter the patch from the last example into program location 11 (the first program.) Hold WRITE, press "1 ", then press "1" again. Release WRITE. The PROGRAM DISPLAY will show "11". Notice also that as soon as you pressed the WRITE button, the MODE display LEDs went to the PROG RAM position, telling you that the next 1 thru 8 button push would be specifying the program number, not the routing assignments.

NOTE: The program memory and display of the MSB Plus Rev 2 use a modified Octal form rather than the normal decimal due to the 8 *select push buttons. That is, instead of showing (and selecting) patches* 1 *thru* 64, *you will use* 11 *thru* 88. *In this system, for example, patch* 21 *is directly after* 18. *In Chapter* 16 *there are conversion tables to translate between decimal program numbers and the MSB Plus's Octal.*

You may think of it in terms of Banks; that is, Bank 1 has programs 11 through 18, Bank 2 has programs 21 through 28, etc., up to Bank 8.

To recall an assignment from memory, enter PROGRAM MODE by pressing the MODE button until the PGM LED is lit. Then simply press any two of the buttons 1 through 8 to recall programs 11 through 88 from memory.

Initialization

To completely and irrevocably clear out all memory, unplug power, hold buttons one and eight, and, while holding, re-apply power. There is no "undo" function, so only do this if you are certain that you do not wish to save any configurations.

CHAPTER 5 CHANGING MSB PROGRAMS VIA MIDI CONTROL

The MSB Plus Rev 2 will respond to MIDI Program Change commands. This means that you can, with the touch of one button to your keyboard controller, completely re-wire your MIDI studio! As shipped from the factory, your MSB Plus is set to respond to Program Changes on MIDI channel 1, on Input 1.

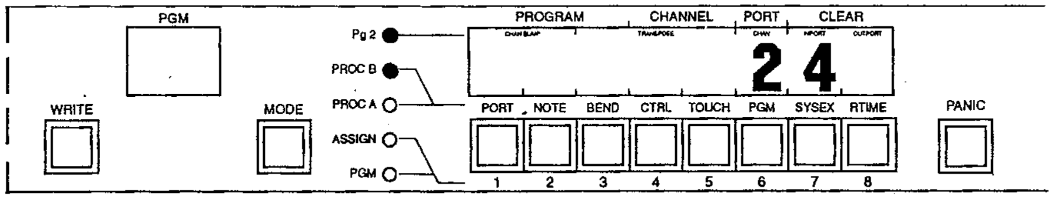
That means that, if your keyboard controller is plugged into input 1, whenever you change programs on your keyboard, the MSB Plus will change programs. Likewise, if your sequencer is connected to input 1, whenever you start your sequence, the MSB may respond to any program changes in the sequence and immediately change. Now, you might not want it to do this! This section will show you how to change the MIDI input and channel for receiving Program Changes, so you can choose if or when you want the MSB Plus to change programs via remote MIDI control.

If all you're interested in is making your MSB Plus stop responding to Program Changes, skip reading all of this discussion and follow the instructions on the top of page 15.

The reception of MIDI program changes will be on one user- selected MIDI channel (that is, 1 through 16) and on one user- selected MIDI input (that is, 1 through 8.) The selected input and channel we will call collectively, for sake of convenience, the "Remote Assignment". You can have a different Remote Assignment for each program, or the same Remote Assignment for all 64 programs.

To recall the input and channel that the MSB Plus Rev 2 will remotely respond to, tap the MODE button until the green PAGE 2 and either Processor LEDs are lit. The Channel and Input assignments are displayed above buttons 6 and 7.

For your convenience, we display this information in two places, PROC A + Pg 2, or PROC B + Pg 2. But remember, this has nothing whatsoever to do with Processors A and B.



In the above example, when a Program Change command on MIDI Channel 2 comes into the MSB Plus Input 4, the MSB Plus Rev 2 will respond by changing to that program number.

WARNING:

Never send the MSB Plus Program Change while MIDI data is flowing through it. This would be analogous to pulling out and quickly rearranging 16 MIDI cables in the middle of a musical performance. If this is done accidentally, you may need to use the PANIC function.

To change the receive Input and Channel, simply tap the buttons below the display. Use button #6 to change the Channel and button #7 to change the Input. The Channel display will step from 1 to 9, then to:

"A" for Channel 10
"b" for Channel 11
"C" for Channel 12
"d" for Channel 13
"E" for Channel 14
"F" for Channel15
"0" for Channel 16

This assignment may now be written into memory as part of your program. In most applications, you'll want all 64 program locations to keep the same remote input and channel assignments (unless you plan to switch around the "Master Controller".)

We have provided a short-cut approach to "globally" write the remote receiving input and channel into all 64 memory locations.

Here is a "hidden function" on the MSB Plus Rev 2 that only clever people like you who read the manual will know about. In order to assure that MIDI Program Change commands will be recognized by the MSB Plus Rev 2 on the same channel and from the same input, regardless of the program last recalled, do the following:

Press and hold the WRITE button, then press the Panic Button. This will write the remote receive channel and input assignments from the current "working buffer" into all 64 patches.

When sending program changes to the MSB Plus, remember that since the display is in octal that you'll need to refer to the Table (Chapter 16) to determine what MIDI Program number to send from normal decimal equipment to the MSB Plus Rev 2.

Example: To recall MSB Plus Rev 2 program #21 remotely, the MSB Plus Rev 2 will need to receive MIDI Program change #9.

The Remote Assignments must be chosen carefully, as they can be a great convenience or a great nuisance. It is strongly recommended, that you dedicate a MIDI channel to the MSB Plus that is not used by another instrument in the system. That way, a program change intended for the MSB Plus will not directly affect a tone module. And a program change intended for a tone module will not re-wire your studio.

If your MSB Plus is responding to Program Changes, and you do not want it to, here's how to make it stop doing this:

Tap the MODE button until both the red and green LEDs are on, that is PROC A or PROC B (either one) and Pg 2.

The numbers above buttons* 6 *and* 7 *set the MIDI channel and input that the MSB will respond to* a *program change on. Notice the words CHAN and INPORT that are labeled on the red plastic lens of the MSB above the display.

As shipped from the factory, both of these numbers are set to* 1. *Change these numbers to* 0, 8, *then hold in the WRITE button. While holding in the WRITE button, press and release the PANIC button.

(You have now programmed all 64 programs to only respond to a MIDI Program Change if and only if the MSB Plus receives a MIDI Program Change command on input number 8, on MIDI channel 16. This is not likely to ever happen, unless of course, your sequencer is hooked up to MIDI input number 8. If this is the case, chose some other INPORT number. For example, put a 7 above button 7. Then hold in the WRITE button, while holding in the WRITE button, press and release the PANIC button.)

You cannot completely "turn off" program change reception, but you can set it for some unlikely combination of input and channel number.

An Example: Say that your master controller synthesizer in your whole MIDI set-up is a Korg M-l. Say you have chosen for whatever reason to attach the M-l MIDI Out to the MSB Plus Rev 2 Input #4. Before every song in a set, you would like to press a single patch select button on the M-l, and have it send a Program Change command to the MSB Plus Rev 2. The MSB Plus Rev 2 will respond by recalling the appropriate program for routing, merging, transposing, etc.

Solution: Select PROC A (or B) and Pg 2. Using buttons #6 and #7, enter Channel 1 (which the M-l sends on) and Input 4. Press and hold WRITE, and while holding WRITE, press PANIC.

Regardless of anything else going on in the system, the M-l will always maintain remote control over the MSB Plus Rev 2.

There may also be times when you do not want to use this shortcut; that is, some of the 64 programs may require alternate remote assignments.

Example: You're still on the M-I, but your friend on the WaveStation wants some control of the MSB Plus too, say while you have to run out and put more money in the meter during the chorus. He is sending MIDI data on Channel 8 and is plugged into MSB Plus Rev 2 Input #2.

Solution: Don't use the shortcut. Not all programs need the same remote assignment now. Simply write the "Channel, Input 4 " assignments into the programs where you want control, and write "Channel 8, Input 2" on programs that you wish to "hand over control". Remember that you will need to bring up the patch that has him in control.

CHAPTER 6 PROGRAM CHANGE COMMAND MAPPING

The MSB Plus Rev 2 can store and send sets of MIDI program change commands. Whenever a programmed assignment is recalled from memory, (either from the front panel, foot switch, or remotely via MIDI), the MSB can send out up to eight MIDI program change commands, one for each output.

This set of eight "patch changes" is unique to each program assignment.

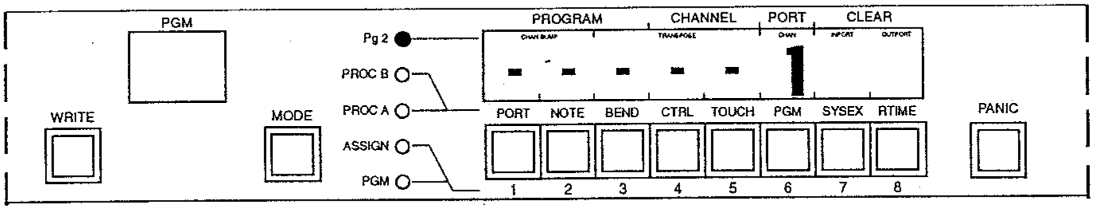
Here is an example of why one might want to use this feature. Say that you have a keyboard controlling four slave tone modules and a couple of MIDI reverb units. For a particular song you select patch 5 on your keyboard. The keyboard dutifully sends a MIDI program change command out to the slaves and the effects. Thus, all of the slaves go to "5". As you can see, this arrangement is not musically useful, unless you've spent hours copying and moving patches around in your slaves and effects. (Consider the improbability of every synthesizer manufacturer agreeing that all patches numbered "5" must sound good layered together.) So between songs you have to walk over to each slave (in the dark, no less) and manually select the patches for the next song.

With program change command mapping, all you have to do is recall a single program on the MSB Plus Rev 2. Pre-programmed sets of patch changes go out to all the instruments so that they are perfectly set up for the next song.

(Instead of all them going to "5", one can go to "13", another to "125, another to "7", or whatever.).

To enter program change commands, tap the mode button until only the green Pg 2 LED is lit.

The display will look like this:

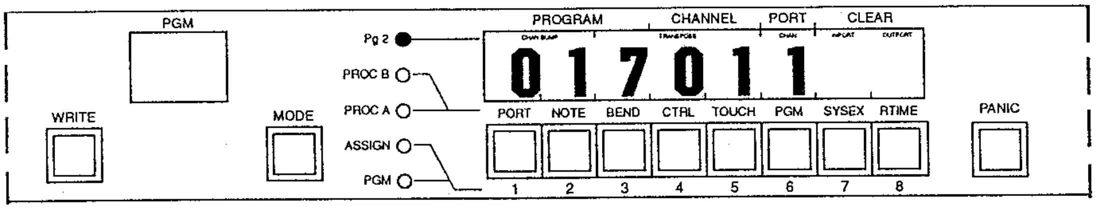


You will be entering which output port (destination device) will get the patch change command. Use button 6 marked PORT.

To enter the program change number to be sent use buttons 1,2, and 3 marked PROGRAM.

To enter the MIDI channel of the program change, use buttons 4 and 5 marked CHANNEL.

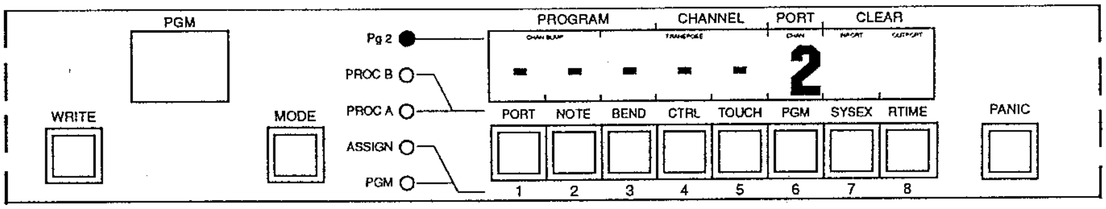
For example, say you want the MSB Plus Rev 2 to be able to tell the instrument connected to MSB output number 1 to go to patch 17. The instrument is set to receive on MIDI.channel1. On Pg 2, you would enter the following:



Now write this to a memory location, say "23" for example. As you write it to memory the MSB will actually send the program changes out, to allow you to verify the results.

This particular change will now be sent out of the MSB Plus Rev 2 every time you recall program 23.

You could have gone on an entered program changes for any or all of the eight output ports. Enter Pg 2 mode again and tap the PORT button. The display will step to the next output port number.



You may enter program information or merely skip over that port. And again, once you have entered up to eight program change commands, you may write the set into memory, along with your routing, processing, and merging assignments.

To edit (remove) a program change command, hold buttons 7 and 8, marked CLEAR.

After you have cleared a program change command, the PORT number will advance to the next step. **Once you have cleared and/or re-entered your program changes, you will need to rewrite the program into a memory location.** (The CLEAR function only affects the so-called "working buffer", and does not actually remove the program change from the MSB Plus Rev 2 memory).

The program change commands are sent out when you write them to memory, when you recall a program from memory, or when a remote program change command is received from some external MIDI device. (See Chapter 5, Changing MSB Programs via MIDI Control.)

Alternately, you may step through the programs in your MSB sequentially using the foot switch. Each time you step to a- new program with the foot switch, any program change commands stored in that program will go out. To assign the foot switch to do this, see Chapter 10, The Foot switch.

Some notes about Program Change Command mapping, triggered by Changing Programs via MIDI control.

There may be situations where you want to combine these two operations. That is, you may want to select a program (using your keyboard controller) to set-up the MSB Plus Rev 2 and send program change commands to the various instruments. As we said earlier, this can be done, but it does require a bit of extra planning on your part.

Keep in mind that if a controller is routed to some instruments, the instruments will first receive the original patch change sent from your controller. Then they will receive the correct, pre- programmed, patch numbers.

If, however, the assignment includes some merging or processing, you may need to select the program on your controller *twice.* Once to set up the MSB, and one more time to be sure that the program change set-up arrives at the appropriate destination instrument.

Because two program changes are going out at the same time, sometimes it is difficult to determine which program change the destination instrument will receive, especially if the routing assignment uses merging and processing. In case you do not get the desired results, try the suggestion below.

Example of a Possible Problem:
You have programmed the MSB Plus to send a program change command Number 14 on MIDI channel 1 to output number 3. This is stored in Program 11 of the MSB Plus. Your intention is to press program number 1 on your controlling keyboard, so MSB will change to program 11. The keyboard is transmitting on MIDI channel 1. When you select program 1, you want the instrument on output 3 to go to program 14. But sometimes, mysteriously, it goes to program 1 instead.

Example of Solution:
Set the MSB to receive on (for example) Channel 8 To do this, go to PROC A + Pg 2 or PROC B + Pg 2, that is, both the red and green mode LEDs on. Push button number 6 repeatedly until an "8" shows (above button 6, and below the word "CHAN"). Write this globally into all memory locations by holding down the WRITE button, and, while holding down WRITE, press PANIC. (See last section if you need more details about this.)

Now, whenever you want the MSB Plus to change programs, change the transmitting channel of your keyboard controller to 8 first. Then press program 1. Then set the transmit channel back to channel 1.

CHAPTER 7 FILTER / DATA PROCESSOR OPERATIONS

This section describes the two data processors, PROC A and PROC B, and explains how to use them in an assignment. Be patient please, this is a powerful feature.

General Description of the Processors;

A processor is assignable to any MSB input. Since you have two to work with, at any given time two inputs can be "processed". Data coming into the assigned inputs is "manipulated" in specific ways. The altered data can then be sent on to any MSB Plus MIDI output or outputs.

Here is what you can do with the incoming data: FILTER, TRANSPOSE, and CHANNEL BUMP.

Processor Features and Suggested Applications: Data Filter Applications

Suppose you have a sequencer sending note data on 16 MIDI channels. The note data represents a keyboard performance. You also want the sequencer to send MIDI clock information to a drum machine. A Processor may be used to filter out note information so that the drum machine receives only MIDI clocks. That way the notes from the sequencer will not trigger the drum sounds.

You are playing a lead synthesizer, while a slaved module is playing just a bass line. You may want to change patches on the lead, but not on the module playing the bass sound. A Processor may be used to filter out program change commands so that they do not affect certain slaves.

Perhaps you have an older synthesizer that crashes whenever it receives an unfamiliar MIDI message, like song position pointer. Or, perhaps you just want to speed up the response time of tone module by removing excess controller messages.

Processors can all be used in both of these applications for general purpose MIDI troubleshooting and problem solving.

You may selectively filter out anyone or more of the following MIDI commands:

Note Commands
Controller Commands
After-Touch Commands
Pitch Bend Commands
Program Change Commands
Real Time Commands
System Exclusive / System Common Commands

<u>Channel Bump Applications</u>

Channel Bump shifts the MIDI channel of all data passing through the Processor. It does this by adding a user-selected interval to the incoming MIDI channel number.

Many synthesizers, such as early Yamaha DX-7's, only transmit on MIDI channel #1. Channel Bump allows these instruments to transmit on any channel or even two channels at once. A sequencer sending on channels 1 through 8 may be bumped up to send on channels 4 through 11.

Channel Bump solves a common sequencing problem, too. Suppose that MSB Plus Rev 2 is assigned to route a sequencer to several MIDI slave tone modules, each set to receive on a different MIDI channel. For example, your modules might be set to receive on channels 1 through 3.

Then you recall a different MSB Plus Rev 2 assignment to route a keyboard controller to the same slaves. But your keyboard is sending on channel 1. Most keyboards can send only on one or two channels at the same time.

So, the keyboard would not be able to control the slaves, unless you were to walk over to the rack of tone modules and change the receive channel of each tone module to channel 1. This is a time-consuming and difficult operation, since a receiving channel number is generally not changeable remotely via MIDI.

Using Channel Bump solves this problem. Two processors can be assigned to the same input, each with a different channel bump. This would allow your keyboard controller to send on three different MIDI channels. Thus the tone modules can always be left on different MIDI channels than the keyboard.

Use buttons 1 and 2 to bump the desired channel up or down. Button 1 toggles the minus ("-") sign and button 2 cycles the number up from 0-8 and back down again in a loop. The following table shows how this system corresponds to the 16 MIDI channels:

MSB+	MIDI	MSB+	MIDI
0	No bump	8	Bump up 8 channels or down 8 channels
1	Bump up 1 channel or down 15 channels	-7	Bump up 9 channels or down 7 channels
2	Bump up 2 channels or down 14 channels	-6	Bump up 10 channels or down 6 channels
3	Bump up 3 channels or down 13 channels	-5	Bump up 11 channels or down 5 channels
4	Bump up 4 channels or down 12 channels	-4	Bump up 12 channels or down 4 channels
5	Bump up 5 channels or down 11 channels	-3	Bump up 13 channels or down 3 channels
6	Bump up 6 channels or down 10 channels	-2	Bump up 14 channels or down 2 channels
7	Bump up 7 channels or down 9 channels	-1	Bump up 15 channels or down 1 channel

So, if your controller transmits on MIDI channel 10 and one of your sound modules only receives on MIDI channel 8, set the Channel Bump to –2. More details for the procedure are on pages 25 and 26.

Transpose Applications

Transpose is used if you want some slaved instruments to play, for example, an octave higher or lower than the master. This of course may be used for special effects and static harmonies, with one slave tracking a fifth higher, another a third lower, etc. Also it is much easier to program MSB Plus Rev 2 to transpose MIDI Notes, rather than to change the key of each receiving instrument.

MIDI notes coming into a processor may be transposed plus or minus five octaves (59 semitones) in semitone steps.

To Use the Processors:
To use the processors, you will first need to assign them to an input. That is, which "source" do you want to process?

You will then need to assign the processor to an output, or some combination of output. That is, which destinations do you want to receive processed data?

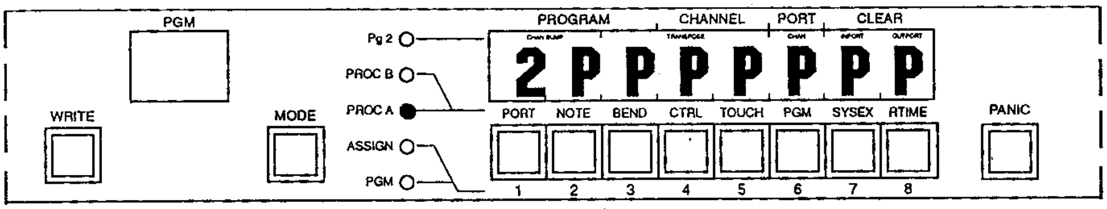
That’s done by putting an “A”" or “B” in the LED display while in Assign mode.

To Assign a Processor to an Input

Tap MODE button until "PROC A" LED is lit, The first Cleft hand) number (over button #1) displays the current input assignment of processor A. Tap button #1 to change this assignment.

Example: Our old DX-7 is going to the MSB Plus Input 2. We want to manipulate (process) the MIDI data coming from the DX- 7) so that later we can send the new, processed data on to another synthesizer.

Solution: Assign PROC A to input 2 using directions above.

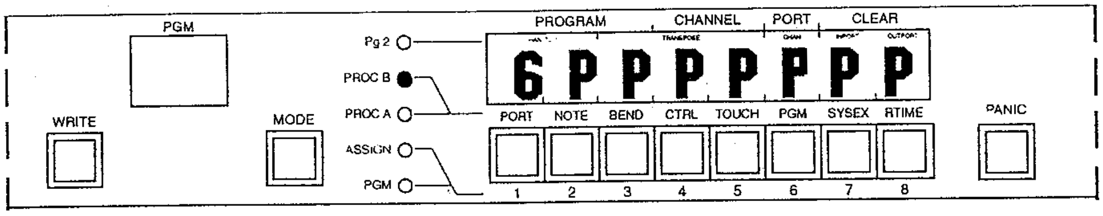


The PROC B is assigned to an input in a similar manner.

Tap MODE button until "PROC B" LED is lit. Tap button #1 to select the MIDI input assignment.

Example: We know that we're going to process our DX- 7, but we also want to process the data coming from our VFX, and we want to process it differently. The VFX MIDI Out is going to the MSB Plus Rev 2 Input Number 6.

Solution: Use Processor B. Assign it to input number 6.

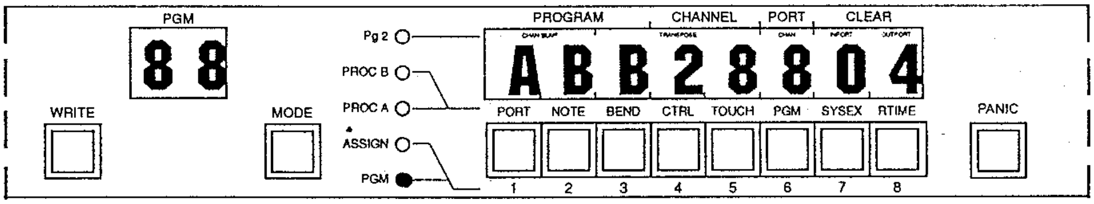


To Assign a Processor to an Output

Return to Assignment Mode. Tap an output assignment button, and it will display a count up to 8. After 8, an "A" appears. This means that PROC A's processed data is being sent to that MSB output.

Tap the button again and the display shows a "b". The output of PROC B is now being sent out of that MSB output.

Example: Write this assignment into memory location 88 by holding down the WRITE button then pressing "8" twice.



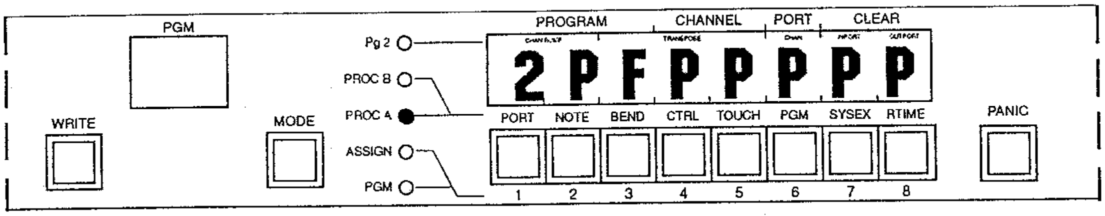
This shows that Output 1 is putting out data that has been processed by PROC A, and Outputs 2 and 3 are putting out data that has been processed by PROC B.

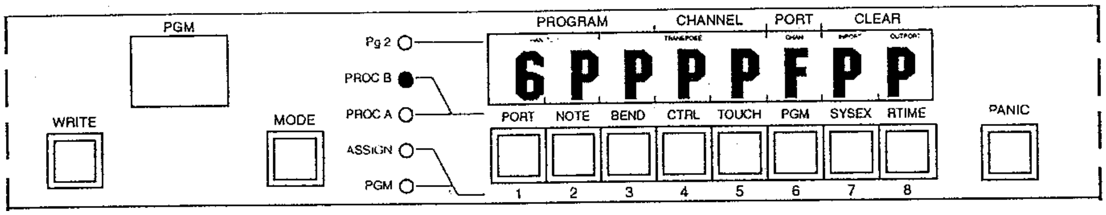
Now we will discuss how to assign the processor's functions.

<u>To Filter a MIDI inputs data</u>

First, assign a processor to the input using the directions above. See the row of "P"s and "F"s? "P" means "Pass" and "F" means "Filter". This indicates whether a given MIDI Command will Pass thru or be Filtered out on its trip thru the Processor, Hit the button under one of the Filter functions. Notice that it toggles between "F" and "P", Use the guide directly below the display to determine which MIDI Command is being affected for each button. Any combination of "P" and "F" may be entered.

Example: Set Processor A to filter out Pitch Bend Data from our, DX-7, and set processor B to filter out Program change Commands from our VFX. (The input assignments are borrowed from the last example.) Refer to the figure for the solution.





Remember that you must also select an output for the processor, that is, which slave is intended to receive the processed data. This is done by returning to the Assign mode. Place an A or B above the buttons that correspond to the outputs that you want to receive processed data.

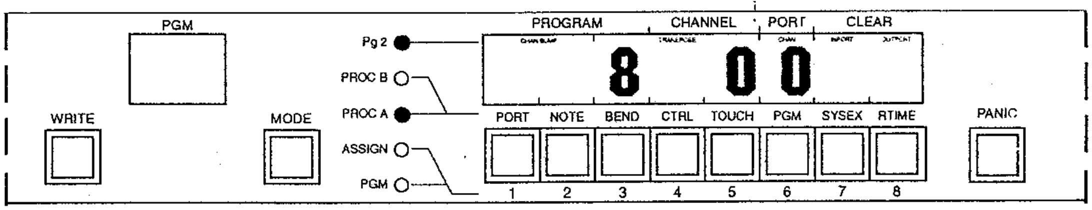
Io Channel Bump a MIDI Inputs data

Tap the MODE button until the green "PAGE 2" and the "PROC A" LEDs are lit. You are now in "PROC A, Page 2" MODE. Use buttons #1 and #2 to enter the increment or decrement of "BUMP." Button #1 toggles the minus sign) and button #2 steps the number from O to 8. The actual working range is -7 to +8, which covers the entire range. When you don't see the minus sign, the plus is implied.

Note: When you go to "PAGE 2") the last 3 digits in the display (right hand side) have nothing whatsoever to do with the processors. They are used in remote and System Exclusive operation (see Chapters 5 and 13) and we "borrow" the Page 2 mode to display and alter them. Actually, they identically show up in both Page 2 Modes (PROC A and PROC B). Ignore them for now.

Example of Channel Bump: You want PROC A from the last example to make your DX-7 send on Channel 9, not its normal Channel 1.

Solution: Go to PAGE 2, PROC A and assign the "bump" value by 8.



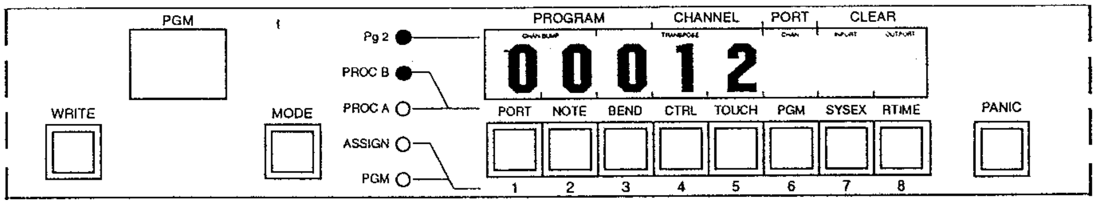
Processor B bump is assigned in a similar manner. Use PAGE 2, PROC B Mode.

To Transpose a MIDI Inputs data

While in Page 2, the next 3 digits to the right of the Channel Bump represent the Transpose interval. The first "digit" indicates whether the transposition is up (blank digit) or down (minus digit). The next two digits show the amount of transposition in that direction. In each case, the button directly below the digit affects the status of the digit.

Example: Use PROC B, which we assigned to the VFX. You want the MIDI note data from the VFX to go out of the processor an octave higher than you played it.

Solution: Enter a transpose interval of + 12 semitones. (Remember the "+" doesn't show; it is implied by the absence of the "-" sign.) While in Page 2, PROC B mode, use buttons 3, 4, and 5 to make the display show a 12.

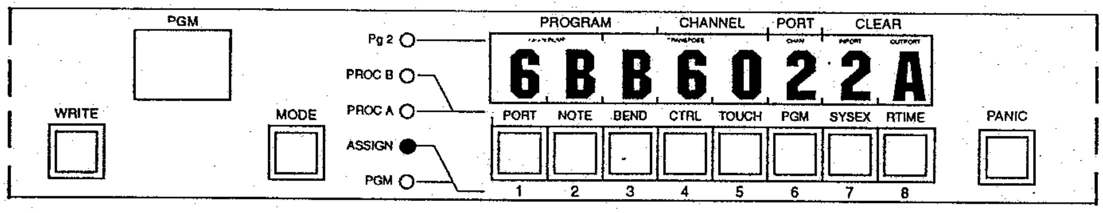


Along with basic routing assignments, the processor assignments can be written into memory as part of a program.

A Big, Hairy Example: Using the DX- 7/VFX example earlier, construct a patch that will give the following output assignments:

Output #	Controlled by
1	VFX (As actually played)
2	Filtered/Transposed VFX
3	Filtered/Transposed VFX
4	VFX (AS actually played)
5	(Off)
6	DX-7 (As played)
7	DX- 7 (As played)
8	Filtered/Channel Bumped DX-7

Solution:



You can see now that both processed and unprocessed versions of MIDI data are available simultaneously.

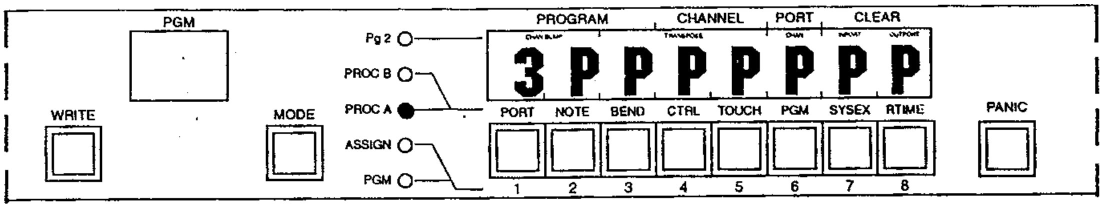
Study this example. You're almost home.

CHAPTER 8 MERGER OPERATION

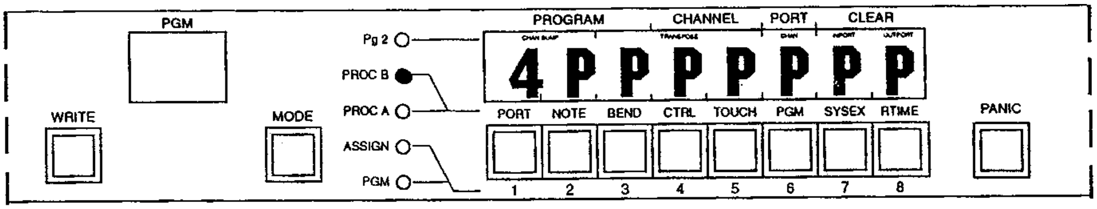
The Merger combines the output of the two data Processors A and B. The combined MIDI data will be sent to any output or outputs simply by calling up a "C" (for "Combined") on the display, in the ASSIGN mode.

Example: You wish to merge the MIDI Out of your Roland D- 50 with the MIDI Out of your MC-500. The MC output is hooked up to the MSB Plus. Rev 2 Input 3. The D-50 Output is hooked up to the MSB Plus Rev 2 Input 4. You want the merged data to feed your K-3 slave tone module, which is hooked up to the MSB Plus Rev 2 Output 1.

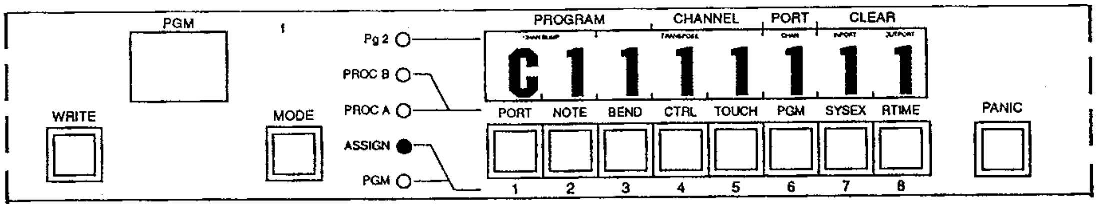
Solution: Tap the MODE button until PROC A LED is lit. Assign processor A to Input number 3.



Tap the MODE button until PROC B LED is lit. Assign Processor B to Input number 4.



Go back to the ASSIGN mode and assign Output 1 to display "C", which is the output of the merger.



Merger assignments can be stored in memory as part of a program, along with routing assignments and processor assignments.

CHAPTER 9 THE PANIC BUTTON

There are those dreaded moments in any MIDI musician's life when a mysterious note or chord gets "stuck". This can happen when the musician does something wrong, like change a MIDI Switch Box assignment at the wrong moment. But there are definitely times when there seems to be no explanation at all, other than "that happens every once in awhile". When this happens during a rehearsal, it is annoying. When this happens in the studio, it is costly. When it happens on stage, it is disastrous. To the rescue is the Panic Button section of the MSB Plus Rev 2.

When activated, the Panic Button sends MIDI Commands out of all eight output ports telling all slaves to "unstick".

You may activate the Panic Button from the front panel, or from a remote foot switch. The display will read "PAnIC" The MSB Plus Rev 2 will send a lot of commands, trying just about everything to unstick that note. It will take about 5 seconds for all of these commands to go out, but in general, it is best to release the Panic Button as soon as you hear the note unstick. On some synths, their input buffers can become filled with the high volume of Note Off commands that are sent. This should cause no problem, but should be avoided just in case.

In our lab tests, however, notes usually unstick themselves in less than a second. It really depends on the receiving synthesizer. Whether or not it responds to the more common All Notes Off command, which the MSB Plus Rev 2 sends out first, along with Sustain Pedal Up commands, which helps in some cases. After these commands are sent, individual Note Off commands are sent starting on Note 1, Channel l on up.

The panic feature may also be activated by a user-supplied foot switch. See next section for details.

CHAPTER 10 THE FOOT SWITCH

An external, user-supplied foot switch may be used for either program advance or to engage the PANIC function.

When the switch is assigned the job of program advance, it will cause the MSB to step to the next program. And, of course, if the new program has program change commands stored in it, these will be sent out as well. This allows you to set up the sounds of all your slaves with a touch of a foot.

To change the function of the switch, hold the mode button and engage the switch.

That is, if the switch had been set for PANIC, following the above instructions would re-assign the switch to control program advance.

Never use the foot switch to effect a patch change while data is flowing through the MSB. This would be analogous to pulling out and quickly rearranging 16 MIDI cables in the middle of a musical performance.

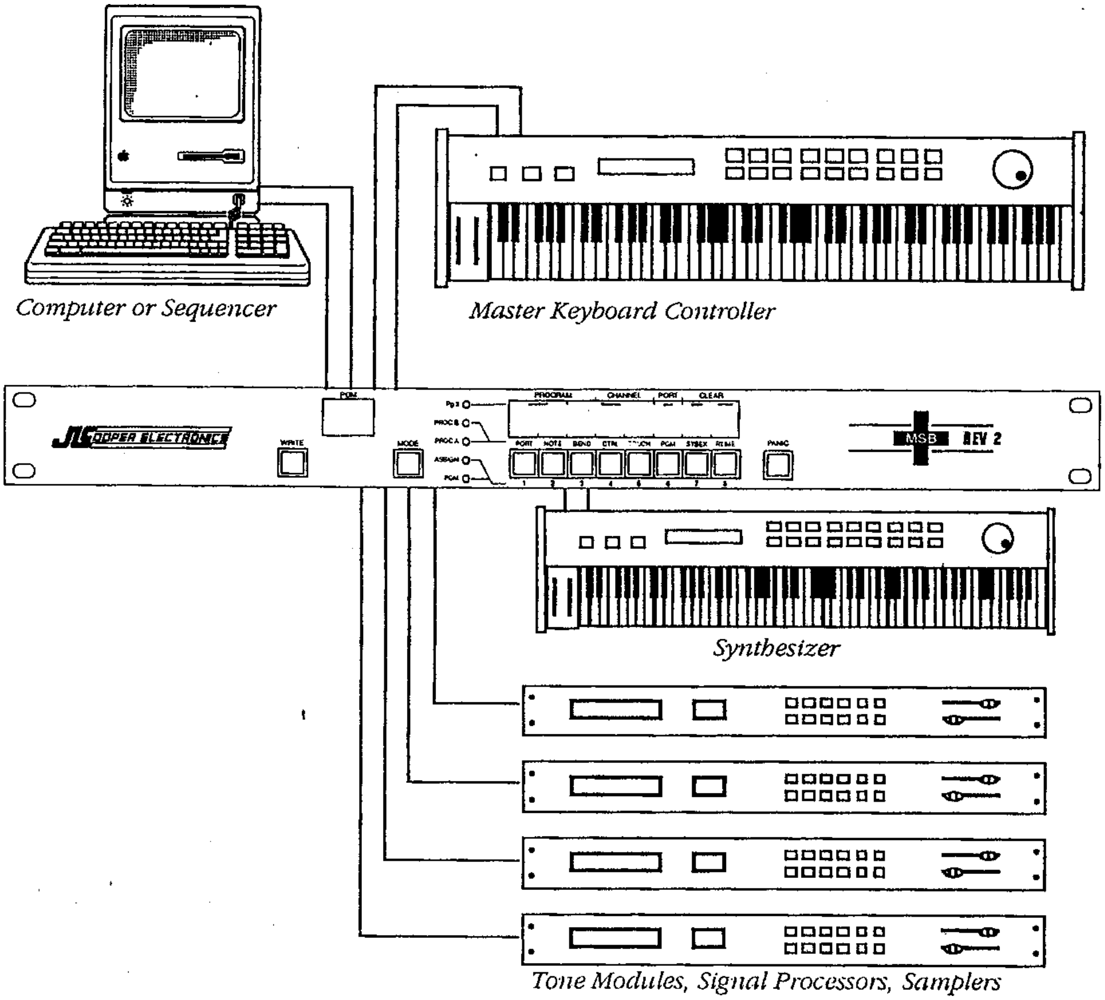
A "normally open" or "normally closed" momentary foot switch may be used. The computer in the MSB Plus Rev 2 tests the foot switch when the power is first turned on to determine what type you're using.

For this reason, be sure that the foot switch is plugged in before you turn on the power.

CHAPTER 11 HOOK-UPS AND SUGGESTED APPLICATIONS

Our customers are invariably more creative than we are in the application of our equipment. Every day someone calls or writes us with some new usage for a product that never occurred to us. Here are three examples to get the wheels spinning.

Example 1:
The Basic MIDI System: Write your routing assignments with programs to allow either keyboard to record on the sequencer, and programs to turn off various modules, and programs to allow the sequencer to control some modules while the keyboard controls others.



Example 2:
Using the Filters: Set up a good pad or wash sound on a module, and a bright lead sound on your keyboard controller. Use a filter to remove program change commands from the data stream. This will enable you to change your lead sound in the middle of a solo, while the background sound stays constant. Filter out note commands from a sequencer so that a drum machine only receives real-time clock. This way only the drum machine's internal song will play, and the keyboard parts in the sequencer will not accidentally fire the drums.

Using the Processors: Program the processors to present two different transpositions of the same part and listen to the results.

Using the Merger: Combine the output of a drum machine (for timing) with a keyboard part and send the data to a sequencer. Filter out the note data from the drum machine. Now you have "two inputs" on your sequencer, so you can record a keyboard part while it is externally synced.

Variations: Program Merging of Master and Remote keyboards. Program Filtering of MIDI sync to drum machine. Program Merging of sequencer and keyboards. Program Filtering of Program Changes to slaves from one or both sources. (Remember that you have 64 Programs to work with.)

Example 3:
Sending on two Channels at the same time: No one said that PROC A and PROC B couldn't be assigned to the same input jack! That means you could have a given instrument send two different transpositions out of two ports at the same time, even with different Channels. Of course the original data could still be available out of other ports.

But what if you merged the Processors now? Any output assigned to "C" would get two sets of MIDI data for every one received by the MSB Plus Rev 2. Each set would consist of processed data.

For example, set one processor to transpose an input up an octave. Set the other to transpose down an octave. Assign both to the same input. Now merge them and send the merged output to a slave synthesizer. Now when you play a note on the Master, the slave plays two notes, one up an octave and one down an octave.

You can also try two different channel assignments, which we present as a solution to a very common problem.

Problem: A band has one song in a set in which 3 slave synthesizer/tone modules are being controlled by a sequencer. Each tone module is set to a different MIDI Channel, because each is playing a separate "track" of a multi-track sequence. During this song, the lead keyboardist is playing only his own synthesizer, or perhaps one additional slave.

In the next song, however, the lead keyboardist wants control over some of those slaves that the sequencer had been playing. But his synthesizer only sends on one channel, and those slaves are each on a different channel! So, switch-box or no switch box, the musician has to walk to the rack and furiously push buttons in the dark to change the slave's MIDI receive channel. Ever use a TX- 7? (Certain tone modules require a phenomenal number of button pushes to effect a channel change!)

MSB Plus Rev 2 Solution: On the first song, the sequencer is patched to the slaved modules as in a normal setup. For the second song, however, a very special MSB Plus Rev 2 program is called-up. PROC A and B are both assigned to the master keyboard. PROC A bumps the channel up by, say, one. PROC B bumps the channel up by, say, two. The Processor outputs are then routed to two of the slaves. Meanwhile, a slave that lives on the same channel that the master keyboard sends on is directly patched to the master.

In other words, the keyboard data could come out of MSB Plus Rev 2 output 1 on MIDI channel 1, output 2 on MIDI channel 2, and output 3 on MIDI channel 3.

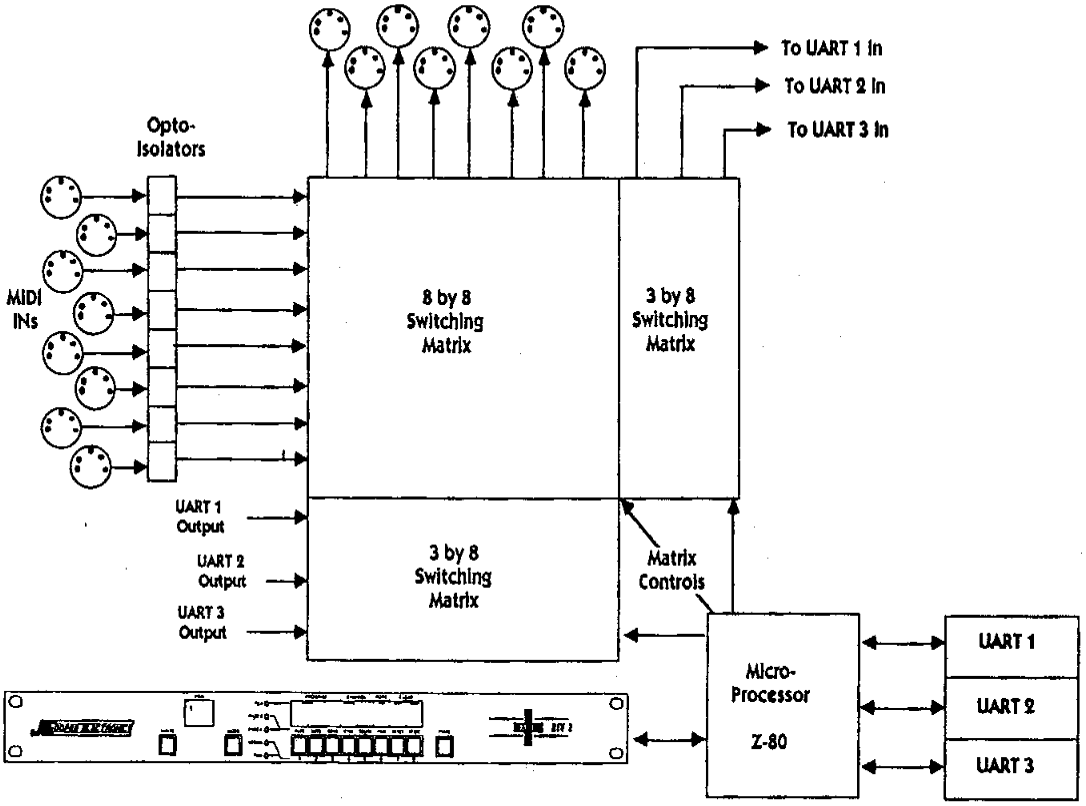
CHAPTER 12 THEORY OF OPERATION

The MSB Plus Rev 2 primarily consists of an.8 by 8 digital switching matrix that is under the direct control of a Z80 microprocessor. Also attached to the Z80 is a low-power memory that is powered by a Lithium battery. It is in this memory that the programs reside.

The MSB Plus Rev 2 also has a set of three "Universal Asynchronous Receiver / Transmitters" (UARTs) which may be (under Z80 control) attached to any input or output. These three UARTs are the path thru which the Z80 can communicate with the various input/output ports.

One of the UARTs is responsible for receiving MIDI Program Change commands. So whatever Input jack is assigned for Remote operation on PAGE 2 has this UART's input electrically attached to it. This UART's output is either used for the Combined output 9r for the System Exclusive output.

The other two UART's inputs are the PROC A and PROC B inputs, and their outputs are used for the outputs of these Processors.



CHAPTER 13 SYSTEM EXCLUSIVE

The MSB Plus Rev 2 will receive System Exclusive information on the same Input jack that is assigned to receive Remote Program Change commands. The SysEx dump output is programmable and can be assigned to any MSB Plus Rev 2 Output.

To initiate a MIDI System Exclusive bulk data dump from the MSB Plus Rev 2, hold down buttons 1 and 8, and, while holding both 1 and 8, press the WRITE button. The data will come out of the output port you have selected.

To Recall and assign System Exclusive receive and send ports, tap the MODE select button until in one of the PAGE 2 Modes. The receive input number will be displayed above button #7 and the send output number will be displayed above button #8. To change assignments, tap buttons 7 and 8 respectively. This can be written into a single program location (along with the Channel assignment used for Remote operation) as a part of the over-all program, or may be written into all program locations using the short-cut method outlined in Chapter 8.

The MSB Plus Rev 2 has several System Exclusive Modes for the advanced user. These Modes allow, for instance, the data contents to be loaded/dumped to a storage device such as a librarian program. Of more interest is the ability to "inform" a computer just what the current configuration is, so that it may display, in plain words, what the hook-up is like. At the time of this manual's printing, JLCooper makes MSB Plus Editor/Librarian software for the Macintosh and Atari computers, and a Windows version for PC's is in the works.

System Exclusive Implementation

This information is provided for the advanced user or programmer who wants to create his or her own program to control the MSB Plus.

Two different System Exclusive implementations are provided. One is for the original MSB Plus; the other is for the Rev 2. I

The original sets of commands will load/dump the old data only (no program changes). The new sets will load/dump both the old data and the program changes.

All System Exclusive commands must start with:

FOh	-Start of System Exclusive
15h	-JLCooper Electronics ID code
OBh	-MSB Plus Rev 2 ID code

After this, any several possible "Action Codes" are possible, possibly followed by sum amount of data. Finally, an End Of Exclusive code (OF7h) is required.

Recognized Receive Action Codes:

Command Extension Data
Oh none 1024 bytes
"Old" Full Memory Dump (no program commands).
1 none none
"Old" Full Memory Dump Request. Upon receipt, the MSB Plus Rev 2 will respond with an "Old" Full Memory Dump.
2 Pgm # 16 bytes
One "Old" Pgm Dump. This data will go into this pgm position without altering the program change data.
3 Pgm # none
One "Old" Pgm Dump Request. On receipt, the MSB Plus Rev 2 will respond with an "Old" Pgm Dump.
4 none 16 bytes
"Old" Working Buff Dump. This data will go into working buffer and not affect memory locations.
5 none none
"Old" Working Buff Dump Request. On receipt, the MSB Plus Rev 2 will respond with a Working Buffer Dump.
10h none 2560 bytes
"Complete" Full Memory Dump (including program commands.)
11h none none
"Complete" Full Memory Dump Request. Upon receipt, the MSB Plus Rev 2 will respond with a "Complete" Full Memory Dump.
12h Pgm # 40 bytes
One "Complete" Pgm Dump. This data will go into this pgm position, complete with program change data.
13h Pgm# none
One "Complete" Pgm Dump Request. On receipt, the MSB Plus Rev 2 will respond with a "Complete" Pgm Dump.
14h none 40 bytes
Complete Working Buff Dump. This data will go into working buffer and not affect memory locations.
15h none none
"Complete" Working Buff Dump Request. On receipt, the MSB Plus Rev 2 will respond with a Complete Working Buffer Dump.

Transmitted Action Codes:

Command Extension Data

Oh none 1024 bytes
This is sent in response to an "Old" Full Memory Dump Req.

2 Pgm # 16 bytes
This is sent in response to an "Old" One Pgm Dump Request.

4 none 16 bytes
This is sent in response to an "Old" Working Buffer request.

10h none 2560 bytes
This is sent in response to a "Complete" Full Memory Dump Request. This may also be sent by depressing "1" and "8", then "Write" on the front panel.

12h Pgm # 40 bytes
This is sent in response to a "Complete" One Pgm Dump Request.

14h none 40 bytes
This is sent in response to a "Complete" Working Buffer Request.

REMEMBER THAT ALL COMMANDS END IN A OF7h!

"Old" Data Format (Pre-REV 2)

Each Program consists of 16 bytes of data. The high order bit is always 0. If a Full Memory Dump is taking place, the 64 programs are sent one after the other, starting on Program 11 (also known as MIDI Pgm 1).

Byte 1	PROC A Filters	(see below)
Byte 2	PROC A Port/Chan Bump	"
Byte 3	PROC A Transpose	“
Byte 4	PROC B Filters	“
Byte 5	PROC B Port/Chan Bump	“
Byte 6	PROC C Transpose	“
Byte 7	Remote Port/Channel	“
Byte 8	Output 1 Input Assignment	

(In all Output Port assignments, 0 = Input 1, 1 = Input 2, and so forth up to 7 = Input 8. 8= Output of PROC A, 9 = Output of PROC B, OAh = Output of Merge, and OBh = Output Disabled.)

…

Byte 15	Output 8 Input Assignment
Byte 16	SysEx Output Port assignment

(0 = Output l, 1 = Output 2, etc. ...7 = Output 8)

In "Complete" data dumps, the following additional data is sent/received, which represents the program command data

Byte 17	Port 1 Program number, Least Significant 4 bits.
Byte 18	Port 1 Program number, Most Significant 4 bits.

When no program change is to be sent, these two nibbles will both be OFh.

Byte 19	Port 1 Program Channel Number.
Byte 20	Port 2 Program number, Least Significant 4 bits.
…	
Byte 40	Port 8 Program Channel number.

PROC A and B Filter Bits:

Bit 0	Note Commands
Bit 1	Bend Commands
Bit 2	Controller Commands

(In air cases, a "1" filters the data out)

Bit 3	After-Touch Commands
Bit 4	Program Change Commands
Bit 5	SysEx and System Common Commands
Bit 6	Real Time Commands

PROC A and B PORT/CHAN BUMP Bits:
Bits 0-3 Channel Bump
Always this number is added to incoming channel number. Unsigned, even though the MSB Plus Rev 2 displays as a signed value.

Bits 4-6 Processor Input Port Assignment. 0 = Input Port l, 1 = Input Port 2, and forth.

PROC A and B Transpose Bits:
Bits 0-5 Number of Semi-tones to transpose by.

Bit 6 Direction of Transposition.
A "1" indicates a transposition down. These 7 bits are NOT in two- complement.

REMOTE Port/Chan Assignments:
Bits 0-3 Remote Channel assignment
0 = Channel 1, 1 = Channel 2, and so forth.

Bits 4-6 Remote Input Port assignment:
0 = Input Port l, 1 = Input Port 2, and so forth.

SYSEX Output Port Assignment:
Bits 0-2 Output Port Assignment, 0 = Output Port 1, 1 = Output Port 2, and so forth.

Be warned that a SysEx dump will "take over" the requested Output Port, and will disrupt anything that might have been coming thru that port!

CHAPTER 14 SPECIFICATIONS

Inputs:	8 MIDI, 1 Foot switch
Outputs:	8 MIDI
Memory:	64 Program Locations, Octal 11-88. Lithium Battery-backed RAM.
Modes:	(Write/ Recall) Program (Enter) Assignment PROC A Filters/Port PROC B Filters/Port PROC A Transpose/Bump (Page 2) PROC B Transpose/Bump (Page 2) Program Change Commands (Page 2)
Power:	Wall Transformer Supply, 9 VDC @ 500 mA, Center Positive
Dimensions:	19” by 1 ¾” by 4 ½”
Weight:	Approximately 2 lbs, Shipping weight 61bs.

CHAPTER 15 SERVICE

If you experience any operational difficulties, let us first reassure you that every unit is 100% factory tested.

One very common cause of problems is a "noisy" AC outlet. We recommend the use of an AC line filter with all computer related equipment. These can be purchased at any computer or hardware store, and many music stores also now carry them.

If the MSB has received a surge, the memory may get scrambled. Try to re-initialize the MSB using the power-on memory clear function.

To Initialize, Hold buttons 1 and 8 during power-up. Remember, this wipes out ALL memory!

If the symptom is one or more bright numbers lit up at random, like a single "8” or "C" in the display, then the MSB Plus has stopped functioning. Re-initialization will *not* help in such cases. The problem may be a bad wall transformer unit, or something more serious like a failed component.

There are no "user-serviceable" parts inside the MSB Plus Rev 2. For warranty service information, please refer to the really fine print at the end of this manual.

CHAPTER 16 TABLES

Display vs. MIDI Program Change number:

Display	MIDI #	Display	MIDI #	Display	MIDI #	Display	MIDI #
11	1	31	17	51	33	71	49
12	2	32	18	52	34	72	50
13	3	33	19	53	35	73	51
14	4	34	20	54	36	74	52
15	5	35	21	55	37	75	53
16	6	36	22	56	38	76	54
17	7	37	23	57	39	77	55
18	8	38	24	58	40	78	56
21	9	41	25	61	41	81	57
22	10	42	26	62	42	82	58
23	11	43	27	63	43	83	59
24	12	44	28	64	44	84	60
25	13	45	29	65	45	85	61
26	14	46	30	66	46	86	62
27	15	47	31	67	47	87	63
28	16	48	32	68	48	88	64

Remote MIDI Channel Assign Display

Display	Actual	Display	Actual
1	1	9	9
2	2	A	10
3	3	b	11
4	4	C	12
5	5	d	13
6	6	E	14
7	7	F	15
8	8	0	16

CHAPTER 17 ADVANCED USER QUICK OPERATION REFERENCE

These instructions are for the advanced MIDI user who may already have had some exposure to working with MIDI patch bays. This is not intended to be comprehensive, it is hoped that the advanced user will still find the opportunity to at least skim through the entire owners manual for specific examples and applications.

First Connect MIDI Cables - (Chapter 4)

Hook up the outputs of up to 8 MIDI devices to the 8 inputs of the MSB Plus. Connect the 8 outputs of the MSB Plus to up to 8 MIDI inputs. Use corresponding input and output numbers for convenience. For example, hook up a keyboard synthesizer using two MIDI cables to MSB Plus MIDI In and Out number 1. Hookup a computer or sequencer with two MIDI cables, to MIDI In and Out number 2 on the MSB Plus.

Likewise, use two MIDI cables for every instrument that will send and receive MIDI. This includes sequencers, keyboard synthesizers, samplers, drum machines, etc.

A keyboard or guitar controller will usually only need one cable, from its MIDI output to an MSB Plus MIDI input.

A signal processor or tone module may only need one cable, from the Output of the MSB Plus to its MIDI Input. The exception to this is when using a computer librarian or data storage device; in which case, use two cables so that both the input and the output are connected.

Routing Assignments: Basic Operation – (Chapter 4)
Tap mode once till the Assign LED is lit. The LEDs are input numbers, buttons are output numbers. Pressing a button increments the number above the button.

For example, pressing button #1 selects which of the eight inputs of the MSB Plus is routed to Output 1 of the MSB Plus. For example, a display of **33333333** means that whatever is connected to MIDI input number 3 on the MSB Plus, is sending MIDI data to MSB Plus outputs 1 through 8.

Since we have connected our instruments using corresponding input and output numbers, a better example is **33033333.** We have put a 0 above button 3. We do not want to route an instrument back into itself.

If the keyboard is connected to MSB Plus input and output 1, and a computer or sequencer is connected to MSB Plus input and output 2, this setting will allow the computer to record the keyboard, while the same keyboard to controls the rest of the system: **21111111**

To play back the sequencer, use: **21222222.** This type of assignment requires different routing to record and play back with.

A more sophisticated approach is to use the same assignment for both recording and play back, use: **21222222** and turn sequencer's "Echo-Thru " function ON, and the controlling keyboard's Local Control OFF. This would allow you to take advantage of an auto-channelizing function on your sequencer.

A third approach is to turn "Echo- Thru" off, and use the MSB merger (see below). The Assignment mode display is: **21CCCCCC**

Processors A and B – (Chapter 7)
Tap MODE button to select Proc A or Proc B. The number above button 1 is the MSB input that will be processed. For example, **4PPPPPPP** means that the Processor is assigned to input 4. After assigning input and selecting desired functions, you must assign the Processor to an Output in the Assign mode. For example, **OOOOAAA4** means that outputs 1 through 4 are turned off, that is the instruments hooked up to those outputs are not receiving any MIDI data. Outputs 5,6, and 7 are receiving processed data, and output 8 is receiving unprocessed data, directly from input 4.

Filters are set to either P (Pass) or F (Filter).

To enter Transposition or Channel Bump, use Pg 2 PROC A or PROC B Modes (That is, both the red and green mode LEDs are lit.)

Merging – (Chapter 8)
First use both processors to select which two inputs will be merged, for example, to merge inputs 4 and 7, select Proc A and set **4PPPPPPP**, then select PROC B and set **7PPPPPPP**. Then go to Assignment mode, and select "C" to merge. For example, **CCCOCCOC** means that the merged signal is sent to all outputs except 4 and 7.

Writing to Memory – (Chapter 4)
Hold the WRITE button, and while holding WRITE, tap a numbered button, then tap another numbered button. For example to write an assignment into program number 21, hold in the WRITE button, and while holding in the WRITE button, tap #2, and then tap #1. Release WRITE.

Recalling from Memory - (Chapter 4)

If not in PGM MODE, tap WRITE once to get to PGM MODE (PGM LED on). Tap two numbered buttons. For example, to recall program 32, first make sure that PGM LED is on. Then tap button #3 and then button #2.

MSB Response to Program Changes – (Chapter 5)

To prevent the MSB Plus from receiving program change commands, see page 15.

To change the input and channel that the MSB Plus will respond to program changes on, and to select which output the MSB will dump its system exclusive data out of, use Pg 2 PROC A or PROC B Modes (That is, both the red and green mode LEDs are lit.) These assignments are displayed in the right 3 digits.

Program Change Mapping – (Chapter 6)

Enter up to eight patch changes by going to Pg 2 (green LED) and entering output port, program number, and channel.

Panic – (Chapter 9)

To activate Panic Button, use front panel Panic Button or your own foot switch. A "normally open" or "normally closed" momentary foot switch can be used. Plug it in BEFORE power up so that the MSB Plus Rev 2 may automatically test it to see which kind it is.

Foot switch – (Chapter 10)

The foot switch may be assigned to either P ANIC or Program Advance by pressing mode and engaging switch.

SysEx – (Chapter 13)

To initiate a bulk data dump, first hold down buttons 1 and 8 and while holding both 1 and 8 press WRITE.

JLCooper Electronics Limited Factory Warranty

JLCooper Electronics ("JLCooper") warrants this product to be free of defects in materials or workmanship for a period of 12 months from the date of purchase.

This warranty is non-transferable and the benefits apply to the original owner. Proof of purchase in the form of an itemized sales receipt is required for warranty coverage.

To receive service under this warranty, customers in the United States should contact the JLCooper factory at (310) 322-9990 and talk to a service technician. If necessary, a Return Authorization number may be issued.

For our customers outside the United States, it is recommended that you first contact your Dealer or Distributor, since they may offer their own service or support policy.

If local support is not obtainable, please send a FAX to JLCooper's Service Department at (310) 335-0110, with a detailed description of the service required.

Upon issuance of return authorization, the product should be properly packed and shipped to Service Department, JLCooper Electronics, 142 Arena St., El Segundo, CA 90245.

Please include the following: copy of the sales receipt, your name and address (no P.O. Boxes, please), a brief description of the problem, and any other related items discussed with the service department and considered necessary to evaluate the product or effect a repair. The return authorization number must be clearly written on the outside of the package.

JLCooper will, without charge for parts or labor, either repair or replace the defective part(s). Shipping costs are not covered by this warranty.

JLCooper's normal repair turn around time at the factory is approximately 15 business days, from receipt of product to shipping. Your actual turn around time will include return shipping.

Actual turn around time will vary depending upon many factors including the repeatability of the customer's reported complaint, the availability of parts required for repair, the availability of related product needed to evaluate the product if necessary.
Priority services are available. These should be discussed with the service technician at the time the return authorization is issued.

This warranty provides only the benefits specified and does not cover defects or repairs needed as result of acts beyond the control of JLCooper including but not limited to: abuse, damage by accident/negligence, modification, alteration, improper use, unauthorized servicing, tampering, or failure to operate in accordance with the procedures outlined in the owner's manual; nor for acts of God such as flooding, lightning, tornadoes, etc.

THE DURATION OF ANY OTHER WARRANTIES, WHETHER IMPLIED OR EXI'RESS, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTY OF MERCHANTABILITY, IS LIMITED TO THE DURATION OF THE EXPRESS WARRANTY HEREIN. JLCOOPER HEREBY EXCLUDES INCIDENTAL AND CONSEQUENTIAL DAMAGES, INCLUDING BUT NOT LIMITED TO: LOSS OF TIME, INCONVENIENCE, DELAY IN PERFORMANCE OF THIS WARRANTY, THE LOSS OF USE OF THE PRODUCT OR COMMERCIAL LOSS, AND FOR BREACH OF ANY EXPRESS OR IMPLIED WARRANTY OF MERCHANTABILITY, APPLICABLE TO THIS PRODUCT. JLCOOPER SHALL NOT BE LIABLE FOR DAMAGES OR LOSS RESULTING FROM THE NEGLIGENT OR INTENTIONAL ACTS OF THE SHIPPER OR HIS CONTRACT AFFILIATES. THE CUSTOMER SHOULD CONTACT THE SHIPPER FOR PROPER CLAIMS PROCEDURES IN THE EVENT OF DAMAGE OR LOSS RESULTING FROM SHIPMENT.