MX-8

MIDI PATCHBAY / PROCESSOR

OWNER'S MANUAL

TABLE OF CONTENTS

INTRODUCTION

Congratulations on your purchase of the MX-8 MIDI Patchbay / Processor. As you become more familiar with the MX-8 you will find it an essential part of your MIDI setup. This manual will assist you in getting started, and will provide a detailed explanation of every function to help you get the most out of your MX-8.

Please take a few moments to fill in the enclosed warranty card and return it to Digital Music. By registering now, you enable us to send you information about the MX-8 and related products (such as new software releases). You will also automatically be registered for future updates to the unit itself. Remember, your returned warranty card is the *only* way we can notify you of new features as they are added.

INSTALLATION

As you unpack your MX-8, your carton should contain this manual, the unit itself, and a power adaptor. Save all packaging materials in case it should ever be necessary for you to ship the unit.

The MX-8 is designed to occupy one rack space in a standard 19" equipment rack. Install it as you would any other piece of rack mount gear.

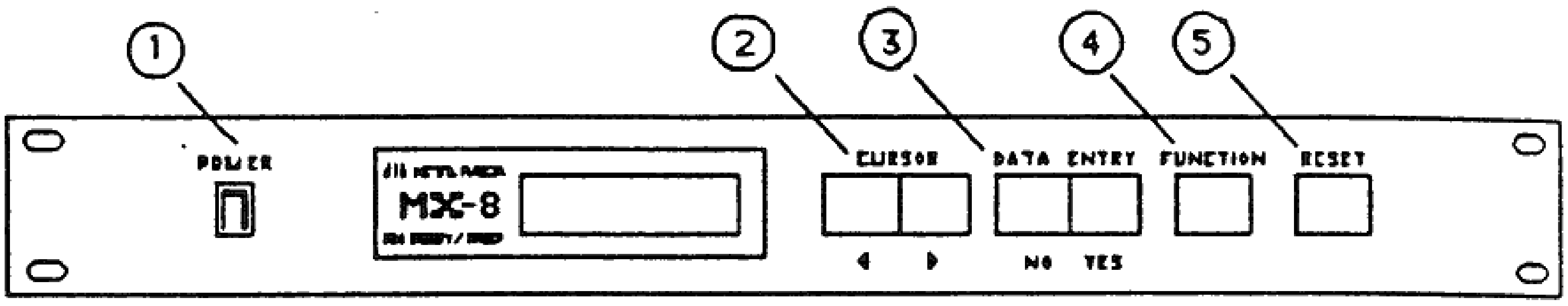
Now you are ready to connect your other MIDI equipment. Simply connect the MIDI OUT from each instrument to one of the six INPUT jacks on the MX-8. Next, connect the MIDI IN from each instrument or device to an OUTPUT on the MX-8. It will simplify matters if you number each synthesizer or tone module and connect it to correspondingly numbered INS and OUTS.

If you have a large setup, you will need more outputs than inputs, and the MX-8 provides them. Use outputs 7 and 8 to connect additional tone generators or MIDI controlled effects devices like reverbs and delays.

After you have installed the MX-8 and made all your MIDI connections, insert the power jack into the MX-8 rear panel receptacle marked "PWR" and plug the power adaptor into a suitable 117v outlet. It is recommended that all connections be made with the power off.

Now that everything is hooked up, let's take a look at the front panel...

FRONT PANEL



(1) POWER — turns the unit on and off.

(2) CURSOR keys — move the cursor (the underscore or flashing rectangle) in the display window from field to field (or choice to choice). For example, you would move the cursor over OUT 2 to change the input assigned to that output. Also change the Current Setup if no edits have been made.

(3) DATA ENTRY keys — allow you to increase or decrease values for fields selected by the cursor, answer yes or no, and enter alpha-numeric characters to name your setups. In short, these are the only keys that change the way the MX-8 works.

(4) FUNCTION key — selects which operation will be edited or reviewed. The MX-8 has a screen for each operation, and the function key scrolls through these screens.

(5) RESET key — sends the All-Notes-Off message to every output on every channel. If your synthesizer continues to play when it should be quiet, you'll be glad this key is here. You can also use the RESET function to reset the pitch bend controller which may have been set to an undesirable value if playback was stopped from a sequencer while the controller was engaged.
Because not every synthesizer responds to the All-Notes-Off message, it is still possible to get a "stuck note" with some synthesizers if you change parameters that alter the note value (transpose) or channel assignment (Velocity Cross Switch, Map, Channel Offset, etc.) *while data is being transmitted.* If you hold the **RESET** button down, after sending the All-Notes-Off message the MX-8 will also send a Note-Off to every note on every channel.

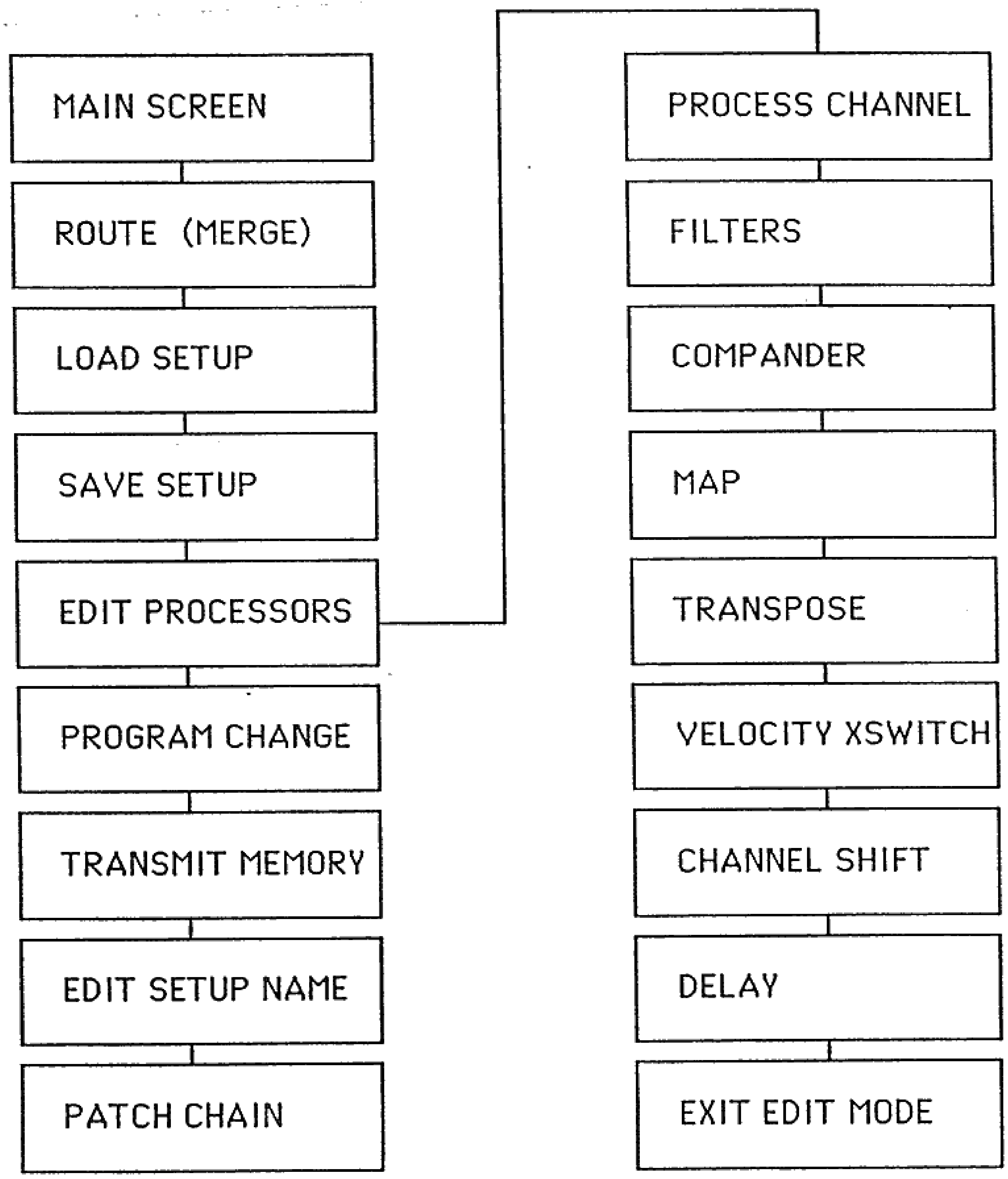
(6) FOOTSWITCH jack (not shown) — insertion jack for optional footswitch, which can be used to step through the setups in either direction.

USING THE MX-8

Once you have made all the cabling connections and turned on the MX-8, you are ready to start using it. Each function has it's own "screen" which contains all the information relevant to that function, and one or more parameters that you can change to suit your needs. You move from screen to screen by depressing the **function** key, then choose which parameter to alter by depressing the **cursor** keys. Finally, you alter the parameter you have selected with the **data entry** keys. On most screens, if you hold a key down it will auto-repeat (scroll rapidly).

FUNCTIONS

The following is an illustration of the MX-8's function screens.



The Function Screens

THE MAIN SCREEN

```
CURRENT SETUP
#3 MY SONG
```

The MAIN screen tells you which setup is currently active. Each setup has a number (1-50) and an alphanumeric name. Each time you load a setup from either the front panel or through a MIDI program change this screen is displayed. You can change the Current Setup by pressing the **cursor** or **data entry** keys unless you have entered Edit Mode. When you press the **function** key, you enter an Edit Mode which lets you alter the parameters of the active setup.

ROUTE / MERGE

```
IN  21M2A131 1>A
OUT 12345678 2>B
```

The ROUTE screen defines the patchbay function. This is the screen where you control your MIDI signal flow. You have connected your instruments to the INPUT and OUTPUT jacks on the rear panel; now you can re-arrange your connections without having to plug or unplug MIDI cables. In addition to the instruments you have connected, there are two processors (A and B), and a signal (M) which contains the merged data from A and B which can be assigned to any output.

Position the **cursor** over an output (1 through 8) then use the **data entry** keys to select the input (1 through 6, A, B, or M) you want routed to that output. Keep repeating this until you have assigned the desired input to every output you've connected. You may find it useful to think of the top line as the *source* from which your MIDI data comes, and the bottom line as the *destination* to which it will be sent.

Please note that when you assign an input to a processor you can still route the original input signal to any of the outputs. In the above screen for example, input 1 is routed directly to outputs 2, 6, and 8. Input 1 processed by processor A is routed to output 5. If you wish to send processed data to your synthesizers you *must* assign a processor to an output.

MERGE — If you wish to merge two inputs, use the **cursor** keys to position the cursor next to A or B and choose the inputs to be processed with the **data entry** keys.

When you select "M" as a source for one of the 8 outputs, the data going to processors A and B is combined and routed to the selected output(s). Note that the output of processor A is still individually available, but processor B, which outputs the merged messages, can not be individually assigned if merging is occurring. In the above screen, inputs 1 and 2 are being merged and sent to output 3.

If both inputs being merged contain timing information (e.g. MIDI clocks), as would be the case if you merge two sequencers or a sequencer and a drum machine, you must filter System Real Time messages from one or both of the inputs to avoid sending conflicting timing messages to the outputs receiving the merged signal. (Filtering and other processor functions to be performed on the designated inputs are selected from different screens — see EDIT PROCESSOR).

In the screen shown above, input 1 is being processed by processor A, input 2 is being processed by processor B. Input 2 is routed to output 1, input 1 is routed to output 2, the merged signal of both processors is routed to output 3, input 2 is routed to output 4 and so on.

Example: Let's say you own a DX-7, an Atari 1040 ST computer, and an MT-32 multi-timbral tone generator. Connect the outputs of each unit to the inputs of the MX-8, with the DX-7 connected to input 1, the Atari to input 2, and the MT-32 to input 3. Now connect MX-8 outputs 1, 2, and 3 to the inputs of the units in the same fashion. To route the DX-7 to the Atari (to record a sequence, for example) you would arrange the ROUTE screen to send IN 1 to OUT 2 (the *source* is the DX-7, the *destination* is the Atari). You do this by positioning the cursor over OUT 2 and using the **data entry** keys until a 1 appears over OUT 2). To hear your sequence played back through the DX-7, you would want to place a 2 above OUT 1 (the *source* is the Atari, the *destination* is the DX-7). If you wanted to play the MT-32 with both the DX-7 *and* the Atari at the same time you would route input 1 to Processor **A** and input 2 to Processor **B** by moving the cursor next to the desired processor with the **cursor** keys and then pressing the **data entry** keys until the screen shows a '1' next to **A** and a '2' next to **B**. Then, positioning the cursor over OUT 3, use the **data entry** keys until an **M** (for MERGE) appears on the display (the *source* is the combined signal from the Atari and the DX-7, the *destination* is the MT-32) .

That's really all there is to it! You can change setups anytime (to send data dumps to your computer, play one synthesizer with the keyboard of another, or make truly monster chords with every synthesizer you have), without ever plugging or unplugging a cable.

You may find that there are some setups that you use all the time, or for particular songs. To save the time and effort of "re-inventing" frequently used setups, you will want to go to the SAVE screen. As always, move to a different screen by pressing the **function** key.

LOADING SETUPS

```
LOAD setup from
#15 GET GOING
```

The LOAD function lets you quickly recall any of the 50 setups you can save to the MX-8's memory. Use the **cursor** keys to scroll through the setups, and when you see the one you want, press the **data entry** key (YES/+1) to load the setup. The MX-8 will ask "are you sure?" because loading a setup will replace the current setup, and it will be lost unless you save it. If you wish to continue, press **data entry** (YES/+1) again. After you load a setup, the display returns to the MAIN screen.

In the example, setup #15 is selected to be loaded. The display also shows the name of the setup ("GET GOING" in this case,) which can be changed to any name you like (see EDIT NAME).

You can also recall a setup through MIDI program changes. The setup numbers correspond to program numbers 1 to 50 on most synthesizers (See PROGRAM CHANGE).

Loading a setup will load *all* of the settings that were stored when the setup was SAVED, which can include a complete chain of eight additional program changes. See PATCH CHAIN.

Loading a setup will load *all* of the settings that were stored when the setup was SAVED, which can include a complete chain of eight additional program changes (see PATCH CHAIN).

SAVING SETUPS

```
SAVE current to
#10 MY SETUP
```

The SAVE function allows you to save your setups for quick recall. When you change your setup you are actually changing a "51st" program, which resides in a special memory location called a buffer. This setup will remain intact even if you turn off the power, but will be lost if you load a different setup unless you SAVE it first.

Use the **cursor** keys to scroll through the 50 programs. When you find one you wish to use for the current setup, press **data entry** (YES/+1). The MX-8 will ask "are you sure?" because saving a setup will over-write any existing setup of the same number. If you wish to continue, press **data entry** (YES/+1) again. After you save a setup you will be returned to the ROUTE screen.

When you save a setup, you are saving all of the routing assignments, plus all of the other functions you have defined from other screens.

EDIT PROCESSORS

```
EDIT (Y/N)?
PROCESSOR-A
```

Use the **cursor** keys to change between processors A and B, then press the **data entry** (YES/+1) to select a processor for editing.

When you select a processor to EDIT, you enter a second level of screens, each of which lets you define a processor function. Once again, the **function** key moves you from screen to screen, the **cursor** keys move from choice to choice, and the **data entry** keys change the way the MX-8 works.
Remember, when you change your setup you are actually changing a "51st" program, which resides in a special memory location called a buffer. This setup will remain intact even if you turn off the power, but will be lost if you LOAD a different setup unless you SAVE it first.

Please note that when you assign an input to a processor you can still route the original input signal to any of the outputs. If you wish to send processed data to your synthesizers you *must* assign a processor to an output.

THE EDIT SCREENS

Selecting the Process Channel

```
PROCESS CH=1
```

You may choose to process any channel (1 to 16) or all channels (OMNI). Use the **data entry** keys to change the processed channel.

Filters

```
FILTER STAT:OFF
AFTERTOUCH
```

Use the **cursor** keys to move from choice to choice (each choice will have its own display) and enable or disable that particular selection with the **data entry** keys. The choices for filtering are: Note On/Off, Aftertouch, Control Change, Program Change, Pitch Bend, System Common/Exclusive, and System Real Time messages. In the example, the status of the Aftertouch filter is "off." This means that Aftertouch messages will **not** be filtered. System Common/Exclusive messages are filtered on all channels regardless of the process channel setting. Please note that when you assign an input to a processor you can still route the original input signal to any of the outputs. If you wish to send processed data to your synthesizers you *must* assign a processor to an output.

Compander

```
COMPANDER      ON
1:2.00 LEVEL= 55
```

COMPANDER is a combination compressor/expander for the *velocity* of notes being processed. Choose the field to be edited with the **cursor** keys; press the **data entry** keys to turn the function on and off or change the values. When the compander is on, you can select a *ratio* which indicates the amount of expansion or compression (1:1 — pronounced 'one to one' — is unity — i.e. no change; 1:1.25 through 1:3.00 are expansion settings; 1:0.75 through 1:0.25 are compression settings).

When you select a compression setting (1:0.75; 1:0.50; 1:0.25), the data processed will have a narrowed dynamic range. The highest velocity value and the lowest velocity value played will be closer to each other than the original unprocessed signals. This is useful for producing 'tight' bass lines or drum parts where the original parts may be played unevenly. The LEVEL value adjusts the overall velocity level.

When you select an expansion setting (1:1.25 through 1:3.00), the data processed will have a broader dynamic range. Playing a note hard (relative to the center point set with the Level control) will result in a note with a velocity value greater than the original. Notes played softly will have even lower velocities than the original. This expanded dynamic range will enable you to make parts more dramatic, or create more realistic drum rolls. Take full advantage of the velocity sensitivity of your tone modules even if your controller (or playing style) can't put out velocities at the extremes of the spectrum.

Please note that when you assign an input to a processor you can still route the original input signal to any of the outputs. If you wish to send processed data to your synthesizers you *must* assign a processor to an output.

Mapping

```
MAP RANGE  #1  OFF
C -2 -> C +1  CH12
```

Define 4 ranges, each with its own MIDI channel. Use the **cursor** keys to move from field to field, and press **data entry** to change the range being edited, whether mapping is on or off, the upper and lower note values for the range being edited, and the channel assignment for the selected range.

The lowest split point is "C-2" the highest is "G+8". When you select a high split point, it defines the low split point for the next range. In the example, range #1 has a low split point of C-2 and a high split point of C+1. Notes in this range from the processed input will be output on channel 12 and the low split point for range #2 will automatically be C#+1.

Since you may not always want to use four zones, you can set range #4 to "none" (i.e. no notes in this range) by raising the lower split point value to G+8 and then incrementing once more with the **data entry** key. If range #4 is set to "none" and you want only two zones to be mapped, range #3 can be similarly set to "none."

Please note that when you assign an input to a processor you can still route the original input signal to any of the outputs. If you wish to send processed data to your synthesizers you *must* assign a processor to an output.

Mapping does *not* effect the channel of program changes or other types of MIDI data.

Transpose

```
TRANSPOSE
+12 semitones
```

Use the **data entry** keys to change the number of semitones to transpose. You can transpose up 63 or down 64 semitones. The transpose function is off when the value is 0.

In the example, the processed notes are being transposed up one octave (12 semitones).

Please note that when you assign an input to a processor you can still route the original input signal to any of the outputs. If you wish to send processed data to your synthesizers you *must* assign a processor to an output.

Velocity Cross Switch

```
VEL XSWITCH ON
THRESH=85  CH1
```

The Velocity Cross Switch function lets you specify a threshold for key-on velocity which, when reached, selects an alternate MIDI channel. Use the **cursor** keys to move from field to field, (choices are: on / off, threshold level, and alternate channel number) and press **data entry** to change the values. In the example, this function is enabled, and all processed notes with a velocity of 85 or more will be output on channel 1. Please note that when you assign an input to a processor you can still route the original input signal to any of the outputs. If you wish to send processed data to your synthesizers you *must* assign a processor to an output.

Channel Shift

```
CHANNL SHIFT ON
Offset  by  1
```

The Channel Shift function has two modes: **Offset** and **Reassign.** In **Offset** mode each channel being processed is increased by the offset value (0 to 15). Values will "wrap around" so that if you offset by 1, messages on channel 16 will be offset to channel 1. In **Reassign** mode, data from a selected channel is redirected to a different channel. Use the **cursor** keys to move from field to field, and press **data entry** to change the values. In the example, the function is on, and all processed channels are being offset by 1 (channel 1 data is output on channel 2, etc.)

With regard to MAP, VELOCITY CROSS SWITCH, and CHANNEL SHIFT functions, be aware that when you change the channel assignment for any MIDI events, the remaining processor functions will operate only on the originally specified process channel unless OMNI mode has been selected. That is, if you're processing channel 1, and MAP a portion of your keyboard to channel 2, the channel 2 messages will not be transposed.

Please note that when you assign an input to a processor you can still route the original input signal to any of the outputs. If you wish to send processed data to your synthesizers you *must* assign a processor to an output.

Digital Delay

```
DLY 3x V-15 CH12
NOTE  4/16T @120
```

The Delay screen may seem a bit cryptic at first, but it is very easy to use. **"DLY"** tells you that you are editing the Delay function. The next field (**3x** in the example) shows the number of repeats. To turn the Delay function off, set the number of repeats to "0". The **"V-15"** stands for velocity, and indicates the amount the velocity of each repeat will be offset (-64 to +63). In the example, each repeat's velocity is decremented by 15 giving you a decay effect with each successive repeat. If it read **"V+15"** each repeat would crescendo. **"CH"** is the channel on which the repeats will be output (when **CH** is

If you position the cursor on the NOTE field, you can toggle (using the **data entry** keys) between two modes for setting the delay time: **TIME** in milliseconds, and NOTE value/tempo. Values for time are from 5 to 3,000 milliseconds in 1 mS intervals.

To program a delay using the Note Value and Tempo mode, choose the number of 16th notes, select triplet or whole unit time, and specify the tempo in quarter notes per minute (the range is from 30 to 250 bpm). In the example, the delay time is set for quarter note triplets at a tempo of 120 beats per minute (4 sixteenth notes = 1 quarter note).

When you use the Delay function you are actually adding new data to the data stream. This data is stored in the "delay buffer" for output after the specified delay time has elapsed. It is possible to overflow this buffer by sending lots of data to the processor while setting long delay times with many repeats. To help avoid an overflow, the MX-8 uses **dynamic filtering**. Aftertouch messages are not delayed once the delay buffer is approximately half full. Still, it is possible to overflow the delay buffer (echoing all channels from a sequencer that's outputting a lot of pitch wheel data, for example). If this happens, the MX-8 will let you know that you must change something (set filters, process only one channel, set fewer repeats or shorter delay times) by displaying "delay overflow" on the screen. This message is accompanied by an All Notes Off message so that you won't have any stuck notes even if you do manage to get an overflow.

If you want to echo back to the synthesizer from which you are playing, we recommend that you set the synthesizer to receive on a different channel than it sends on, then assign the repeats to the receive channel. By doing this you avoid looping data back to the synthesizer it comes from. If you do loop data, there is no serious consequence (it might be just what you want to do if you're transposing to play one finger intervals); however, since each note you play will, in effect, be played twice, the number of separate voices your synthesizer can play will seem to be cut in half.

Please note that when you assign an input to a processor you can still route the original input signal to any of the outputs. If you wish to send processed data to your synthesizers you *must* assign a processor to an output.

Exit Edit Mode

```
EXIT  (Y/N)?
EDIT MODE
```

Press **data entry** (YES /+1) to return to the main level ROUTE screen.

Please note that when you assign an input to a processor you can still route the original input signal to any of the outputs. If you wish to send processed data to your synthesizers you *must* assign a processor to an output.

PROGRAM CHANGE

```
PGM CHG -> B  ON
MIDI channel =1
```

The PROGRAM CHANGE function specifies the channel (1-16) on which the MX-8 will receive MIDI program change commands. Either processor may be selected to receive program change commands. When this function is ON, it is enabled for every setup. That is, this is a global function of the MX-8. Use the **cursor** keys to move from field to field, and press **data entry** to turn the function on or off, change the channel, or toggle between the two processors. The channel selected is also the channel the MX-8 uses for system exclusive functions (sending and receiving memory dumps).

This function lets you change which keyboard (or sequencer) selects setups for the MX-8, and the ability to shift from one instrument to another and instantly re-configure your equipment to respond to commands from a different source is a feature you will use again and again. There is, however, an aspect of the MIDI 1.0 specification which can cause a problem in certain (rare) circumstances...

In an effort to reduce redundant data messages, the creators of the MIDI specification included what is called "running status." Synthesizers which implement running status don't send a status byte (to identify the source of the message that follows) with every message. They will send a status byte with the first message of its kind (Note On, for example) and then omit status messages until a different type of message is sent (like a program change). As a result, if you send several consecutive program changes from *different sources* , the MX-8 *may* not respond to all of the program changes if the status byte identifying the new controller's program change message is "missing" (due to running status). Sending any other kind of message will generate a new status byte, and subsequent program changes will be recognized by the MX-8. It is unlikely that you will ever encounter this situation.

SYSTEM EXCLUSIVE

```
XMIT MEMORY?
SYSEX OUTPUT #2
```

The XMIT MEMORY function will allow you to transmit the whole memory of the MX-8 to the output port specified on the screen. Most software developers send a *request* for a MIDI data dump when performing librarian or editor functions. The necessary codes are discussed in the System Exclusive appendix to the manual. Use XMIT MEMORY for those situations where it is necessary to initiate a data dump from the front panel of the MX-8, (e.g. to load the setups from one MX-8 to another, or for use with a librarian that cannot send a request for data).

First define the output by moving the cursor (with the **cursor** keys) to the numerical field on the second line of the display (the # 2 in the screen depicted above). Then press **yes/+1** to send the data. The MX-8 will output its data to the specified output port on the channel defined in the Program Change function. (Remember that when sending the MX-8 a request for a memory dump from an external storage device, the request must routed to Processor A).

EDIT SETUP NAME

```
EDIT setup NAME
PGM: MY SETUP
```

Before you save a setup that you have just created or modified, you can give it a descriptive name. Use the **cursor** keys to move from letter to letter, then press the **data entry** keys to change the characters. Hold the keys down to scroll faster.

PATCH CHAIN

```
PATCH CHAIN #1
OUT2 CH1 PGM65
```

The **PATCH CHAIN** function sends up to 8 program changes to any output on any channel (1-16). For each of the 8 "links" in the chain, you use the **cursor** keys to move from field to field, then use the **data entry** keys to change the values. If you enter a "0" in the OUT field of a given link, that link of the chain will be disabled. In the example shown, the cursor is selecting "link #1", which will send program 65 on channel 1 to the synthesizer or tone module connected to output 2. The program change command will be transmitted whenever this program is called from the front panel or through MIDI. You can use the **data entry** keys to select "link #2" and define an output, channel and program number for the second link in the chain. Continue in this fashion until all eight links have been defined as needed. Remember that a setup must be saved to a permanent memory location before it can be recalled.

Example: Suppose you want to use your DX-7 (connected to input and output 1) as a controller and on a specific song you want to have the DX-7 play program number 25. Also, a slave tone generator, connected to output 3, to play program 15. Let's further suppose that the MX-8 setup that defines how the synthesizers are routed, as well as all the processing functions you want for this song are saved to setup number 8. First **LOAD** setup 8 into memory (see the section on the **LOAD** function for this procedure). Then configure the Patch Chain as follows: use the **function** key to get to the Patch Chain screen. Then use the **cursor** keys to position the cursor under the 'link' number (next to the # sign on the top line of the display). Use the **data entry** keys until your screen says "#1". Next, press the **cursor** keys to move to the 'output' field (the number next to the word **OUT** on the lower line of the display). Use the **data entry** keys to place a "1" on the screen. Move the cursor over to the **CH** field and enter the receive channel number for the DX-7. Lastly (for this link), move the cursor over to the **PGM** field and change the number there to 25. For the next "link" in the patch chain, move the cursor to the top line of the display and increase the number there to read "#2". Again, move the cursor to the **OUT** field, and enter "3" for output 3 (the tone module). Next, enter the receive channel of the tone module in the **CH** field. Finally, enter "15" in the **PGM** field to send program 15 to the tone module when this setup is loaded. (Different manufacturers use different numbering schemes for program numbers. It may help you to make a table of program change numbers for your individual synthesizers so you will know which program you will get when you send out program numbers from the MX-8).

Now that you've configured the example Patch Chain, you must save it to MX-8 setup number 8 using the **SAVE** function: use the function button to get to the **SAVE** screen. Then use the **yes/+1** key to save the modified setup.

From now on, whenever you **LOAD** setup #8 (either from the front panel or through MIDI) the Patch Chain will be output and your synthesizers will receive the program changes that you have stored with the setup.

When you first turn on the MX-8 the main screen shows the contents of the buffer as you last left it, and does not output the **PATCH CHAIN** unless you re-load the setup.

Appendix A

ADDITIONAL FEATURES

Dynamic Filtering — MIDI signals take 320 microseconds per byte to be transmitted. If more data than can be transmitted in that time is being sent to the processors (when merging two sequencers, for example) or data is being added to the data stream (with numerous short delays), data must be buffered (temporarily stored for later output). If the buffer gets full and the data glut continues, messages may be lost causing stuck notes, out of tune synthesizers and other nasty situations. To avoid this the MX-8 uses **dynamic filtering.** At the point where the buffered data will require 30 mS to be output, (remember — this delay is a by-product of MIDI, *not* the MX-8) the MX-8 automatically begins thinning out pitch bend and aftertouch messages to reduce the chance that an overflow condition will occur. Pitch bend messages that reset the pitch bend controller to it's center value are always passed through, so your synthesizer will never be left out of tune.

In the unlikely event that your data does overflow the buffer, the MX-8 will display a "MIDI DATA OVERFLOW" message on the screen (which will remain visible until you press a key or send a program change) and send an All-Notes-Off message to avoid stuck notes in case a Note-Off message was lost when the overflow occurred.

Appendix B

TROUBLESHOOTING

The MX-8 is laid out in a logical way, and you may never encounter any difficulties using it, but here are some areas to examine if you do have a problem:

My controller doesn't play my slaves...

Check to make sure that you have set the MIDI channels on the slaves to receive on the same channel that your controller is sending on.

If the controller is routed to a processor, and that processor is routed to the slaves, try routing the controller directly to the slaves without going through a processor. If that works, examine the processor settings to determine where the signals are being directed:

> Check to make sure that you are not mapping, channel shifting, re-assigning, or cross-switching channel assignment to a channel that is different from the receive channel of the slave units.
>
> Make sure the Note On/Off filter is not set to ON.
>
> If you are transposing, be sure that the slaves are able to reproduce notes in the note range you have selected.

Make sure your connections are going where you think they are going. It's possible to mistake a "thru" for an "out," or to think a synthesizer is plugged in to a different port than it is in fact plugged in to.

If you still haven't found the problem, plug the controller directly into the slave to make sure the synthesizers are functioning properly.

My drum machine plays wildly out of time...

If you are merging two sources that contain Real Time Clock information, you must filter that message from one (or both) sources, or your drum machine will receive two sets of timing data, with unpredictable (and usually undesireable) results. Decide which source you want for timing, and set the filter of the *other* processor to remove the unwanted clocks.

I keep getting "stuck" notes...

Some synthesizers (Casio CZ's for instance) don't respond to the All-Notes-Off command sent by the MX-8 every time a function parameter is altered. Therefore, if you change certain functions while data is being played, it is easy to "hang" one of these synthesizers. If you alter transposition, for example, while a note is being played, the Note Off command for that note will be transposed to a different value, and the note will play "forever." The same is true for channel changes.

Press (and hold) the Reset key. This will send All-Notes-Off commands followed by Note-Off commands for every note on every channel (this takes about 1.5 seconds). It is sometimes necessary to let Reset cycle more than one time to get every stuck note to stop playing.

The best solution for this problem is to know your equipment. If your synthesizers don't respond to the All-Notes-Off message, it is best not to change parameters while notes are being played.

I keep getting error messages...

MX-8 error messages (such as Delay overflow or MIDI DATA OVERFLOW) are indications that too much data is being introduced to the data stream. If this happens, you will experience unwanted delays or even lost data unless you reduce the data flow somehow.

Fix conditions that cause Delay overflow by setting filters to remove some data (Aftertouch is a notorious flow-stopper), process on one channel only, or set the delay function to fewer repeats and/or shorter delay times.

MIDI DATA OVERFLOW can be the result of merging very dense sources (like two sequencers).

MIDI is fast, but it does have limits, and only one message at a time can be transmitted. If there are more messages being generated than can be sent out, they must be temporarily stored. Eventually this storage area (the buffer) will fill up if the glut of data continues.

Either eliminate some of the data at the source (mute tracks from the sequencer) or filter it with the MX-8 processors. Adding numerous very short repeats with the MX-8 Delay function isn't going to help this condition, either.

Data error messages will crop up on some synthesizers if you turn the MX-8 on or off while connected to synthesizers that are turned on. This is harmless.

The MX-8 responds slowly or not at all...

Check to make sure that you have not inadvertantly connected a synthesizer to its own "thru" instead of its "out." This would cause a data loop that would be the MIDI equivalent of audio feedback, and can "lock out" the keys on the MX-8.

If you're using a computer, some MIDI interfaces (like the Roland MPU-401) have an echo function as a default setting, causing them to act like thru ports unless under the command of a program. Most programs disable this echo function when you load them (it can't be disabled by a MIDI command), but if you are not using a program, take care not to route the computer to itself, or you will create the "feedback loop" described above.

Sometimes the MX-8 ignores program changes...

The MX-8 requires a status byte to identify the source of a program change. Some synthesizers and sequencers use "running status" to reduce data redundancy. With running status, a status byte is sent once for a given message type (program change, for example) and then is omitted until a different type of message is sent. Therefore, if you send several consecutive program changes from different sources, the MX-8 may not respond to all of the program changes if the status byte identifying the current controller is missing (i.e. had been sent in a previous message and is now not included with running status messages).

Send a different kind of message (play a note, move a modulation wheel, etc.). This will generate a status byte for the new message and any subsequent program change.

Appendix C

SYSTEM EXCLUSIVE IMPLEMENTATION

MESSAGE FORMAT

F0	Start of system exclusive message.
00	
00	
07	Digital Music Corp. ID#.
0n	Where n=channel number
0t	Where t=message type
(*dd..dd*)	Data bytes.
(*cc*)	Checksum...7-bit sum of *dd..dd*, 1's complement.
F7	End of message (EOX).

Note: Do not include parameters in parenthesis if LENGTH=0.

MESSAGE TYPES

TYPE	LENGTH	DESCRIPTION
0	83	Transfer of active setup either from host to MX-8 or from MX-8 to host.
1	4150	Transfer of all 50 stored setups either from host to MX-8 or from MX-8 to host.
2	0	Request from host to MX-8 for the MX-8 to download its active setup.
3	0	Request from host to MX-8 for the MX-8 to download all 50 stored setups.

Example: To request a dump of all 50 stored setups with the MX-8 set to MIDI channel 1, send this message: **F0 00 00 07 00 03 F7.**

SETUP PARAMETER DETAIL

PARAMETER	LENGTH	DESCRIPTION
Setup Name	10	10 character name encoded (Δ!#&-'/0..9/A..Z)
Routing	10	Route assignments for each output (1..8), followed by
processors		A,B respectively. Values are 0..8 for outputs and 0..5 for processors, where 0..5 are inputs1..6; 6,7 are processors A,B; and 8 is merging.

Note: Do not select processor "B" and merging concurrently.

PARAMETER	LENGTH	DESCRIPTION
Reserved	1	Unused.
Chain Mask1	1	D0..D3 enable chain elements 1..4, respectively.
Chain Mask2	1	D0..D3 enable chain elements 5..8, respectively.
Chain Array	16	Patch chain elements (2 bytes each). Byte1, D0..D3=channel# (0..15), D4..D6=output# (0..7); byte2=pgm# (0..127).
Processor-A	22	See processor parameter list below.
Processor-B	22	" " " " " .

PROCESSOR PARAMETER DETAIL

PARAMETER	LENGTH	DESCRIPTION
Process Channel	1	D0..D3=channel#, D4=OMNI enable.

Processor Flags	1	D0=channel shift enable.
		D1=shift mode (0=offset, 1=reassign).
		D2=delay mode (0=milliseconds; 1=note, tempo).
		D3=mapping enable.
		D4=delay divisor (0=16ths, 1=16th triplets).
		D5=compander enable.
		D6=velocity cross switch enable.

Repeats	1	# of repeats for delay, 0..7 where 0=off.

Filters	1	D0=note on/off.
		D1=aftertouch.
		D2=control change.
		D3=program change.
		D4=pitch bend.
		D5=system exclusive/common.
		D6=system real time.
		Setting a bit turns on the respective filter.

Split Points	3	Split points between ranges 1&2, 2&3, and 3&4,
respectively.		
Split Channel	4	Output channels for split ranges 1 through 4, respectively.
Delay Time	2	TIME MODE: Byte1 contains D0..D6, byte2 contains D7..D11 of time. NOTE MODE: Byte1 contains D0..D6, byte2 (D0) contains D7 of tempo.
D1..D4 in		byte2 are note value multiplier.
Echo Channel	1	D0..D3=alternate echo channel. D4=enable.
Velocity Offset	1	Echo velocity offset in 7-bit 2's complement form (-64..+63).
Compander Ratio	1	Compression/Expansion ratio. Range is 1..12; where 1=1:0.25, 2=1:0.50, 3=1:0.75, ... ,12=1:3.00.
Compander Level	1	Relative velocity output level for compander. Range is 0..99; where 50 is normal.

PARAMETER	LENGTH	DESCRIPTION
Shift From/Offset	1	REASSIGN MODE: Channel to shift from (0..15). OFFSET MODE: Offset to add to channel (0..15).
Shift To	1	Channel to reassign to.
Transpose	1	Transpose value in 7-bit 2's complement form -64..+63 semitones.
Velocity Cross Pt.	1	Velocity cross switch crossover point.
Velocity Cross Ch.	1	Velocity cross switch alternate channel.