Synapse

MIDI Routing System

Owners Manual
Fourth Edition
July 1993

JLCooper Electronics

12500 Beatrice Street • Los Angeles, CA 90066 U.S.A.

Greetings

Thank you for purchasing the JLCooper Synapse, MIDI Patch Bay and Processor. The Synapse is a MIDI routing system engineered for the professional MIDI studio and the advanced MIDI stage artist.

This product represents years of research and development in MIDI design, from the company that first brought you MIDI accessories.

Because Synapse is such a powerful MIDI tool, take the time to familiarize yourself with this owners manual, especially before attempting to use the advanced data processing features.

Please fill out the enclosed warranty card soon and mail it in so that we can keep you informed of any software updates as they become available.

JLCooper also makes a Synapse editor/librarian software disk. This program allows you to edit and store all of Synapse's internal memory to a computer. Synapse software operates as a "desk accessory" so that it can be run along with another program such as a MIDI sequencer. Contact your JLCooper dealer for details.

Greetings

Thank you for purchasing the JLCooper Synapse, MIDI Patch Bay and Processor. The Synapse is a MIDI routing system engineered for the professional MIDI studio and the advanced MIDI stage artist.

This product represents years of research and development in MIDI design, from the company that first brought you MIDI accessories.

Because Synapse is such a powerful MIDI tool, take the time to familiarize yourself with this owners manual, especially before attempting to use the advanced data processing features.

Please fill out the enclosed warranty card soon and mail it in so that we can keep you informed of any software updates as they become available.

JLCooper also makes a Synapse editor/librarian software disk. This program allows you to edit and store all of Synapse's internal memory to a computer. Synapse software operates as a "desk accessory" so that it can be run along with another program such as a MIDI sequencer. Contact your JLCooper dealer for details.

TABLE OF CONTENTS

Synapse Features	5
Front Panel Display and Push Buttons	6
Modes of Operation	8
Hook Up	11
Assignment Mode	13
Basic MIDI Routing	13
Synapse Presets	16
Program Mode	17
Writing to Memory	17
Recalling from Memory	17
MIDI Data Processors	19
General Information	19
Features and Suggested Applications	21
The Processor Pages	23
Assigning the Processors to a MIDI Input	23
Assigning the Processors to a MIDI Output	24
Programming the Processors: Examples	25
Data Filter	25
Channel Filter	27
Channel Bump	28
Velocity Rescale	29
Zone Transpose	29
Zone Channel Bump	30
Keyboard Splits	31
MIDI Merging	35
Patch Map	39
Panic	41
Remote Operation	43
MIDI Time Code Display	45
Linking Several Synapses	47
Scope	49
Technical Information	51
System Exclusive	
Initialization	
Service	
Processor Flow Chart	52
Synapse Remote Editor/ Librarian Information	53

Synapse FEATURES

This is an overview of the features found in the JLCooper Synapse that will be covered in this manual.

- 16 MIDI inputs
- 20 MIDI outputs
- Large LED display
- Programmable MIDI Routing, Splitting, Merging and Processing

  Each of the three Processors include
  - Data Filtering
  - Channel Filtering
  - Channel Bumping
  - Velocity Rescale.
  - Four overlapping zones with programmable low note, high note, channel bump and transpose. A total of twelve zones available.
- Any of three Processors can be merged, allowing up to three-input Merging
- All routing and Processing assignments are stored in 64 Memory Locations backed by a Lithium Battery
- Patch Map allows up to 16 Program Change Commands to be sent on any combination of MIDI channels and MIDI output ports, 16 per each preset
- Remote Operation via MIDI program change and system exclusive.
- Scope, a diagnostic tool that displays the last incoming MIDI status byte.
- Panic function that eliminates stuck notes.
- MIDI Time Code Display
- Several Synapses can be linked together to create larger MIDI systems.

Synapse FRONT PANEL

All Synapse functions are controlled by the front panel buttons. From left to right, we have:

WRITE
This is used for writing (that is, storing) an assignment into memory.

Also, pushing the WRITE button at any time returns Synapse to the Program Mode.
(This is the mode for recalling assignments from memory.)

If WRITE and MODE are held down at the same time, the PANIC function is initiated.

MODE
This button is used for selecting the desired Synapse pages to enter, review, or alter all the parameter settings in Synapse.

Pushing the MODE button repeatedly steps Synapse forward through 18 different pages.

Holding the MODE button and pressing any of buttons #2 through #18 causes Synapse to go to the selected page #2 through #18.

Holding the MODE button and pressing the #1 button causes Synapse to back-step two pages.

As previously mentioned, pressing WRITE and MODE at the same time initiates the PANIC function.

Program Number and Mode Display
The two-digit display shows the currently selected program number. This display also indicates the currently selected Processor or Patch Map page.

Assignment Buttons and Twenty Digit Display
The row of twenty assignment push buttons under the display are used to enter information, and the twenty digit display tells you what you've entered or what you've already stored in memory.

The specific function of the twenty buttons varies depending on the page currently selected by the MODE switch. Synapse always turns on in Program Mode, where the first ten buttons are used for recalling a program from memory. Alternately, buttons #19 and #20 serve as program number increment and decrement keys, respectively.

In the Assignment and Processor Modes, holding any of the twenty buttons for more than a second will cause the number in the display above the button to automatically scroll upwards.

Synapse FRONT PANEL

All Synapse functions are controlled by the front panel buttons. From left to right, we have:

WRITE
This is used for writing (that is, storing) an assignment into memory.

Also, pushing the WRITE button at any time returns Synapse to the Program Mode.
(This is the mode for recalling assignments from memory.)

If WRITE and MODE are held down at the same time, the PANIC function is initiated.

MODE
This button is used for selecting the desired Synapse pages to enter, review, or alter all the parameter settings in Synapse.

Pushing the MODE button repeatedly steps Synapse forward through 18 different pages.

Holding the MODE button and pressing any of buttons #2 through #18 causes Synapse to go to the selected page #2 through #18.

Holding the MODE button and pressing the #1 button causes Synapse to back-step two pages.

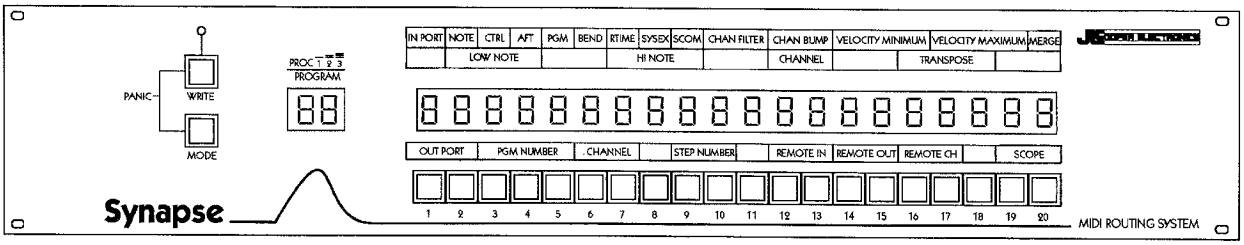
As previously mentioned, pressing WRITE and MODE at the same time initiates the PANIC function.

Program Number and Mode Display
The two-digit display shows the currently selected program number. This display also indicates the currently selected Processor or Patch Map page.

Assignment Buttons and Twenty Digit Display
The row of twenty assignment push buttons under the display are used to enter information, and the twenty digit display tells you what you've entered or what you've already stored in memory.

The specific function of the twenty buttons varies depending on the page currently selected by the MODE switch. Synapse always turns on in Program Mode, where the first ten buttons are used for recalling a program from memory. Alternately, buttons #19 and #20 serve as program number increment and decrement keys, respectively.

In the Assignment and Processor Modes, holding any of the twenty buttons for more than a second will cause the number in the display above the button to automatically scroll upwards.



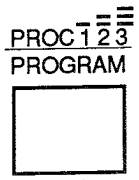
MODES OF OPERATION

Although Synapse has many modes of operation, most of the time you will only need to use two: Program Mode and Assignment Mode. Synapse always turns on in Program Mode.

The Program Mode is for recalling programs from memory. The two-digit display shows th currently selected program number. Assignment Mode is for routing MIDI inputs to MIDI outputs. The two-digit display is blank in Assignment Mode.

To select the Program Mode at any time for recalling programs from memory, tap the WRITE button once.

To select the Assignment Mode to route inputs to outputs, tap the MODE button once.



Synapse has a total of 19 pages of operation. There are three easy ways to get to the differe pages, depending on your personal preference.

(1) Pushing the MODE button repeatedly causes Synapse to cycle through the 1 pages.

(2) Holding in the MODE button and pressing any of buttons #2 through #19 causes Synapse to go directly to the selected page #2 through #19.

(3) Holding in the MODE button and pressing button #1 causes Synapse to back step two pages.

Repeatedly tapping the MODE button produces the sequence shown on the next page.

After the Program and Assignments Modes are the pages for programming the MIDI Data Processors.

Each Processor has five pages: A master page and four zones.

After the Processor pages is the page for entering a Patch Map.

After the Patch Map page is the page for displaying incoming MIDI Time Code.

Pushing the MODE button one more time returns Synapse to the Program Mode.

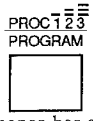
MODES OF OPERATION

Although Synapse has many modes of operation, most of the time you will only need to use two: Program Mode and Assignment Mode. Synapse always turns on in Program Mode.

The Program Mode is for recalling programs from memory. The two-digit display shows the currently selected program number. Assignment Mode is for routing MIDI inputs to MIDI outputs. The two-digit display is blank in Assignment Mode.

To select the Program Mode at any time for recalling programs from memory, tap the WRITE button once.

To select the Assignment Mode to route inputs to outputs, tap the MODE button once.



Synapse has a total of 19 pages of operation. There are three easy ways to get to the different pages, depending on your personal preference.

(1) Pushing the MODE button repeatedly causes Synapse to cycle through the 19 pages.

(2) Holding in the MODE button and pressing any of buttons #2 through #19 causes Synapse to go directly to the selected page #2 through #19.

(3) Holding in the MODE button and pressing button #1 causes Synapse to back step two pages.

Repeatedly tapping the MODE button produces the sequence shown on the next page.

After the Program and Assignments Modes are the pages for programming the MIDI Data Processors.

Each Processor has five pages: A master page and four zones.

After the Processor pages is the page for entering a Patch Map.

After the Patch Map page is the page for displaying incoming MIDI Time Code.

Pushing the MODE button one more time returns Synapse to the Program Mode.

page 1
Program Mode

page 2
Assignment Mode

page 3
First Processor
Master Page

page 4
First Processor
Zone 1

page 5
First Processor
Zone 2

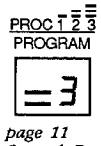
page 6
First Processor
Zone 3

page 7
First Processor
Zone 4

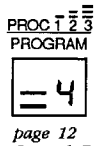
page 8
Second Processor
Master Page

page 9
Second Processor
Zone 1

page 10
Second Processor
Zone 2



page 11
Second Processor
Zone 3



page 12
Second Processor
Zone 4

page 13
Third Processor
Master Page

page 14
Third Processor
Zone 1

page 15
Third Processor
Zone 2

page 16
Third Processor
Zone 3

page 17
Third Processor
Zone 4

page 18
Program Change
(Patch Map)

HOOK UP

Special Notice about AC Line Conditioning:
All computer controlled equipment (including sequencers, drum machines, tone generators, and signal processors) must be protected from variations in the AC line. The Synapse is no exception. Lightning strikes, "brown outs", even nearby motorized equipment can scramble memories and in some cases cause damage to the hardware.

It is strongly recommended that all your computer-related equipment (including the Synapse) be plugged into an AC Line Conditioner. Many brands of multiple-outlet power strips at least have circuit breakers and surge suppressors. But ideally the Synapse should also be protected with a power strip that includes Line Filtering as well.

Many brands of AC Line Conditioners are sold in Music, Consumer Electronics, Computer, and Hardware stores.

When hooking up your system, try to use good quality MIDI cables with a known history. (Please avoid the use of ancient, mystery molded cables from Siberia.)

Any keyboard, controller, sequencer, computer, tone module, effect, drum machine, –anything with a MIDI output, hooks up to the Synapse's sixteen MIDI inputs.

Likewise, the twenty MIDI outputs of Synapse are hooked up to the MIDI inputs of any device that will receive MIDI.

Certain devices, such as MIDI-controlled audio signal processors, might only need to receive MIDI data. In that case, use only a Synapse MIDI output to go to the signal processor's MIDI input. (That is why Synapse has more outputs than inputs.) However, bi-directional communication may be necessary if you intend to use that signal processor with a MIDI System Exclusive Librarian software package. In that case, you will want to hook up both inputs and outputs.

For the sake of simplicity,you may use corresponding input and output numbers, though this is not required. For example, for a given sequencer, hook up the cables to Synapse input #1 and output #1, hook up a keyboard controller to Synapse input #2 and output #2, etc.

It is a good idea to write down (in the form of a table) a list of what instruments go to which inputs and outputs.

Since the twenty buttons will correspond to the twenty outputs (see next chapter), you might find it easier to keep track of things if you run a thin strip of masking tape below the push buttons. Label each button with the name of the MIDI device hooked up to the output. For example, if MIDI output number 14 on Synapse goes to the MIDI input of a DX-7 II, write "DX-7" underneath button 14 on a piece of tape.

ASSIGNMENT MODE

This chapter describes the use of Synapse as a MIDI routing device.

All routing assignments are entered in Assignment Mode. This mode is characterized by the blank program display (see figure 1). There are several easy ways to quickly put Synapse into Assignment Mode.

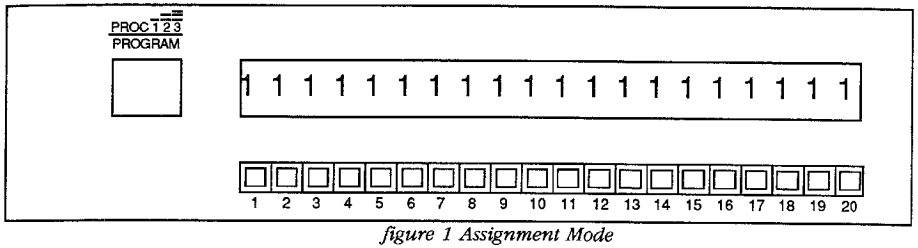


figure 1 Assignment Mode

To select the Assignment Mode, turn on Synapse and Press MODE Once.

Synapse always powers up in Program Mode, so pressing MODE once steps Synapse to its second page, Assignment Mode.

To Return to the Assignment Mode at any time, do either of the following

(1) Tap WRITE once and then tap MODE once, or

(2) Hold down MODE and press button #2.

Both of these methods achieve the same results.

The principle of routing MIDI inputs and outputs is as follows:

Each of the 20 MIDI output ports of Synapse has its own selector button. A number is displayed on the LEDs above each selector button. This number indicates which of the sixteen MIDI inputs is controlling that output port.

For example, MIDI data comes out of a sequencer and goes into a Synapse MIDI Input. Any input of Synapse can be used, but say for example that the sequencer's MIDI out is hooked up to Synapse's MIDI In number 3.

Lets say that Synapse's MIDI Outputs numbered 12 through 15 go to some tone modules. The buttons numbered 12 through 15 on Synapse correspond to the tone modules.

To route the MIDI output of the sequencer to the MIDI inputs of the tone modules, a number "3" must be displayed in the LEDs above buttons numbered 12 through 15. Buttons 12 through 15 are pushed until a "3" is displayed. This is shown in figure 2.

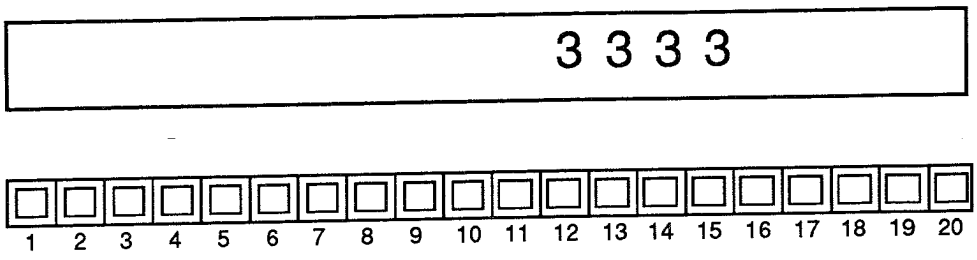


figure 2 Synapse input three is routed to outputs twelve through fifteen

Remember that the LEDs indicate sources of MIDI (sending data), the button numbers indicate destinations of MIDI (receiving data).

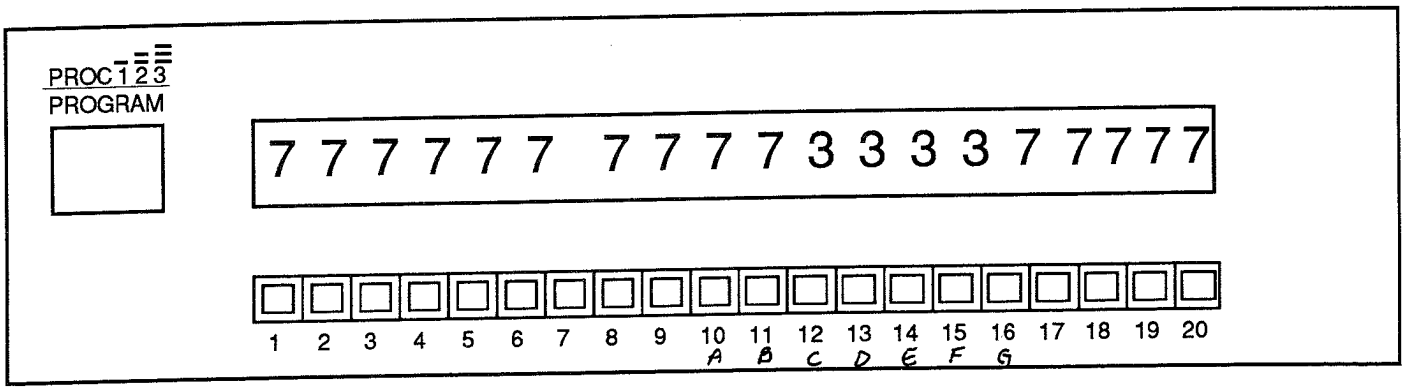
Let's look more closely at an output assignment button and its associated LED display. There are sixteen inputs that a given output could be assigned to. But there is only a one digit LED available to display this information. (And we are probably already pushing a world-record for 7-segment LEDs). A modified hex style display is used to indicate input numbers from
10 through 16.

A = 10
B = 11
C = 12
D = 13
E = 14
F = 15
G = 16

→ Note that a **blank display** indicates that the output is not receiving any MIDI data from any source.

If a button is held down, the display will scroll from 1 through 9, then A through G, and then blank. This is followed by special symbols to indicate that a Processor or Merger is sending to that output and will be covered in detail in the following chapters

Now we will add to the example above. A Keyboard Synthesizer is hooked up to Synapse MIDI input and output number 7. Create a Synapse assignment to route the keyboard to the remaining outputs. See figure.



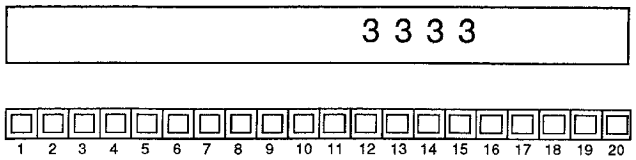


figure 2 Synapse input three is routed to outputs twelve through fifteen

Remember that the LEDs indicate sources of MIDI (sending data), the button numbers indicate destinations of MIDI (receiving data).

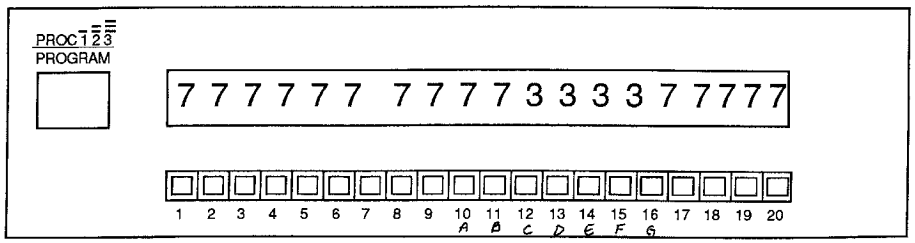
Let's look more closely at an output assignment button and its associated LED display. There are sixteen inputs that a given output could be assigned to. But there is only a one digit LED available to display this information. (And we are probably already pushing a world-record for 7-segment LEDs). A modified hex style display is used to indicate input numbers from
10 through 16.

A = 10
B = 11
C = 12
D = 13
E = 14
F = 15
G = 16

→ Note that a **blank display** indicates that the output is not receiving any MIDI data from any source.

If a button is held down, the display will scroll from 1 through 9, then A through G, and then blank. This is followed by special symbols to indicate that a Processor or Merger is sending to that output and will be covered in detail in the following chapters

Now we will add to the example above. A Keyboard Synthesizer is hooked up to Synapse MIDI input and output number 7. Create a Synapse assignment to route the keyboard to the remaining outputs. See figure.



Now when you play the keyboard, MIDI instruments hooked up to the remaining outputs are controlled.

Notice, however, that we have placed a blank above button seven. If a number seven were displayed above button seven, you would be routing the keyboard back to itself. (This will not hurt anything, it has the same effect as looping a MIDI cable from the keyboard's input right to the output. This layering would rob half of your keyboard synthesizer's 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.

As another example, lets add a drum machine on Synapse input and output number 10 (A). Assume that our sequencer (input and output 3) is a computer that also supports a MIDI System Exclusive Librarian software package. The Keyboard Synthesizer is still on input and output 7, and for this example the Tone Modules are on 4, 5, 6, and 9. We want to create a routing assignment on Synapse such that:

(1)The Sequencer/Computer sends MIDI to two of the Tone Modules, the Keyboard Synthesizer and the Drum Machine.

(2)One of the tone modules can send its data back to Sequencer/Computer.

(3)The Keyboard Synthesizer controls one of the tone modules.

(4)The drum machine fires the remaining tone module. The solution is shown in figure 4.

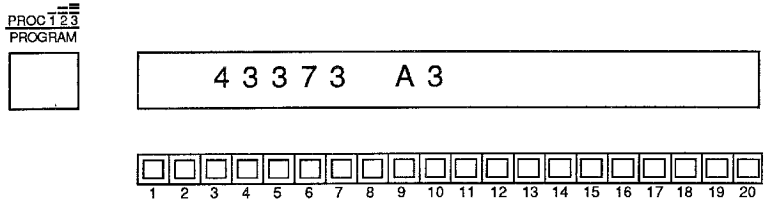


figure 4 Notice that there is two-way communication between the Computer and the Tone Module. (3 goes to 4, 4 goes to 3)

To create this assignment, first note that the Sequencer/Computer is sending MIDI data to outputs 4 and 5 (two of the Tone Modules), output 7 (the keyboard synth) and output 10 (the drum machine. So a "3" must be displayed above 4, 5, 7, and 10.

The first tone module is hooked up to Synapse input and output 4. To send MIDI data from the tone module into the Sequencer/Computer, a 4 must be displayed above button 3.

One of the tone modules is controlled by the keyboard, so a "7" is displayed above button 6. The remaining tone module is controlled by the drum machine, so an "A" is displayed above button 9.

Synapse Presets

Synapse memory locations 44 through 63 come pre-programmed with routing assignments. Each program routes one input to all the outputs. Each successive program number uses the next input number. That is, when you recall program 44, all 2's are shown across the display. This routes input 2 to all the outputs. When you recall program 45, all 3's are shown across the display. This routes input 3 to all the outputs.

Use these presets as a time-saving starting point for creating your own assignments. For example, say that you want to assign every output to input 9. Instead of scrolling on each button until a "9" is displayed, you could simply recall Program #51 already stored in Synapse.

These assignments have been pre-loaded merely as a convenience. You do not have to use them, and of course you can always write over them. See next chapter for more details about writing to memory and recalling from memory.

Synapse Presets
Synapse memory locations 44 through 63 come pre-programmed with routing assignments. Each program routes one input to all the outputs. Each successive program number uses the next input number. That is, when you recall program 44, all 2's are shown across the display. This routes input 2 to all the outputs. When you recall program 45, all 3's are shown across the display. This routes input 3 to all the outputs.

Use these presets as a time-saving starting point for creating your own assignments. For example, say that you want to assign every output to input 9. Instead of scrolling on each button until a "9" is displayed, you could simply recall Program #51 already stored in Synapse.

These assignments have been pre-loaded merely as a convenience. You do not have to use them, and of course you can always write over them. See next chapter for more details about writing to memory and recalling from memory.

PROGRAM MODE

The Program Mode is for writing (storing) and recalling assignments. There are 64 programs stored in memory. Synapse always turns on in Program Mode, it is the first page of operation.

To quickly return to program mode at any time simply tap the WRITE button once.

In this mode, the buttons are used to select program numbers to recall assignments from memory or write assignments to memory.

Programs are numbered 01 through 64.

Use just the first ten buttons to select program numbers. For example:
To recall program number 24, push 2, then push 4.
To recall program number 40, push 4, then push 10.

There is no "0" button, so you may use use the #10 button to mean either 0 or 10, like this:
To recall program number 09, push 10, then push 9.

Pushing Button #19 decrements the Program Number.
Pushing Button #20 increments the Program Number.
Holding either of these buttons will cause Synapse to scroll through the Program Numbers.

To write (that is, store) a program into memory, push and hold in the WRITE button. While holding WRITE, press the two numbers that correspond to the program number.

For example:
To write an assignment into program number 50, push and hold in the WRITE button. While holding WRITE, press 5, then press 10.
To write a program into memory location number 01, hold the WRITE button. While holding WRITE, press 10, then press 1.

Special Notice to former MSB 16/20 users: You will notice that the Synapse uses true decimal program numbering, unlike the J.L. Cooper MSB 16/20 which used Octal numbers 11 through 88.

Example
Suppose that you have two sequencers. The first is hooked up to Synapse MIDI input and output 1, the second is hooked up to Synapse MIDI input and output 2. You have some number of slave devices (rack-mount expander modules and samplers and such) connected to Synapse MIDI outputs 3 through 20.

Create two different assignments, and write them to two different memory locations, so that you can select which sequencer controls all the slaves.

First, put Synapse into Assignment Mode. Remember that there are several ways to do this. If you are already in Program Mode (a two-digit number is shown in the Program Display), simply push MODE once. Otherwise, push WRITE once then push MODE once.

For the first assignment, you'll want a "1" above every button except buttons 1 and 2. This routes sequencer 1 to outputs 3 through 20, but does not route sequencer 1 back to itself. See figure 1.

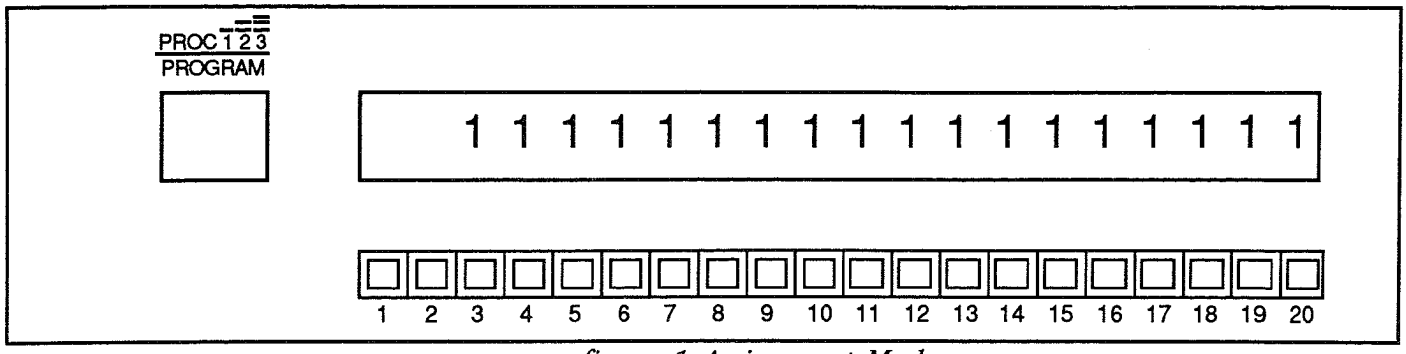
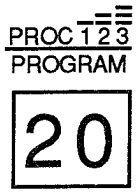


figure 1 Assignment Mode
Sequencer 1 is the master



Write this assignment to program number 20. Hold down WRITE, and while holding WRITE press 2 and then 10.

Now put Synapse back into Assignment Mode by pushing the MODE button once.

For the other assignment, you'll want a "2" above every button except buttons 1 and 2. This routes sequencer 2 to outputs 3 through 20, but does not route sequencer 2 back to itself. See figure 3.

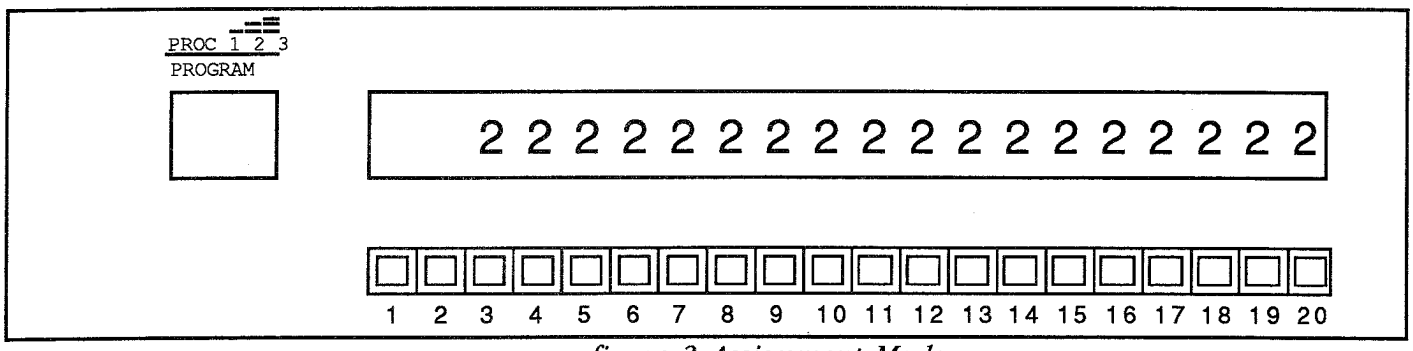


figure 3 Assignment Mode
Sequencer 2 is the master

Write this assignment to program number 21. Hold down WRITE, and while holding WRITE press 2 and then 1.

Now whenever you are in Program Mode, selecting either 20 or 21 determines which sequencer is controlling all MIDI slave devices. In essence, the touch of a few buttons will completely re-wire your MIDI studio. Remember that you can quickly return to Program Mode at any time by pressing the Program button.

The previous example was given for the sake of clarity, to show how to write two different assignments into memory. In actual use, however, you will find that it is possible in some circumstances to achieve the same results using only one memory location. The Merger in Synapse allows both sequencers to simultaneously control all the MIDI slave devices. This is a more advanced operation of Synapse and will be covered in later chapters.

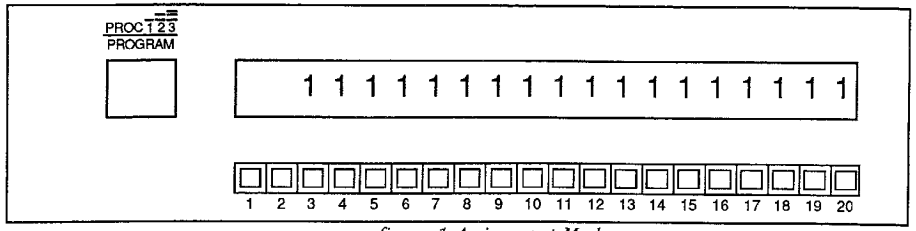


figure 1 Assignment Mode
Sequencer 1 is the master

Write this assignment to program number 20. Hold down WRITE, and while holding WRITE press 2 and then 10.

Now put Synapse back into Assignment Mode by pushing the MODE button once.

For the other assignment, you'll want a "2" above every button except buttons 1 and 2. This routes sequencer 2 to outputs 3 through 20, but does not route sequencer 2 back to itself. See figure 3.

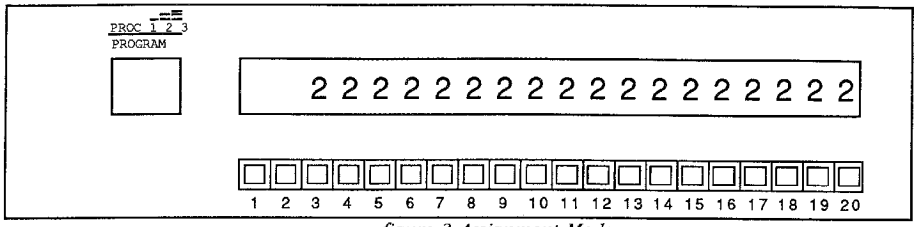


figure 3 Assignment Mode
Sequencer 2 is the master

Write this assignment to program number 21. Hold down WRITE, and while holding WRITE press 2 and then 1.

Now whenever you are in Program Mode, selecting either 20 or 21 determines which sequencer is controlling all MIDI slave devices. In essence, the touch of a few buttons will completely re-wire your MIDI studio. Remember that you can quickly return to Program Mode at any time by pressing the Program button.

The previous example was given for the sake of clarity, to show how to write two different assignments into memory. In actual use, however, you will find that it is possible in some circumstances to achieve the same results using only one memory location. The Merger in Synapse allows both sequencers to simultaneously control all the MIDI slave devices. This is a more advanced operation of Synapse and will be covered in later chapters.

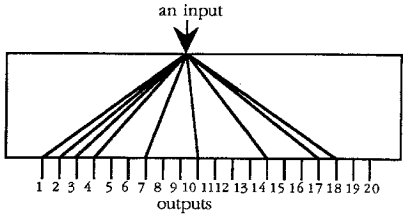
MIDI DATA PROCESSORS

General Information

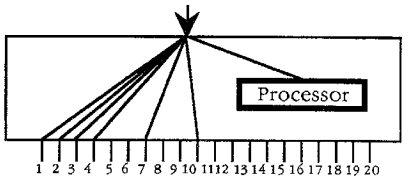
Synapse's three MIDI Data Processors can significantly modify any MIDI information before it reaches an output. The microprocessor inside Synapse actually intercepts MIDI data and changes it. Notes can be transposed, Velocity altered, Channels changed, and specific types of information can be removed.

The following block diagrams show how the Processors work along with MIDI routing.

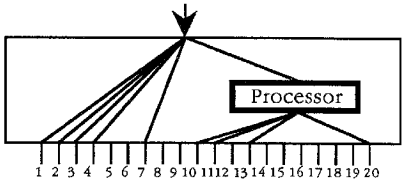
Normally, MIDI data comes into a Synapse MIDI input and is routed to some number of outputs.



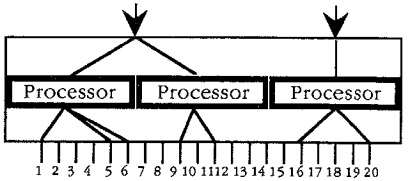
MIDI data coming into Synapse can also be routed into a Processor.



The Processors then are routed to some number of outputs. You can see that MIDI can pass though Synapse in both processed and unprocessed forms, at the same time.



More than one Processor can be assigned to the same input. That means several different versions of the original MIDI signal are simultaneously available. For example, different outputs of Synapse can be sending the same keyboard note data, but with different MIDI velocity ranges and different MIDI channel numbers.



You have three processors to work with. These three can be assigned to process three different inputs, or can even be assigned to the same input.

Please Note: From the above diagrams you can clearly see that a Processor doesn't actually do anything unless it has been assigned to <u>both</u> an input and an output destination.

Processor Features and Suggested Applications
Each processor can perform the following functions: Data Filter, Channel Filter, Channel Bump, Velocity Rescale, 4 Zone Transpose, and 4 Zone Channel Bump.

Data Filter
Data Filtering selectively removes any one or more of the following types of MIDI messages:

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

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

Channel Filter
The Channel Filter removes all MIDI data passing through the Processor except data on one selected MIDI channel. MIDI data goes into the Channel Filter after passing through the Data Filter.

Channel Filter Applications
Some early MIDI instruments, like the Emulator I and the Memory Moog, are permanently in Omni Mode. This makes them nearly impossible to use with a sequencer, because they will always try to play every track. The Processor's Channel Filter allows you to select a single MIDI channel that will control the older instrument, essentially "de-omnifying" it.

The Channel Filter also allows you to select a specific channel for further processing. You might have a sequencer playing 16 tracks on 16 MIDI channels through Synapse. You could pick out, say for example, Track 7 only for rescaling its velocity or mapping its MIDI channel.

Channel Bump
Channel Bump shifts the MIDI channel of Controller, Pitch Bend, After Touch, and Program Change data passing through the Processor. This Channel Bump does not affect notes. It does this by adding a user-selected interval to the incoming MIDI channel number. MIDI data goes into Channel Bump after passing through the Channel Filter.

Channel Bump Applications
Channel Bumping can save Sequencer memory by permitting a single MIDI Volume track on one MIDI channel to control several different tone modules on different MIDI channels

A keyboard could be processed with Channel Bump along with Zone Channel Bump (see below), so that both After Touch and Note data can be sent out of Synapse on more than one channel at a time.

Processor Features and Suggested Applications
Each processor can perform the following functions: Data Filter, Channel Filter, Channel Bump, Velocity Rescale, 4 Zone Transpose, and 4 Zone Channel Bump.

Data Filter
Data Filtering selectively removes any one or more of the following types of MIDI messages:

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

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

Channel Filter
The Channel Filter removes all MIDI data passing through the Processor except data on one selected MIDI channel. MIDI data goes into the Channel Filter after passing through the Data Filter.

Channel Filter Applications
Some early MIDI instruments, like the Emulator I and the Memory Moog, are permanently in Omni Mode. This makes them nearly impossible to use with a sequencer, because they will always try to play every track. The Processor's Channel Filter allows you to select a single MIDI channel that will control the older instrument, essentially "de-omnifying" it.

The Channel Filter also allows you to select a specific channel for further processing. You might have a sequencer playing 16 tracks on 16 MIDI channels through Synapse. You could pick out, say for example, Track 7 only for rescaling its velocity or mapping its MIDI channel.

Channel Bump
Channel Bump shifts the MIDI channel of Controller, Pitch Bend, After Touch, and Program Change data passing through the Processor. This Channel Bump does not affect notes. It does this by adding a user-selected interval to the incoming MIDI channel number. MIDI data goes into Channel Bump after passing through the Channel Filter.

Channel Bump Applications
Channel Bumping can save Sequencer memory by permitting a single MIDI Volume track on one MIDI channel to control several different tone modules on different MIDI channels

A keyboard could be processed with Channel Bump along with Zone Channel Bump (see below), so that both After Touch and Note data can be sent out of Synapse on more than one channel at a time.

Velocity Rescale
The Velocity Rescale function acts as a sort of MIDI dynamic compressor/expander. The velocity values of incoming notes are changed and made to fall within a range selected by the user. MIDI data goes into Velocity Rescale after passing through the Channel Bump.

Velocity Rescale Applications
Quite often certain synthesizer patches only sound good at certain velocities. Some instruments can sound too bright and distorted if they receive high velocities, and velocity response does vary from keyboard to keyboard. For example, Yamaha synths can sound very different than expected when controlled by Kurzweil keyboards. Velocity rescaling allows master and slave keyboards to track MIDI velocity with a uniform, predictable response.

The Processor also features a special "V-squared" velocity curve which compresses the velocity range up to a certain threshold. A nearly constant velocity is sent out unless the keys on the controlling keyboard are struck hard. At that point, notes are sent out at maximum velocity.

Zone Channel Bump and Transpose
Each Processor has four Zones. For each Zone, a low note and a high note is selected. MIDI note data that falls within the note value range can be both Channel Bumped, and Transposed. Zones allow you to split up your keyboard controller, so that the channel it sends is dependant upon which notes you play.

The Zone Channel Bump is like the Channel Bump already described, except it only applies to notes that fall within the Zone. Transpose adds to or subtracts from MIDI note numbers passing through the Processor within the Zone. Notes may be transposed plus or minus five octaves in semitone steps (±59 semi tones). MIDI data goes into the four Zones after passing through the Velocity Rescale.

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

In live performance, the Processors permit you to define Zones on your keyboard such that the notes you play determine which tone module will sound. If the slave is a multi-timbrel tone module, you can decide in real time which voices you will hear simply by playing different ranges of the keyboard. Although a lot of keyboards have this capability already built in, many do not. Those that do have a limit to the number of MIDI channels that you can send on for a given note.

Zoning solves a common sequencing problem, too. Suppose that Synapse 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 recieve on channels 1 through 8.

Then you recall a different Synapse 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.

Zone Channel Bump and Transpose Applications Continued

Using Zone Channel Bump solves this problem. A Processor or combination of Processors can be set up to make your controlling keyboard send data on many channels at once, controlling all the slaved tone modules. Thus the tone modules can always be left on different MIDI channels than the keyboard.

Zone **Channel Bumping** can be as simple as a single keyboard split. Or it can be as massive as having a single keyboard sending on up to 13 MIDI channels at once!

Zone **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 Synapse to transpose MIDI Notes, rather than to change the key of each receiving instrument.

Zone Channel Bump and Transpose Applications Continued

Using Zone Channel Bump solves this problem. A Processor or combination of Processors can be set up to make your controlling keyboard send data on many channels at once, controlling all the slaved tone modules. Thus the tone modules can always be left on different MIDI channels than the keyboard.

Zone **Channel Bumping** can be as simple as a single keyboard split. Or it can be as massive as having a single keyboard sending on up to 13 MIDI channels at once!

Zone **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 Synapse to transpose MIDI Notes, rather than to change the key of each receiving instrument.

The Processor Pages
Each processor has five pages, one master page and four zones. A Processor master page is shown in figure 5.

Remember that you can step up through the pages of Synapse by tapping the MODE button repeatedly. Back step two pages by holding MODE and tapping the #1 button. Or, press and hold in MODE and while holding, press any one of buttons #2 through #18 to directly recall the desired page. Tap WRITE once if you need to return to page 1, Program Mode.

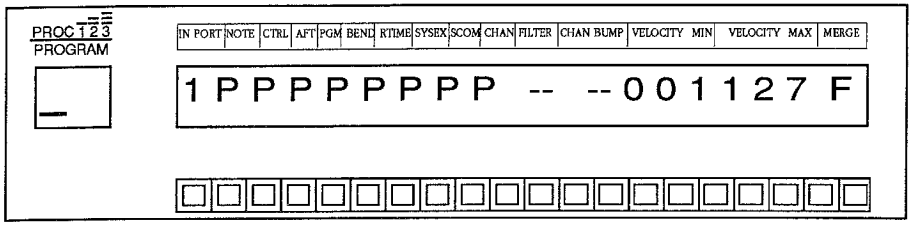


figure 5 The first processor, master page. Note the single dash in the Program Display

Assigning the Processors to a MIDI Input
To use the Processors, you will first need to assign them to an input port. That is, which "master" do you want to process? You will also need to assign the Processor to an output port, or some combination of output ports. That is, which slaves do you want to receive processed data?

Use MODE button to select the first Processor master page. Refer to the topmost legend on the front panel. The first (left hand) number (over button #1) displays the current input assignment of the processor. For example, if an "8" is displayed, that means Synapse input #8 is going into that Processor.

Tap button #1 to change this assignment. Hold button #1 to scroll. The display uses the same input convention as the Assignment Mode, inputs 1 through 9, followed by inputs A (10) through G (16). A blank indicates that this Processor is turned off.

Important. The mere fact that you have assigned a processor to an input does not influence Synapse outputs that have been assigned to that the input. For example, say that a Processor is assigned to input number 4. If in Assignment Mode a "4" is displayed above button #1, that still means that input 4 is routed unprocessed to output 1. Refer to the block diagrams at the beginning of this chapter.

The second and third Processors are assigned an input port in a similar manner. Select their master page (either), then use the #1 button to assign an input port.

Assigning the Processors to a MIDI Output

Return Synapse to the Assignment Mode. There are two ways to quickly do this. Either back step by holding mode and tapping the #1 key until the program display blanks. Alternately, you may tap the WRITE button once and the MODE button once.

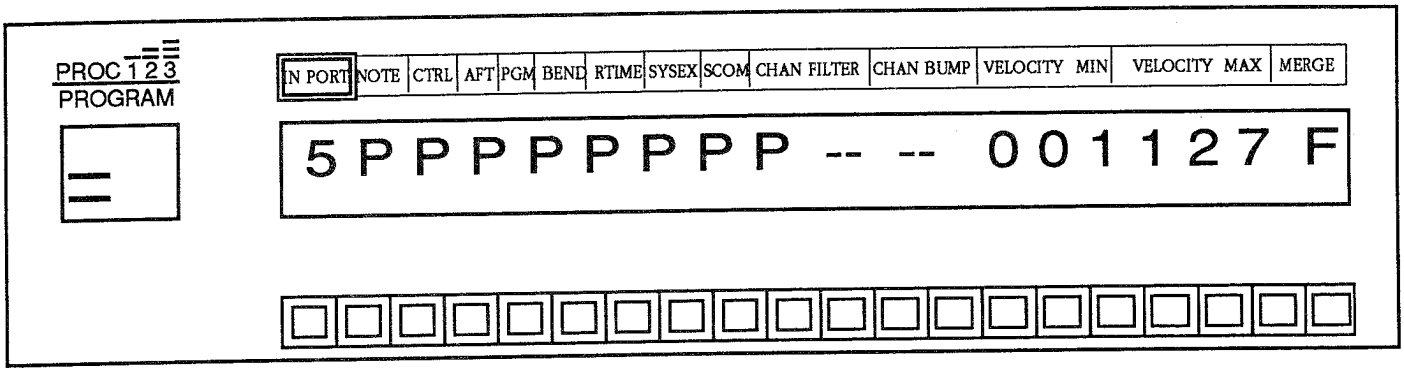
Tap an output assignment button to increment the number in the display, or hold to scroll. The display counts from 1 through 9, and then A through G, then blank (off) . After the blank, a "▬ " symbol appears. This means that the output of the first Processor is being sent to that Synapse output port.

Tap the button again and the display shows a "〓 ". The output of the second Processor is now being sent out of that Synapse output port.

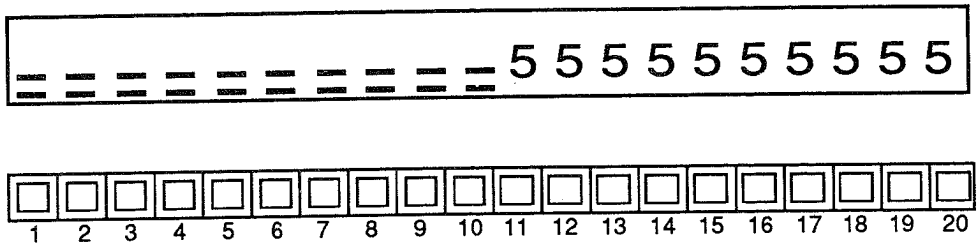
Tap the button again and the display shows a "≡ ". The output of the third Processor is now being sent out of that Synapse output port.

For example, say that you have a keyboard controller plugged into Synapse MIDI input 5. You want to process the keyboard data using the second Processor, and send the processed data to Synapse outputs 1 through 10. You want to route the unprocessed data to outputs 11 through 20.

Solution: First assign the second processor to input 5. Tap the MODE button until the display indicates the second Processor's master page, [▬]. Then push button #1 until a "5" is displayed above it.



Then return to the Assignment Mode, and push each assignment button until you see this display:



Assigning the Processors to a MIDI Output

Return Synapse to the Assignment Mode. There are two ways to quickly do this. Either back step by holding mode and tapping the #1 key until the program display blanks. Alternately, you may tap the WRITE button once and the MODE button once.

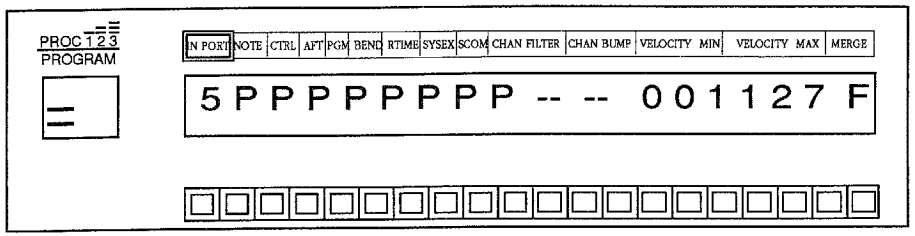
Tap an output assignment button to increment the number in the display, or hold to scroll. The display counts from 1 through 9, and then A through G, then blank (off) . After the blank, a "▬ " symbol appears. This means that the output of the first Processor is being sent to that Synapse output port.

Tap the button again and the display shows a "〓 ". The output of the second Processor is now being sent out of that Synapse output port.

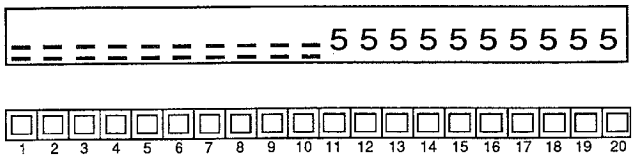
Tap the button again and the display shows a "≡ ". The output of the third Processor is now being sent out of that Synapse output port.

For example, say that you have a keyboard controller plugged into Synapse MIDI input 5. You want to process the keyboard data using the second Processor, and send the processed data to Synapse outputs 1 through 10. You want to route the unprocessed data to outputs 11 through 20.

Solution: First assign the second processor to input 5. Tap the MODE button until the display indicates the second Processor's master page, [▬]. Then push button #1 until a "5" is displayed above it.



Then return to the Assignment Mode, and push each assignment button until you see this display:



Programming the Processors

Now that we have the background information on what the processors do and how they are assigned, we will discuss the specifics of programming the processors and give examples.

To Filter MIDI Data

First, assign a Processor to an input using the previous directions. Observe the row of P's and F's. "P" means Pass and "F" means Filter. This indicates whether a given MIDI message will Pass through or be Filtered out on its trip through the Processor. Push the button under one of the Filter functions. Notice that it toggles between "F" and "P". Use the top-most legend above the display to determine which MIDI message is being affected for each button. Any combination of P's and F's may be entered.

For example, suppose that your MIDI system is arranged as follows:
The output of a sequencer goes into the Synapse MIDI input number 1,
the output of a keyboard goes into the Synapse MIDI input A (10),
Synapse output 5 goes to a drum machine input,
Synapse outputs 14 through 20 go to the inputs of tone modules.

Set the first Processor to filter out Note data from the sequencer, and set the third Processor to filter out Pitch Bend from the keyboard.

Then send the output of the first Processor to the drum machine. Send the output of the third Processor to the tone modules on outputs 18 through 20.

Complete the assignment by routing the sequencer (unprocessed) to tone modules 14 through 16. Route the keyboard (unprocessed) to tone module 17.

Solution:
You will need to assign the two Processors to the desired input numbers. Then program the Processors by selecting which filters are to be active. Then route the Processors and the inputs to the desired outputs. You may do this in any order.

First select the first Processor's master page by pushing the MODE button until the symbol for the first Processor is displayed. Assign it to the sequencer (input 1) and push the second button to filter out NOTE commands. See figure 8.

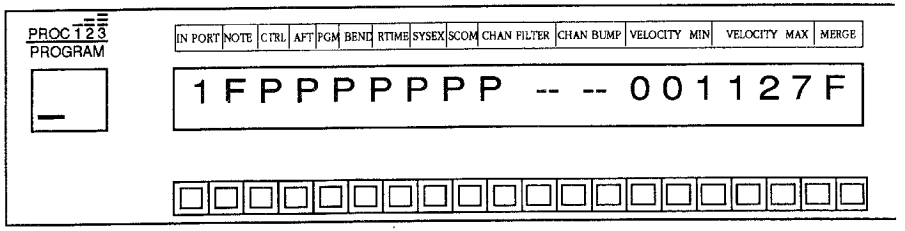


figure 8 The first processor, master page. Note the "F" above button #2.

To Channel Bump MIDI Data
On any Processor master page, use push buttons #12 and #13 to select the channel bump interval that be added to the MIDI channel of data passing through the Processor. This affects data after it has passed through the Data and Channel Filters.

That is to say, if data comes into the Processor on MIDI channel 1, and a "1" is entered under Channel Bump, this data will come out of the Processor on MIDI channel 2.
A zero indicates no channel bumping.

This Channel Bump shifts the MIDI channel of Controller, Pitch Bend, After Touch, and Program Change data passing through the Processor. This Channel Bump does not affect notes.

If the sum of the channel number coming in plus the channel bump interval is greater than 16, the channel numbers will wrap around. For example, if MIDI comes in on channel 13, and a channel bump interval of 4 is selected, MIDI will go out on channel 1.

For example, say that the MIDI output of a sequencer goes into the Synapse MIDI input number 2. Synapse MIDI outputs 5 through 8 go to four tone generators.

The sequencer has a MIDI Volume track on channel 1. The four tone generators are on channels 1, 2, 3, and 4. Use Channel Bump so that all four modules will respond to the one MIDI Volume track.

Solution:
You will need to assign all three Processors to the same input number. Then enter a Channel Bump of 1 on the first Processor, 2 on the second Processor, and 3 on the third Processor. Then assign the sequencer directly to Synapse output 5. That is the module on channel 1. Assign the first, second, and third Processors to outputs 6, 7, and 8, respectively.

First select the first Processor's master page by pushing the MODE button until the symbol for the first Processor is displayed. (Holding in MODE and pressing #3 achieves the same result.) Assign it to the sequencer (input 2) use push buttons #12 and #13 to enter "01" under the legend CHAN BUMP.

Then select the second Processor's master page by pushing the MODE button until the symbol for the second Processor is displayed. (Holding in MODE and pressing #8 achieves the same result.) Assign this Processor also to input 2, and use push buttons #12 and #13 to enter "02" under the legend CHAN BUMP.

Then select the third Processor's master page by pushing the MODE button until the symbol for the second Processor is displayed. (Holding in MODE and pressing #13 achieves the same result.) Assign this Processor also to input 2, and use push buttons #12 and #13 to enter "03" under the legend CHAN BUMP.

Then return to the Assignment Mode. Either tap WRITE once and then MODE once, or hold MODE and press #2.

Place a "2" above button #5. Place the symbol for the first Processor above button #6.
Place the symbol for the second Processor above button #7.
Place the symbol for the third Processor above button #8.

To Channel Bump MIDI Data
On any Processor master page, use push buttons #12 and #13 to select the channel bump interval that be added to the MIDI channel of data passing through the Processor. This affects data after it has passed through the Data and Channel Filters.

That is to say, if data comes into the Processor on MIDI channel 1, and a "1" is entered under Channel Bump, this data will come out of the Processor on MIDI channel 2.
A zero indicates no channel bumping.

This Channel Bump shifts the MIDI channel of Controller, Pitch Bend, After Touch, and Program Change data passing through the Processor. This Channel Bump does not affect notes.

If the sum of the channel number coming in plus the channel bump interval is greater than 16, the channel numbers will wrap around. For example, if MIDI comes in on channel 13, and a channel bump interval of 4 is selected, MIDI will go out on channel 1.

For example, say that the MIDI output of a sequencer goes into the Synapse MIDI input number 2. Synapse MIDI outputs 5 through 8 go to four tone generators.

The sequencer has a MIDI Volume track on channel 1. The four tone generators are on channels 1, 2, 3, and 4. Use Channel Bump so that all four modules will respond to the one MIDI Volume track.

Solution:
You will need to assign all three Processors to the same input number. Then enter a Channel Bump of 1 on the first Processor, 2 on the second Processor, and 3 on the third Processor. Then assign the sequencer directly to Synapse output 5. That is the module on channel 1. Assign the first, second, and third Processors to outputs 6, 7, and 8, respectively.

First select the first Processor's master page by pushing the MODE button until the symbol for the first Processor is displayed. (Holding in MODE and pressing #3 achieves the same result.) Assign it to the sequencer (input 2) use push buttons #12 and #13 to enter "01" under the legend CHAN BUMP.

Then select the second Processor's master page by pushing the MODE button until the symbol for the second Processor is displayed. (Holding in MODE and pressing #8 achieves the same result.) Assign this Processor also to input 2, and use push buttons #12 and #13 to enter "02" under the legend CHAN BUMP.

Then select the third Processor's master page by pushing the MODE button until the symbol for the second Processor is displayed. (Holding in MODE and pressing #13 achieves the same result.) Assign this Processor also to input 2, and use push buttons #12 and #13 to enter "03" under the legend CHAN BUMP.

Then return to the Assignment Mode. Either tap WRITE once and then MODE once, or hold MODE and press #2.

Place a "2" above button #5. Place the symbol for the first Processor above button #6.
Place the symbol for the second Processor above button #7.
Place the symbol for the third Processor above button #8.

To Rescale the Velocity of MIDI Notes
On any Processor master page, use push buttons #14 through #19 under the legend VELOCITY MINIMUM and VELOCITY MAXIMUM. Any MIDI Note data passing through the Processor will have its velocity values changed to comply with the user-selected range of 001 through 127. This affects data after it has passed through the Data and Channel Filters, and the Channel Bump.

If a VELOCITY MINIMUM of 000 is selected, the Synapse Processor constructs a special "V-squared" velocity curve. (The MAXIMUM has no effect). The MIDI velocities passing through the Processor are kept fairly low until a threshold is exceeded, at which point the velocity output jumps to 127. This is useful if you are controlling a tone module which brings up a different sound whenever a certain velocity is exceeded.

For example, say that the MIDI output of a keyboard goes into the Synapse MIDI input number 13 (D). Synapse MIDI output number 1 goes to a tone generator.

Every time you play the keyboard, you notice that the sound coming from the module is much brighter than you remember it. If you try to play gently, the sound is inaudible. You had first designed that sound and tried it out using a different brand of keyboard controller. You want to rescale the velocity from the keyboard to compress the dynamic range so that the sound give satisfactory results and predictable tracking of your performance.

Solution:
You will need to assign a Processor to input D. Then enter a VELOCITY MINIMUM of 32, as an experiment. Enter a VELOCITY MAXIMUM of, say, 95. Then return to the Assignment Mode and place the symbol for the processor above button #1. Because the technique is so similar to the previous examples, this is left as an exercise for the reader.

To Use the Zone Functions
After MIDI data passes through all of the Processor's master page functions, it goes into four Zones. A Zone is defined as a range of notes by specifying a Low Note and a High Note. Any MIDI notes that fall within the user-selected range will be processed with either a Channel Bump and/or a Transposition.

It is not manditory that you use any or all of the Zones. Simply setting the Low Note and the High Note both equal to zero will defeat any Zone.

Likewise, if you wish to have a Transposition or a Channel Bump that affects every note across the range of the keyboard, the Low Note should be set to 000 and the High Note to 127. In that case the Zones do not cause any keyboard splitting, they are simply used to change channels and transpositions.

To Zone Transpose MIDI Data
MIDI notes coming into a Processor may be transposed plus or minus five octaves (±59 semitones) in semitone steps. Here is a simple example of using one Zone of one Processor to effect a transposition. Suppose you have a keyboard that is hooked up to Synapse MIDI input 1. Suppose also that Synapse MIDI outputs #2 and #3 go to two other synthesizers. When you play the keyboard, you want the synthesizer on output #3 to play one octave higher.

Solution:
Assign the first Processor to input 1, using the technique you learned in the previous examples. Then push the MODE button once to go to Zone 1 of the first processor. See figure 11, next page.

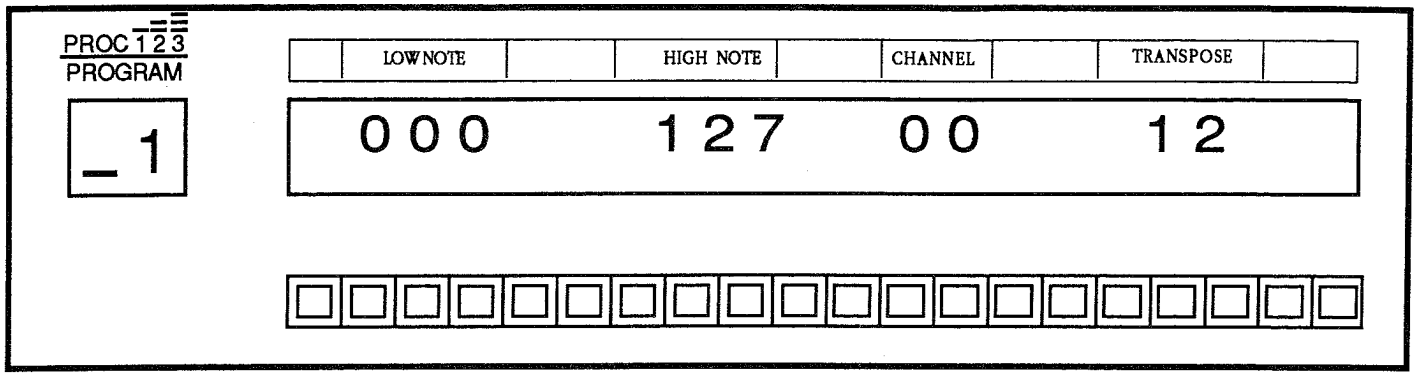


figure 11 The first processor, Zone 1

Since we want the transposition to affect the whole range of the keyboard, we will leave the low and high note assignments unchanged. Enter a transposition interval of 12 semitones.

Then return to the Assignment Mode. Either tap WRITE once and then MODE once, or hold MODE and press #2.

Place a "1" above button #2. This routes the keyboard directly to the first synthesizer. Place the symbol for the first Processor above button #3. This routes the processed keyboard to output 3, which will now sound one octave higher than synth 2.

To Zone Channel Bump Data

Here is an example of using two Zones of one Processor to effect a channel bump. Suppose you have a Yamaha DX-7 that is hooked up to Synapse MIDI input 1. Older DX-7's, unless they have been retro-fitted with special software, only send on MIDI channel 1. Suppose also that Synapse MIDI outputs #1, #2 and #3 go to three tone modules. The tone modules are on MIDI channels 1, 2, and 3, respectively. When you play the keyboard, you want it to control all three modules. To do this, we need to make the DX-7 send on channels 1, 2, and 3 all at once.

Solution:
Assign the first Processor to input 1, using the technique you learned in the previous examples. Then push the MODE button once to go to Zone 1 of the first processor. See figure 12.

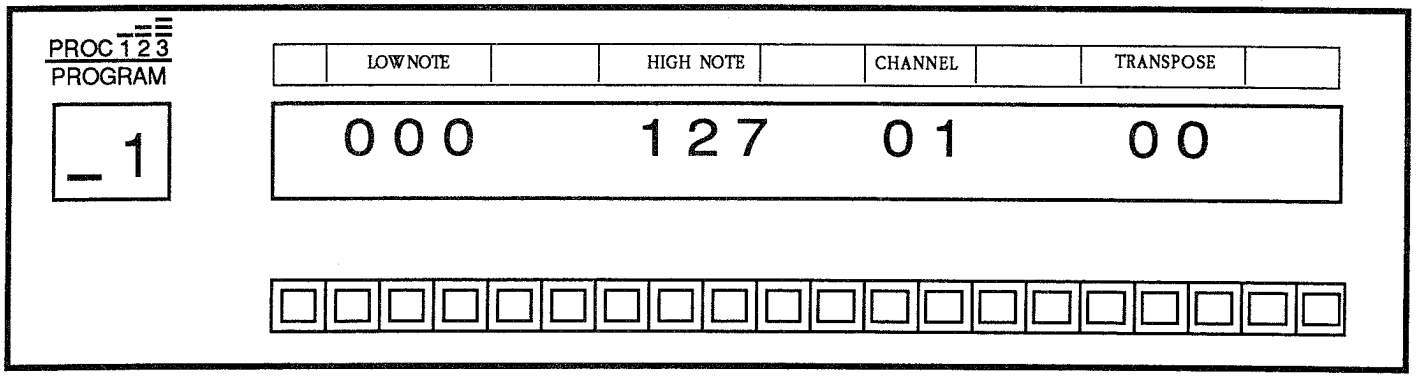


figure 12 The first processor, Zone 1

Enter a Channel Bump of 01 underneath the legend CHANNEL. Any notes passing through this Zone will have 1 added to their channel number.

Push the MODE button again to go to Zone 2 of the first processor. See figure 13.

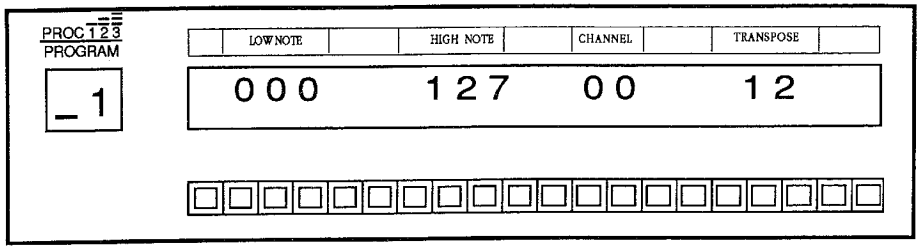


figure 11 The first processor, Zone 1

Since we want the transposition to affect the whole range of the keyboard, we will leave the low and high note assignments unchanged. Enter a transposition interval of 12 semitones.

Then return to the Assignment Mode. Either tap WRITE once and then MODE once, or hold MODE and press #2.

Place a "1" above button #2. This routes the keyboard directly to the first synthesizer. Place the symbol for the first Processor above button #3. This routes the processed keyboard to output 3, which will now sound one octave higher than synth 2.

To Zone Channel Bump Data
Here is an example of using two Zones of one Processor to effect a channel bump. Suppose you have a Yamaha DX-7 that is hooked up to Synapse MIDI input 1. Older DX-7's, unless they have been retro-fitted with special software, only send on MIDI channel 1. Suppose also that Synapse MIDI outputs #1, #2 and #3 go to three tone modules. The tone modules are on MIDI channels 1,.2, and 3, respectively. When you play the keyboard, you want it to control all three modules. To do this, we need to make the DX-7 send on channels 1, 2, and 3 all at once.

Solution:
Assign the first Processor to input 1, using the technique you learned in the previous examples. Then push the MODE button once to go to Zone 1 of the first processor. See figure 12.

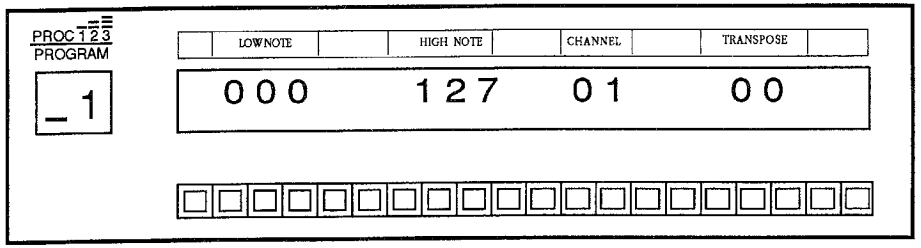


figure 12 The first processor, Zone 1

Enter a Channel Bump of 01 underneath the legend CHANNEL. Any notes passing through this Zone will have 1 added to their channel number.

Push the MODE button again to go to Zone 2 of the first processor. See figure 13.

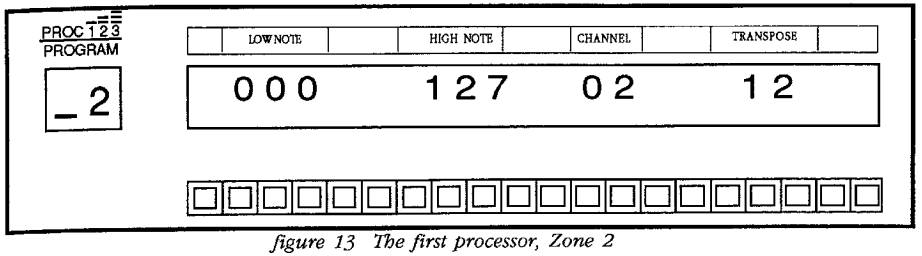


figure 13 The first processor, Zone 2

You will notice that this Zone is turned off when you first select it. The LOW NOTE and the HIGH NOTE are both set equal to 000. Enter a HIGH NOTE of 127, so that the channel bump will affect the entire range of the keyboard.

Enter a Channel Bump of 02 underneath the legend CHANNEL. Any notes passing through this Zone will have 2 added to their channel number.

Now any notes coming into the first Processor on MIDI channel 1 will come out of the Processor on MIDI channels 2 and 3 at the same time.

Then return to the Assignment Mode. Either tap WRITE once and then MODE once, or hold MODE and press #2.

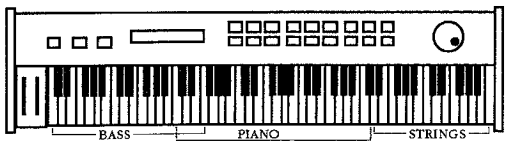
Place a "1" above button #1. This routes the DX-7 directly to the first module on MIDI channel 1. Place the symbol for the first Processor above buttons #2 and #3. This routes the processed DX-7 to outputs 2 and 3. Now when you play the DX-7, all three modules will sound.

This example may be easily expanded. You have two more Zones to work with in the first Processor. And two more Processors to work with. If all the Processors were assigned to the same input number, and all the Zones were in use, a single keyboard could send MIDI data on thirteen channels simultaneously. And each channel could have a different transposition interval.

Keyboard Splits
In the previous example, channel bumping and transposition were applied across the range of the keyboard. In this example, several overlapping zones will be set up on one keyboard controller.

Important Note: On most keyboards, the lowest MIDI note is 036, not 000.

Suppose that we have a keyboard controller that only sends on MIDI channel 1. Say that we also have several tone modules that are set up to play a Bass sound when they receive MIDI notes on channel 1, Piano on channel 4, and Strings on channel 8. We want to set up three Zones so that the lower range of the keyboard plays the Bass. The middle range plays the Piano but overlaps the Bass. The top range of the keyboard plays the strings transposed down one octave.



Let us further assume for this example that when we move the mod wheel on our keyboard controller, the modulation data should only affect the String sound on channel 8.

The keyboard controller goes into Synapse input 3, the tone modules are on outputs #8, #9, and #10. We will arbitrarily choose the second Processor for this example.

First select the second Processor master page, using the MODE button. Assign the input number using button #1. Refer to figure 15.

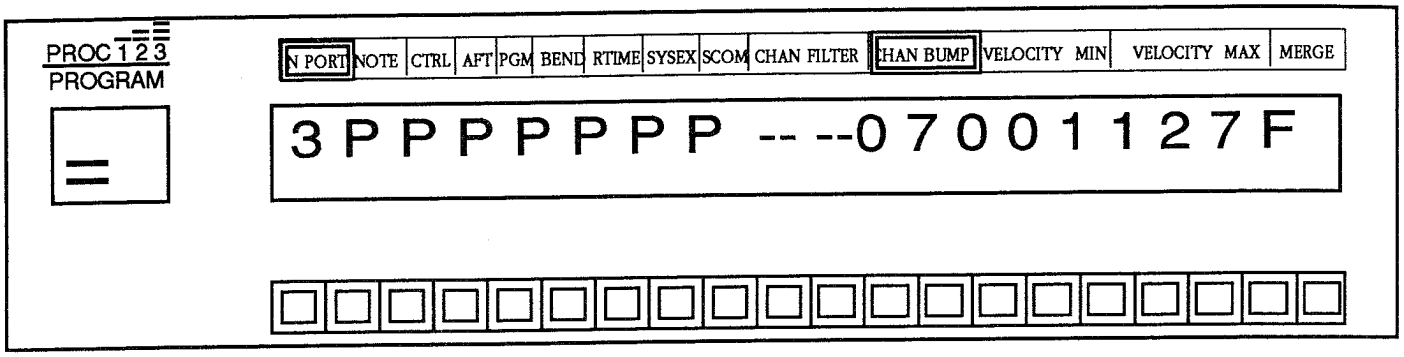


figure 15 The second Processor, master page. Set the input and controller channel bump.

Remember that the master page Channel Bump does not affect notes. We enter a 07, so that when the modulation wheel is moved on the controller, the data passes through the Processor and comes out on channel 8.

Then push the MODE button once to go to Zone 1 of the second Processor. Zone 1 will be our Bass on channel 1, so we need not enter any Channel Bump or Transposition. We enter a LOW NOTE of 36, and a HIGH NOTE of 65. This gives us a range of two octaves plus one fourth. . See figure 16.

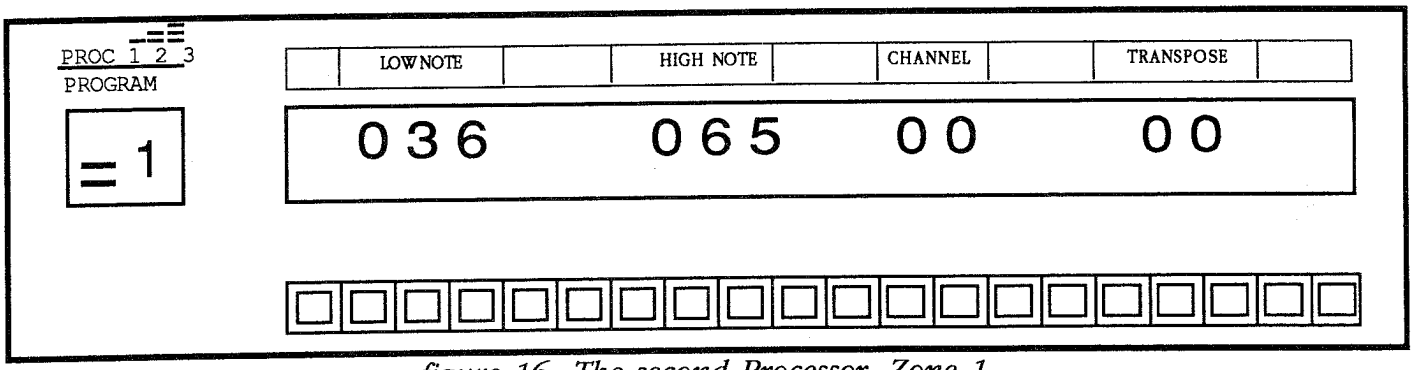


figure 16 The second Processor, Zone 1

Push the MODE button once to go to Zone 2 of the second Processor. Our Piano is on channel 4, so we enter a Channel Bump of 3. (That is the number added to the incoming channel number.) No Transposition is needed. We want a three octave range starting and ending on C. The LOW NOTE equals 60, the HIGH NOTE equals 96. This will overlap the range of the Bass. Refer to figure 17.

Let us further assume for this example that when we move the mod wheel on our keyboard controller, the modulation data should only affect the String sound on channel 8.

The keyboard controller goes into Synapse input 3, the tone modules are on outputs #8, #9, and #10. We will arbitrarily choose the second Processor for this example.

First select the second Processor master page, using the MODE button. Assign the input number using button #1. Refer to figure 15.

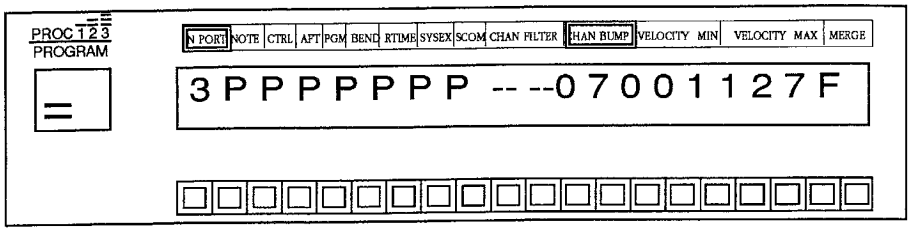


figure 15 The second Processor, master page. Set the input and controller channel bump.

Remember that the master page Channel Bump does not affect notes. We enter a 07, so that when the modulation wheel is moved on the controller, the data passes through the Processor and comes out on channel 8.

Then push the MODE button once to go to Zone 1 of the second Processor. Zone 1 will be our Bass on channel 1, so we need not enter any Channel Bump or Transposition. We enter a LOW NOTE of 36, and a HIGH NOTE of 65. This gives us a range of two octaves plus one fourth. . See figure 16.

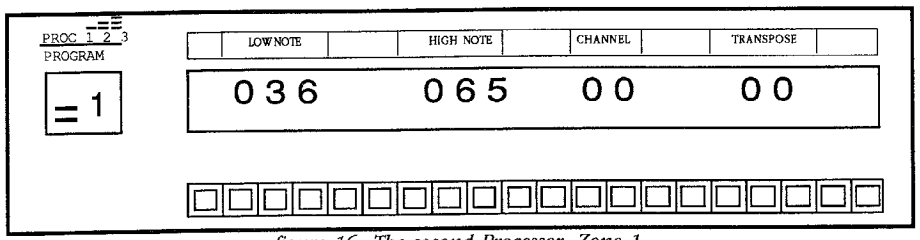


figure 16 The second Processor, Zone 1

Push the MODE button once to go to Zone 2 of the second Processor. Our Piano is on channel 4, so we enter a Channel Bump of 3. (That is the number added to the incoming channel number.) No Transposition is needed. We want a three octave range starting and ending on C. The LOW NOTE equals 60, the HIGH NOTE equals 96. This will overlap the range of the Bass. Refer to figure 17.

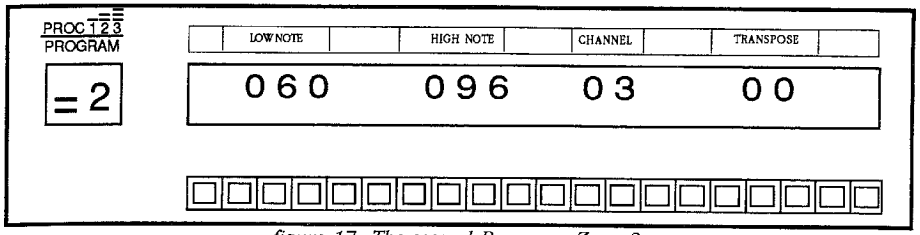


figure 17 The second Processor, Zone 2

Push the MODE button once again to select ZONE 3 of the second Processor.
The Strings are on channel 8, so we want to bump the channel by 7.
To transpose down one octave, enter a -12 under the legend TRANSPOSE.
The LOW NOTE equals 97, and the HIGH NOTE equals 127. Refer to figure 18.

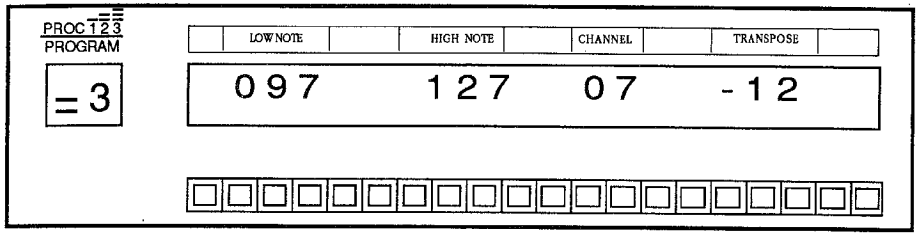


figure 18 The second Processor, Zone 3

Now return to the Assignment Mode. Either tap WRITE once and then MODE once, or hold in MODE and press button #2.

Assign the second Processor to outputs #8, #9, and #10. Refer to figure 19.

You may now write this assignment to some program number, say 50.
Hold in the WRITE button. While holding in WRITE, press button #5, then #10.

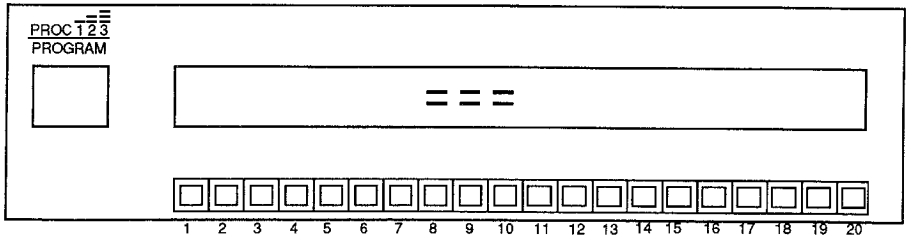


figure 19 The Assignment Mode.

MIDI MERGING

The Synapse MIDI Merger combines the outputs of any selected Processors. This permits up to three input merging, or two input merging leaving a Processor free for some other purpose.

Merging is used whenever you need to send two or three sources of MIDI data at once to a given destination.

For example, most MIDI sequencers only have one input. Merging allows up to three keyboardists to record into a sequencer at the same time. Or perhaps the sequencer is receiving MIDI Time Code from a synchronizer. Keyboard data may be merged with MTC and both sent into the sequencer on the same MIDI cable. (Some synchronizers, like the JLCooper PPS-100, already have a merge feature built in.)

Also, tone modules usually have only one MIDI input. Merging allows both a keyboard controller and a computer to simultaneously control a module or modules. This permits immediate auditing of sounds loaded in by a librarian software package. Or a multi-timbrel module can be played by two keyboards at once, or a keyboard and a sequencer.

Although most sequencers feature a "soft-thru" or an "echo-thru" function, using the Synapse Merge is usually more efficient. The sequencer's computer will be less burdened, and you will have more control of the destination of the merged data. Also, MIDI data can be first processed by the three MIDI Data Processors *before* being merged.

The procedure for using the Merger is as follows:
First assign the Processors to the inputs of the Synapse that correspond to the instruments that you want to merge. Use the master page for each Processor.

Above button #20 is a "P" or an "F".
"P" means "Pass the output of this Processor to the Merger."
"F" means "Filter the output of this Processor to prevent it from going into the Merger."

Return to Assignment Mode. For any output that you want to send the merger to, scroll until the U symbol shows. You can think of "U" as symbolizing the Union of the Processors. (Or perhaps it symbolizes the Union of the emotions with the intellect. Or perhaps we are just making excuses that there is no room in the display for a letter "M".)

For example, say that we have two keyboards and a wind controller on Synapse inputs 3, 4, and 5. A computer sequencer is hooked up to Synapse input and output 1. Output 9 goes to a sampler, and outputs 10 through 20 go to various tone modules.

We want to create an assignment on Synapse that uses the Merger, so that the three controllers can all load the sequencer. We also want to send the merged data to the modules on outputs 15 through 20, so we can hear what we are playing. We want the wind controller only to drive the sampler. While recording on the sequencer, some pre-recorded sequencer tracks will be playing back. We want those tracks to control the modules on outputs 10 through 14.

First we must assign the three Processors to the three controllers that we want to merge.

Select the first Processor's master page, either by pushing the MODE button repeatedly or by holding in MODE and pressing #3.

Then assign it to the first keyboard controller on input 3.

Turn "Pass to Merger" on by pressing button #20 to change the "F' to a "P". See figure 1.

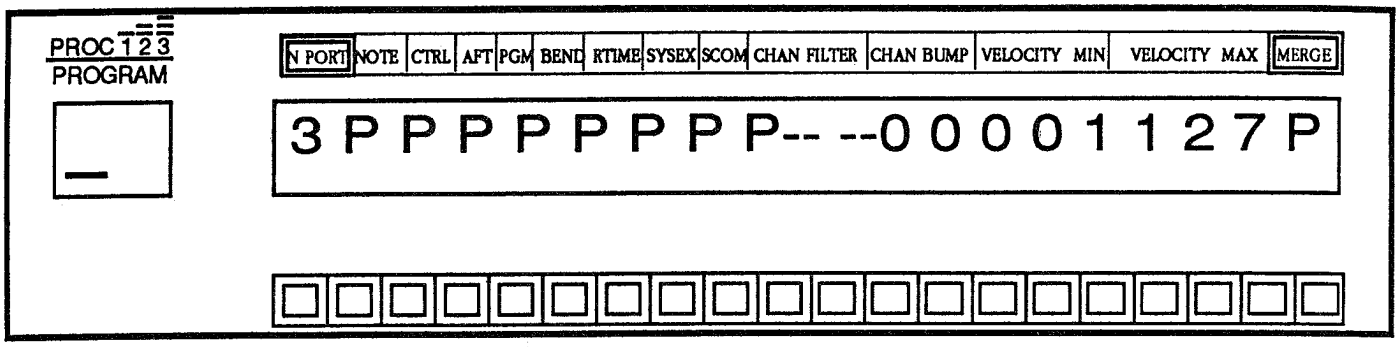


figure 1 The first Processor, master page. Select the input and set Pass to merge.

Now select the second Processor's master page, either by pushing the MODE button repeatedly or by holding in MODE and pressing #8.

Then assign it to the second keyboard controller on input 4.

Turn "Pass to Merger" on by pressing button #20 to change the "F' to a "P". See figure 2.

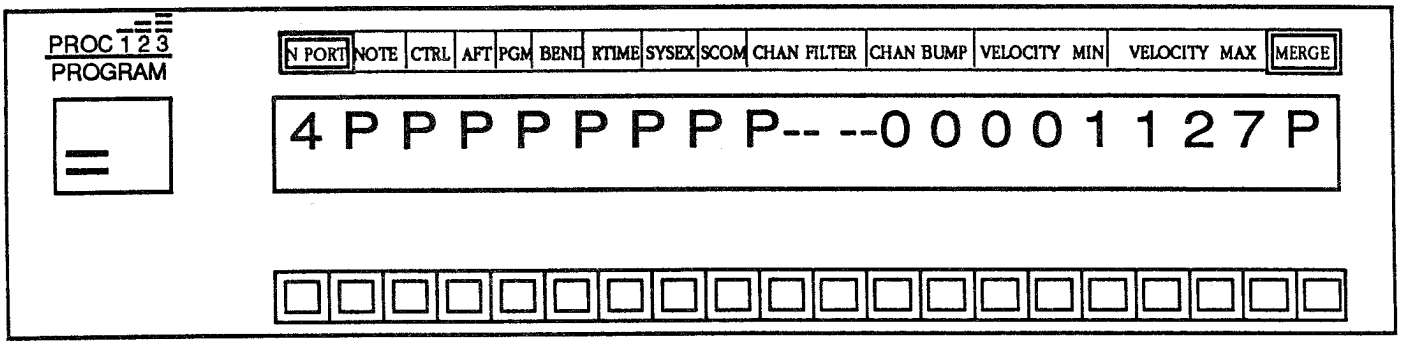


figure 2 The second Processor, master page. Select the input and set Pass to merge.

Then select the third Processor's master page, either by pushing the MODE button repeatedly or by holding in MODE and pressing #13.

Then assign it to the wind controller on input 5.

Turn "Pass to Merger" on by pressing button #20 to change the "F' to a "P". See figure 3.

First we must assign the three Processors to the three controllers that we want to merge.

Select the first Processor's master page, either by pushing the MODE button repeatedly or by holding in MODE and pressing #3.

Then assign it to the first keyboard controller on input 3.

Turn "Pass to Merger" on by pressing button #20 to change the "F' to a "P". See figure 1.

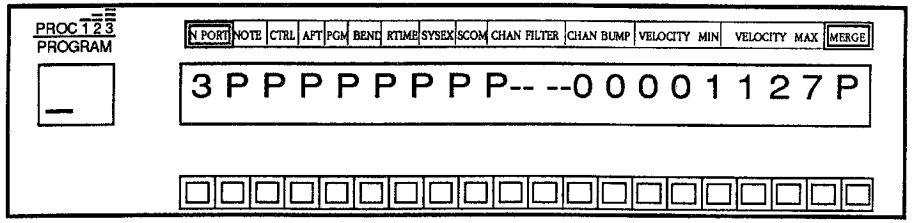


figure 1 The first Processor, master page. Select the input and set Pass to merge.

Now select the second Processor's master page, either by pushing the MODE button repeatedly or by holding in MODE and pressing #8.

Then assign it to the second keyboard controller on input 4.

Turn "Pass to Merger" on by pressing button #20 to change the "F' to a "P". See figure 2.

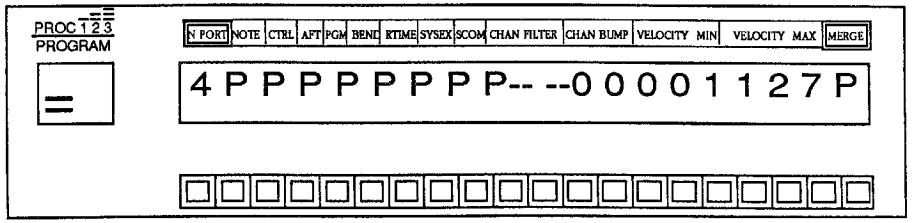


figure 2 The second Processor, master page. Select the input and set Pass to merge.

Then select the third Processor's master page, either by pushing the MODE button repeatedly or by holding in MODE and pressing #13.

Then assign it to the wind controller on input 5.

Turn "Pass to Merger" on by pressing button #20 to change the "F' to a "P". See figure 3.

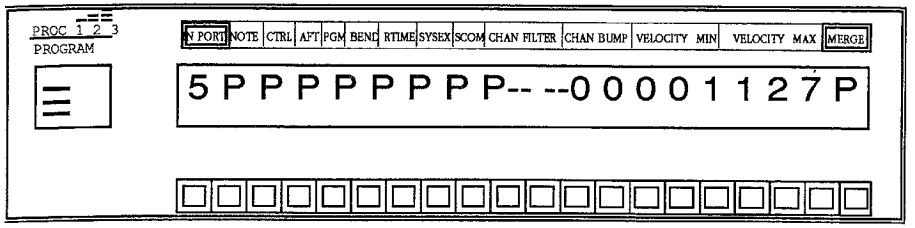


figure 3 The third Processor, master page. Select the input and set Pass to merge.

Now return Synapse to the Assignment Mode. Either tap WRITE once, and then MODE once, or hold in MODE and while holding push button #2.

In this example, we want to send the merged MIDI data to the sequencer on output #1, and the modules on outputs #15 through #20.

Place a "U" above buttons #1, and #15 through #20 by holding each button until it scrolls to "U". See figure 4.

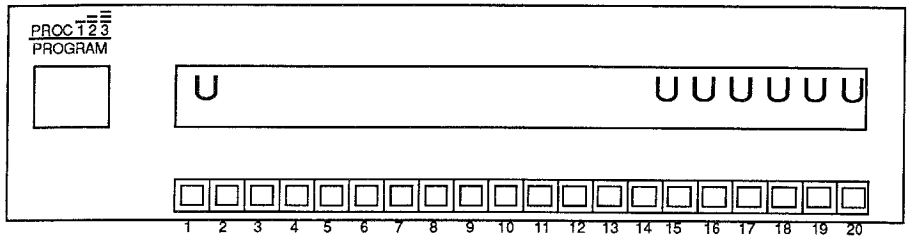


figure 4 The Assignment Mode. "U" means that the merger is being sent to those Synapse outputs.

Now we will complete the assignments as specified in this example.

The sampler on output #9 is to be controlled only by the wind controller. Push button #9 until a 5 is displayed.

Lastly, assign outputs #10 through #14 to the sequencer which is on input 1. See figure 5.

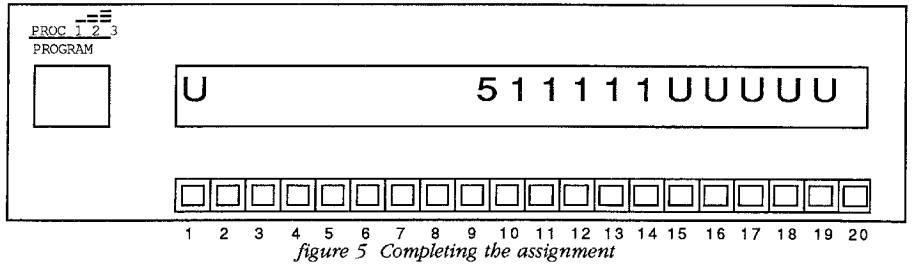


figure 5 Completing the assignment

Please Note
MIDI data merging is a complicated and difficult process. It involves a finite amount of computer processing time. The Synapse, however, is the fastest and most sophisticated merger available. Under moderate loads of data, no noticable delays will be introduced. But if it is not necessary to use the merger, leave it turned off by selecting the Processor master pages, and entering an "F" to filter the Processor from the merger.

For optimal Processor performance, leave the MERGE display set to "F" whenever a Processor is not being Merged.

Also keep in mind that MIDI is serial, and does have a finite bandwidth. The limitations of the MIDI bandwidth begin to become apparent when heavy loads of MIDI data are merged.

For example, if a MIDI guitar controller sending on five channels at once is merged with a 16-track sequence and a MIDI Volume Controller (like the J. L. Cooper FaderMaster), the MIDI bandwidth will likely be pressed to its limits.

A given instrument on the receiving end may show a "MIDI Buffer Full" warning, and note could be significantly delayed.

The Synapse merger is very fast and efficient, as it allows MIDI setups never before possible. But please keep your expectations within reasonable limits regarding just how much data can be sent on a given MIDI cable. Since we cannot offer any recommendations in this area, you may need to experiment to discover how much data is too much.

Please Note
MIDI data merging is a complicated and difficult process. It involves a finite amount of computer processing time. The Synapse, however, is the fastest and most sophisticated merger available. Under moderate loads of data, no noticable delays will be introduced. But if it is not necessary to use the merger, leave it turned off by selecting the Processor master pages, and entering an "F" to filter the Processor from the merger.

For optimal Processor performance, leave the MERGE display set to "F" whenever a Processor is not being Merged.

Also keep in mind that MIDI is serial, and does have a finite bandwidth. The limitations of the MIDI bandwidth begin to become apparent when heavy loads of MIDI data are merged.

For example, if a MIDI guitar controller sending on five channels at once is merged with a 16-track sequence and a MIDI Volume Controller (like the J. L. Cooper FaderMaster), the MIDI bandwidth will likely be pressed to its limits.

A given instrument on the receiving end may show a "MIDI Buffer Full" warning, and note could be significantly delayed.

The Synapse merger is very fast and efficient, as it allows MIDI setups never before possible. But please keep your expectations within reasonable limits regarding just how much data can be sent on a given MIDI cable. Since we cannot offer any recommendations in this area, you may need to experiment to discover how much data is too much.

PATCH MAP

The Patch Map feature allows the Synapse to store and send out sets of MIDI Program Change commands. Whenever a pre-programmed assignment is recalled from memory, (either from the front panel, or remotely via MIDI), the Synapse can send out up to 16 MIDI Program Change commands on any combination of outputs. This set of 16 patch changes is unique to each program assignment. Since the Synapse stores 64 program assignments, that means that the Synapse can hold up to 1024 program changes.

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 synth 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.

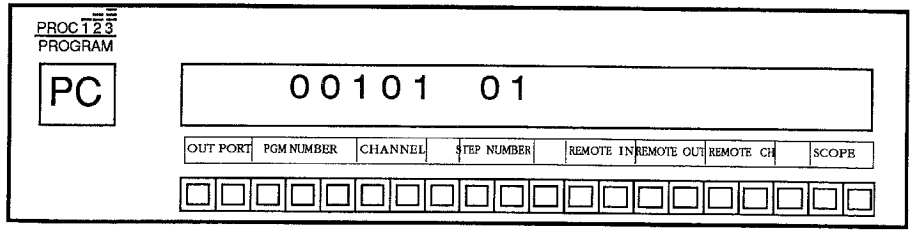
With Synapse's Patch Map, all you have to do is recall a single program on the Synapse. Pre-programmed sets of patch changes go out to all the slaves so that they are perfectly set up for the next song. Instead of all the slaves going to "5", one can go to "13", another to "125", another to "7", or whatever.

To load in a Patch Map, select the Program Change page.

To Select to the Program Change page, do the following:

Hold down MODE and press button #18 or

The display will look like this.



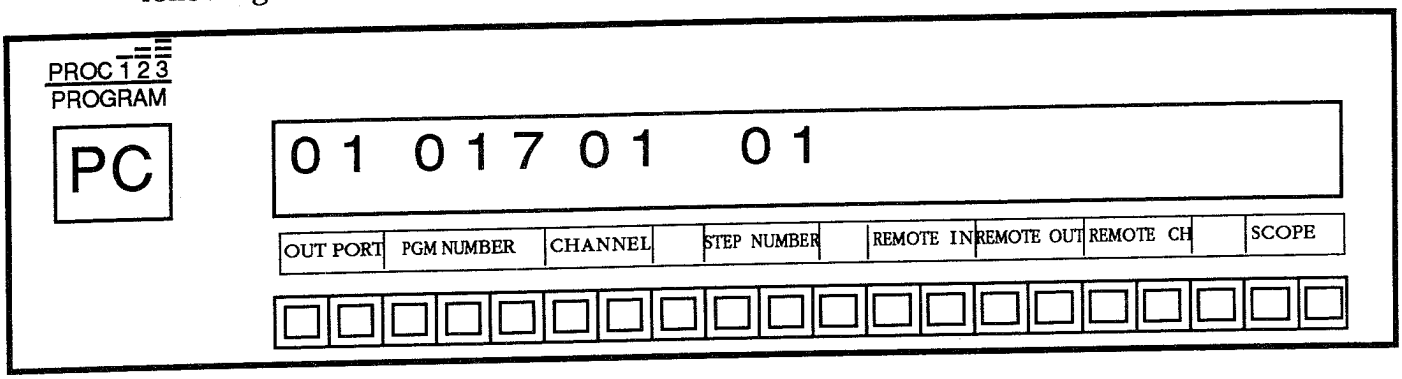
Refer to the legend below the display. We are only interested in the numbers above the first ten buttons. (The rest of the display concerns the Remote and Scope functions to be covered in later chapters.)

You will be entering which **Output Port** (slave device) will get the Program Change command or string of commands. Use buttons 1 and 2 labeled OUT PORT.

To enter the **Program Change** number to be sent use buttons 3, 4, and 5 marked PGM Number.

To enter the **MIDI channel** of the Program Change, use buttons 6 and 7 marked CHANNEL.

For example, say you want the Synapse to be able to tell a keyboard on output #1 to go to patch 17. The slave is set to receive on MIDI channel 1. On the PC page, you would enter the following.



Now write this to a memory location, say "23" for example. Hold WRITE and press 2 and 3. As you write it to memory the Synapse will actually send the program changes out, to allow you to verify the results.

This particular change will now be sent out of the Synapse every time you recall program 23.

Up to sixteen Steps can be entered, each step being a Program Change.
The **STEP NUMBER** is changed with buttons 9 and 10.

For example, if you want to enter a series of Program Change Commands, enter the OUT PORT, PGM NUMBER, CHANNEL, and then increment the STEP NUMBER. Repeat the process for each Program Change.

You may enter multiple Program Changes on only one output or combinations of outputs if you chose. For example, Synapse output #5 could have eight Program Changes going out on eight different MIDI channels. This may be used to set up the voices of a multi-timbrel module.

A Patch Map may be modified by removing and replacing individual Program Changes. To remove a single Program Change from the Patch Map, select the STEP NUMBER of the offending command.

Hold buttons #1 and #8 to remove a single Program Change Command.

You may or may not choose to re-enter a new Program Change. The now modified Patch Map should be written into the memory.

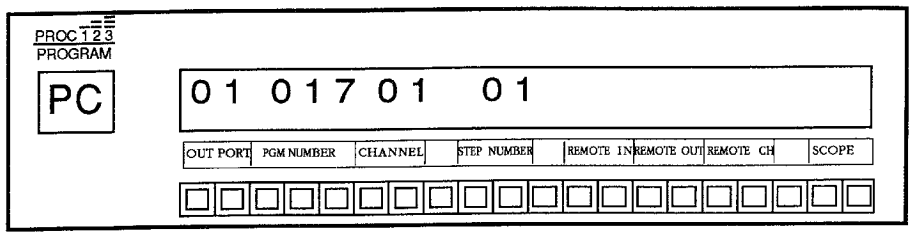
Please Note:
Every time you recall a Synapse assignment from memory (while in Program Mode), the Patch Map will be sent out. If you use buttons #19 and #20 to scroll up or down through Synapse's program memory, this will result in a very high volume of Program Change commands being sent out. This could cause a receiving device to indicate "buffer full".

You will be entering which **Output Port** (slave device) will get the Program Change command or string of commands. Use buttons 1 and 2 labeled OUT PORT.

To enter the **Program Change** number to be sent use buttons 3, 4, and 5 marked PGM Number.

To enter the **MIDI channel** of the Program Change, use buttons 6 and 7 marked CHANNEL.

For example, say you want the Synapse to be able to tell a keyboard on output #1 to go to patch 17. The slave is set to receive on MIDI channel 1. On the PC page, you would enter the following.



Now write this to a memory location, say "23" for example. Hold WRITE and press 2 and 3. As you write it to memory the Synapse will actually send the program changes out, to allow you to verify the results.

This particular change will now be sent out of the Synapse every time you recall program 23.

Up to sixteen Steps can be entered, each step being a Program Change.
The **STEP NUMBER** is changed with buttons 9 and 10.

For example, if you want to enter a series of Program Change Commands, enter the OUT PORT, PGM NUMBER, CHANNEL, and then increment the STEP NUMBER. Repeat the process for each Program Change.

You may enter multiple Program Changes on only one output or combinations of outputs if you chose. For example, Synapse output #5 could have eight Program Changes going out on eight different MIDI channels. This may be used to set up the voices of a multi-timbrel module.

A Patch Map may be modified by removing and replacing individual Program Changes. To remove a single Program Change from the Patch Map, select the STEP NUMBER of the offending command.

Hold buttons #1 and #8 to remove a single Program Change Command.

You may or may not choose to re-enter a new Program Change. The now modified Patch Map should be written into the memory.

Please Note:
Every time you recall a Synapse assignment from memory (while in Program Mode), the Patch Map will be sent out. If you use buttons #19 and #20 to scroll up or down through Synapse's program memory, this will result in a very high volume of Program Change commands being sent out. This could cause a receiving device to indicate "buffer full".

PANIC

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 changing a Processor assignment while a sequence is passing through. But there are definitely times when there seems to be no explanation at all, other than "that happens every once in awhile". AC noise may be the culprit, or a new piece of software with a bug. 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 function on Synapse.

When activated, the Panic function sends MIDI commands out of all twenty output ports telling all slaves to "unstick".

The Panic function is initiated by holding the WRITE and MODE buttons at the same time. The display will scroll the word "PAnic" from left to right.

The Synapse will send a lot of commands, trying just about everything to unstick that note. It will take about 2 seconds for all of these commands to go out, but in general, it is best to release the Panic function as soon as you hear the note unstick. Due to the high volume of MIDI messages, some MIDI slave devices may display a "Buffer Full" warning. This should cause no problem, but should noneless be avoided by releasing the Panic as soon as possible.

In our lab tests, however, notes usually unstick themselves in less than a second. It really depends on the receiving instrument. Some respond to All Notes Off commands, which the Synapse sends out first. These are followed by Sustain Pedal Up commands, which may help in some cases. After these commands are sent, individual Note Off commands are sent out starting at Note 1 on MIDI Channel 1, and continuing on up to Note 127 on MIDI Channel 16.

REMOTE OPERATION

The Synapse can be set to respond to MIDI Program Change commands. The reception and recognition of a Program Change command causes Synapse to recall a previously programmed setup of routing, processing and merging. Synapse will also respond by sending out a Patch Map of up to sixteen pre-programmed Program Change commands. With one touch of a button on your keyboard controller you can instantly re-wire your whole MIDI studio!

You can select which Synapse Input Port and MIDI Channel will respond to the remote Program Change command. This must be carefully chosen, since in any MIDI system program changes could come in on any input at any time. You certainly would not want one to accidentaly be recognized by the Synapse.

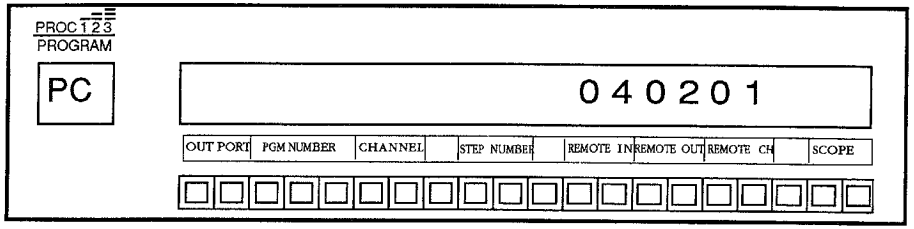
To recall the input and MIDI channel that the Synapse will remotely respond to, select the Program Change page.

To Select the Program Change page, do the following:

Hold down MODE and press button #18

Both of these methods achieve the same results. (In the first case, you are directly selecting the 18th page. In the second case, you are back stepping to the 18th page.)

Observe the right half of the display.



The input assignment is displayed above the legend "REMOTE IN" (buttons 12 and 13). The MIDI receive channel is displayed above the legend "REMOTE CH" (buttons 16 and 17).

In the above illustration, when a Program Change command on MIDI Channel 1 comes into the Synapse Input Port 4, the Synapse will respond by changing to that program number.

To change the assigned Remote Input and Channel, simply tap the buttons below the display.

If you do not want the Synapse to respond to remote program changes, simply set the Remote Input to blank.

The Remote Assignment is global, that is, there is only one remote assignment for Synapse regardless of the currently selected program number.

The Remote Assignment is automatically saved to memory, there is no need to perform a WRITE operation.

WARNING
Never send a remote Program Change while data is flowing through the Synapse. This would be analogous to pulling out and quickly rearranging 36 MIDI cables while trying to send MIDI data. If this is done accidentaly, you may need to use the PANIC function.

Combining Remote Operation with Patch Map.
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 Synapse, and have Synapse in turn send out a Patch Map. This can be done, of course, but it will require a bit of extra planning on your part.

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

It is strongly recommended, however, that you dedicate a MIDI channel to Synapse if you plan on using Remote operation. That way, a program change intended for Synapse will not directly affect a slave device. And a program change intended for a tone generator will not re-wire your studio.

The Remote Assignment is global, that is, there is only one remote assignment for Synapse regardless of the currently selected program number.

The Remote Assignment is automatically saved to memory, there is no need to perform a WRITE operation.

WARNING
Never send a remote Program Change while data is flowing through the Synapse. This would be analogous to pulling out and quickly rearranging 36 MIDI cables while trying to send MIDI data. If this is done accidentaly, you may need to use the PANIC function.

Combining Remote Operation with Patch Map.
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 Synapse, and have Synapse in turn send out a Patch Map. This can be done, of course, but it will require a bit of extra planning on your part.

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

It is strongly recommended, however, that you dedicate a MIDI channel to Synapse if you plan on using Remote operation. That way, a program change intended for Synapse will not directly affect a slave device. And a program change intended for a tone generator will not re-wire your studio.

MIDI Time Code Display

Whenever MTC comes into the user-selected "Remote Input", Synapse provides a large, easy to read display of Hours, Minutes, Seconds and Frames. The incoming SMPTE frame rate is also shown. (30, 30d, 24 or 25 frames per second).

The "Remote Input" is selected on the 18th page of Synapse. Press the MODE button, and while holding down MODE, Press button #18. Buttons #12 and #13 (below the front panel legend "Remote In") are used to enter a number between 1 and 16. This number indicates which of Synapse's 16 MIDI inputs will receive MTC.

MTC is displayed on the "19th page" of Synapse. Press the MODE button, and while holding down MODE, Press button #19 to display MTC.

LINKING SEVERAL SYNAPSES

The Synapse Expansion Bus permits joining several Synapses to form a 16 input system with 40, 60, or 80 outputs. Theoretically speaking, up to 12 Synapses could be chained. Chaining Synapses provides more outputs, not more inputs. However, in certain specialized and limited applications, the extra inputs could be used as well.

The MIDI inputs of the first Synapse are split and sent to all Synapse linked via a special expansion cable available from JLCooper.

The same input designations 1 through 9, and A through G, apply to all Synapses in the chain. Say for example there is a keyboard controller coming into input 10 (A) on the first Synapse. Suppose that you wish to route that controller to 60 destinations on three chained Synapses. In that case, an "A" would be displayed across all outputs on all the Synapses.

Do not use the inputs of the other units, noting the specific exceptions below.

You can make a chained Synapse change programs when a program is selected on the first Synapse. Hook up a MIDI cable from an output of of the first Synapse to an input of the second Synapse. Then use the Patch Map on the first Synapse, to send a Program Change to the output that corresponds to that MIDI cable. On the second Synapse, use the Remote function, entering the input number that corresponds to that MIDI cable.

Also, any unused inputs on the first unit may be used on the second unit. For example, if inputs 14 through 16 are not being used on the first Synapse, they can be used on the second Synapse so you have access to a total of six MIDI Data Processors.

If you attempt to use corresponding inputs on more than one Synapse, you have created a a false merger, also known as a MIDI "Y-cord". This is because both sets of inputs are in parallel. You could leave instruments hooked up to both sets of inputs **provided that they are not both sending data at the same time.** But be careful, some instruments constantly send data. For example, the Yamaha DX-7 always sends Active Sensing, and most sequencers always send MIDI clock, even if they are not playing back. If two corresponding inputs are receiving MIDI data at the same time you will run into massive data errors. If you observe the these limitations, it is possible to leave two controllers hooked up to the same input numbers on both Synapses, allowing you to move from one controller to another without having to recall a new assignment.

SCOPE

The SCOPE is a MIDI diagnostic tool that allows you to view the last MIDI Status Byte coming into a specific Synapse input. The input is the same as the input selected for Remote operations, that is, the same input that Synapse is set to receive Program Change Commands on.

SCOPE is displayed on the Program Change page above buttons 19 and 20.
The Program Change Page can be quickly selected in either one of two different ways.

To Select to the Program Change page, do either of the following:

(1) Hold down MODE and press button #18 or

(2) Tap WRITE once, and then hold in MODE and button #1.

Both of these methods achieve the same results. (In the first case, you are directly selecting the 18th page. In the second case, you are back stepping to the 18th page.)

The input being monitored by SCOPE is selected with buttons 12 and 13.

MIDI Status Bytes are displayed in two-digit hexidecimal notation. The following table shows what the display means.

8 n Note Off
9 n Note On
A n Polyphonic After Touch
B n Control Change
C n Program Change
D n Channel Pressure (After Touch)
E n Pitch bend

These are all Channel Voice messages, where "n" equals the MIDI channel according to the following list:

n	channel
0 =	1
1 =	2
2 =	3
3 =	4
4 =	5
5 =	6
6 =	7
7 =	8
8 =	9
9 =	10
A =	11
B =	12
C =	13
D =	14
E =	15
F =	16

(Continued on next page.)

F 0 through F 7 are System Common messages:
F 0 = System Exclusive
F 1 = MIDI Time Code Quarter Frame
F 2 = Song Position Pointer
F 3 = Song Select
F 4 = Undefined
F 5 = Undefined
F 6 = Tune Request
F 7 = EOX: End of System Exclusive

F 8 through F F are System Real Time messages:
F 8 = TIMING clock
F 9 = Undefined
F A= Start
F B= Continue
F C= Stop
F D= Undefined
F E = Active Sensing **Note that this is not displayed by Scope.**
F F= System Reset

Here is a hypothetical scenario involving the use of the Scope to diagnose a problem. Suppose you have a recent version of some computer sequencing software. This sequencer is driving a fairly large number of synths and samplers through Synapse. Lets say that one of the synthesizers is some vintage dinasaur with a very primative MIDI implementation.

Every time you click on the Rewind button on your sequencer, the old synthesizer "crashes" and has to be turned off and re-initialized. You want to use a Synapse Processor to filter out the MIDI message that is causing the crash, but you do not know which filter to select. The Scope could save you some trial and error time.

Assign the Remote In of Synapse to the sequencer. Click the Rewind button and observe the Status Byte. Suppose that the display says F 2, MIDI Song Position Pointer. This falls into the class of System Common messages. Assign a Processor to the sequencer and set it to filter out System Common. Then route the Processor to the Synapse output that is sending data to the older synthesizer.

It is likely that you will at first only see F 8's (MIDI Clock) coming from a sequencer. You should in that case turn the disable the sequencer's MIDI Clock output to observe other Status Bytes.

F 0 through F 7 are System Common messages:
F 0 = System Exclusive
F 1 = MIDI Time Code Quarter Frame
F 2 = Song Position Pointer
F 3 = Song Select
F 4 = Undefined
F 5 = Undefined
F 6 = Tune Request
F 7 = EOX: End of System Exclusive

F 8 through F F are System Real Time messages:
F 8 = TIMING clock
F 9 = Undefined
F A= Start
F B= Continue
F C= Stop
F D= Undefined
F E = Active Sensing **Note that this is not displayed by Scope.**
F F= System Reset

Here is a hypothetical scenario involving the use of the Scope to diagnose a problem. Suppose you have a recent version of some computer sequencing software. This sequencer is driving a fairly large number of synths and samplers through Synapse. Lets say that one of the synthesizers is some vintage dinasaur with a very primative MIDI implementation.

Every time you click on the Rewind button on your sequencer, the old synthesizer "crashes" and has to be turned off and re-initialized. You want to use a Synapse Processor to filter out the MIDI message that is causing the crash, but you do not know which filter to select. The Scope could save you some trial and error time.

Assign the Remote In of Synapse to the sequencer. Click the Rewind button and observe the Status Byte. Suppose that the display says F 2, MIDI Song Position Pointer. This falls into the class of System Common messages. Assign a Processor to the sequencer and set it to filter out System Common. Then route the Processor to the Synapse output that is sending data to the older synthesizer.

It is likely that you will at first only see F 8's (MIDI Clock) coming from a sequencer. You should in that case turn the disable the sequencer's MIDI Clock output to observe other Status Bytes.

TECHNICAL INFORMATION

System Exclusive Implementation
The JLCooper Synapse can transfer the contents of its internal memory via MIDI System Exclusive.

System Exclusive is received on the same MIDI input used for Remote and Scope operations. This is found on the Program Change page.

To Select the Program Change page, do either of the following:

(1) Hold down MODE and press button #18 or

(2) Tap WRITE once, and then hold in MODE and button #1.

Buttons 12 and 13 select the input for Synapse System Exclusive. The input number is displayed above the legend REMOTE IN.

Buttons 14 and 15 select which output will send Synapse System Exclusive. The output number is displayed above the legend REMOTE OUT.

To initiate a System Exclusive Bulk Data Dump, hold button #20, and while holding in #20, press button #1.

The data is in the following format:
F0h 15h 12h cc pn <data> F7h
cc=**00** A Full Dump Follows (8832 bytes)
cc=**01** Full Dump Request
cc=**02** One Program Dump where **pn** equals the program number (138 bytes)
cc=**03** One Program Dump request

Synapse Initialization Procedure

To completely and irrevocably clear out and reset Synapse memory, hold buttons #1, #2, and #3 simultaneously while turning Synapse power on.

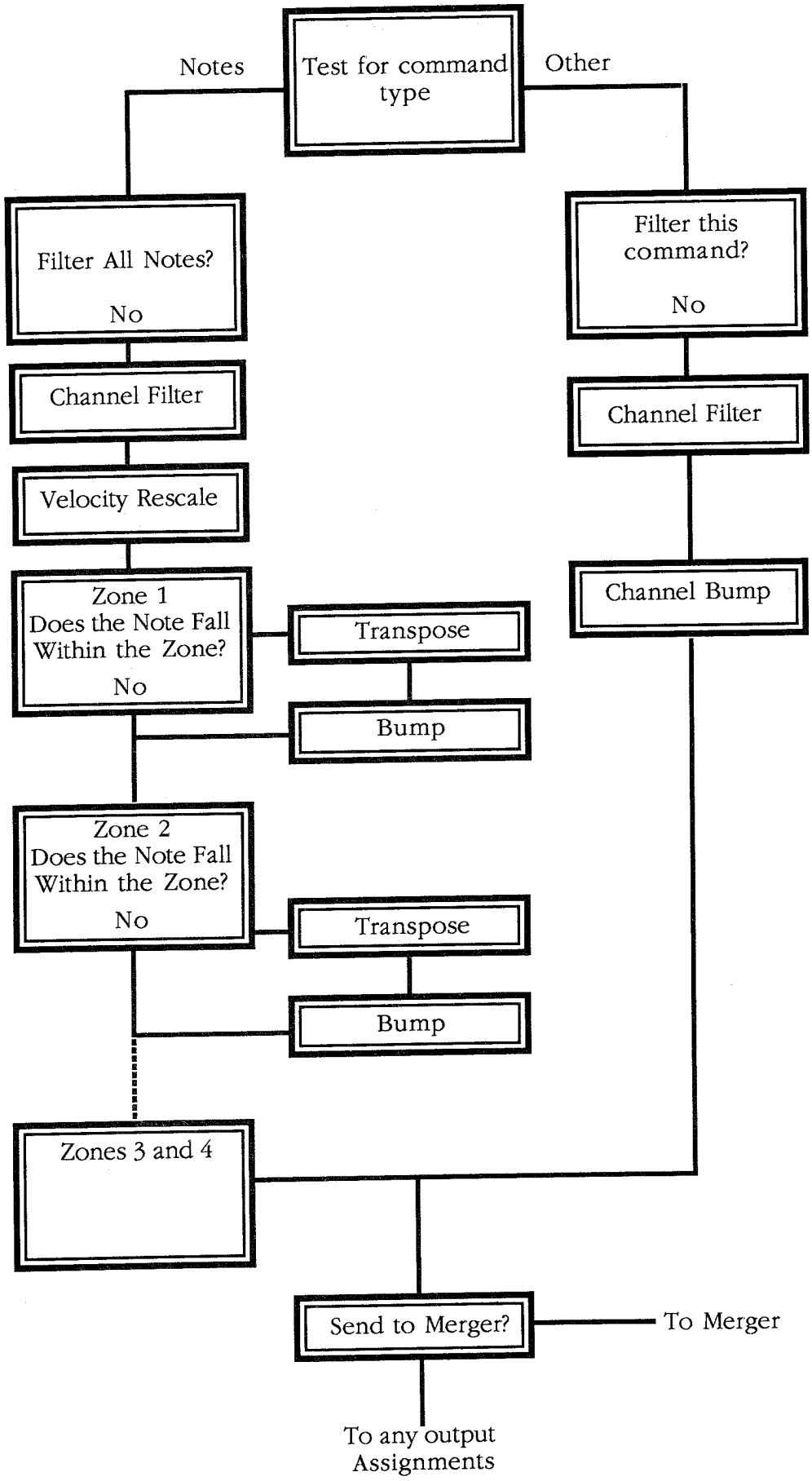
Program Numbers 44 through 63 will reload with their factory default settings.

Servicing
If you experience any operational difficulties, let us reassure you that every unit is 100% factory tested. It worked when it left the factory, otherwise it wouldn't have been shipped.

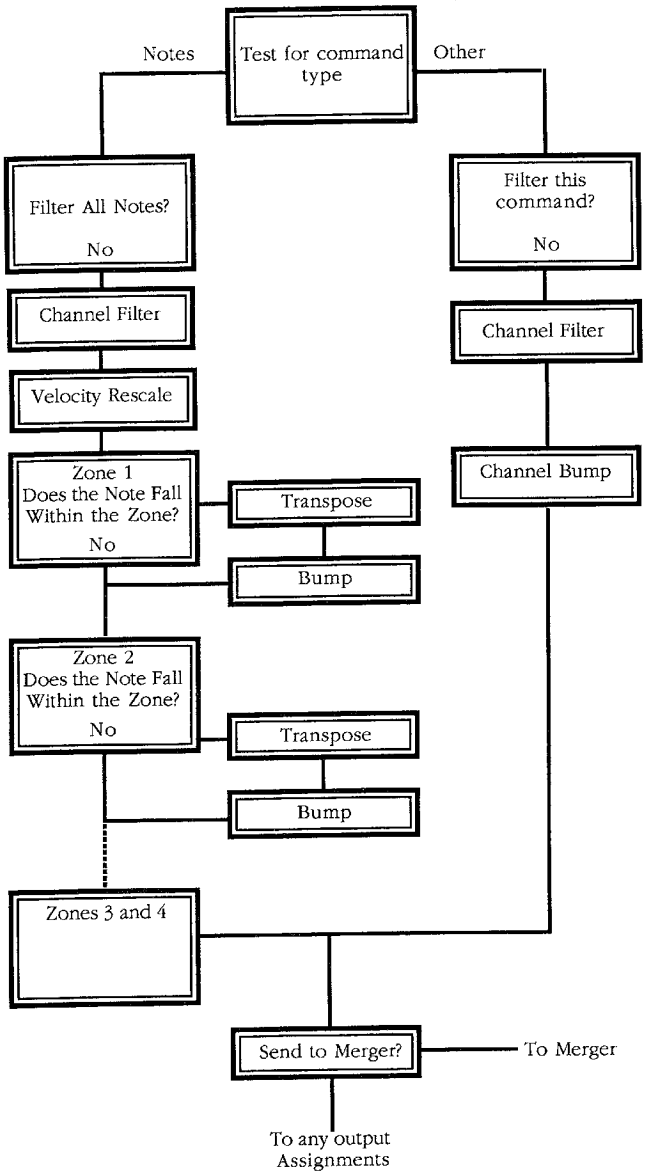
The most common cause of problems is a "noisy" AC outlet. We recommend the use of an AC line filter with all computer related equipment. These can be purchased at any computer or hardware store, and most music stores also now carry them. Be sure that it has both surge suppression and line filtering.

There are no "user-serviceable" parts inside Synapse. For warranty service in the U.S. in the event of a malfunction, call the factory to obtain a Return Authorization before sending the unit back. Please, read the manual and debug your system before calling the factory.

MIDI Data Processor Flow Diagram



MIDI Data Processor Flow Diagram



Synapse Remote Editor / Librarian Software Info

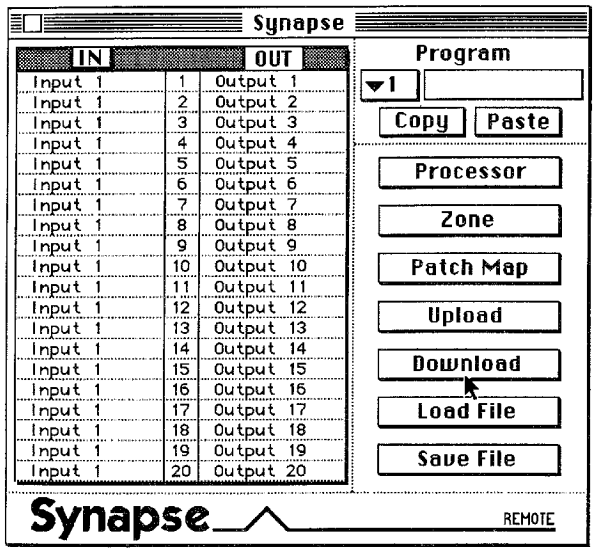
The optional Synapse Remote Editor / Librarian software permits controlling Synapse from either a Macintosh, Atari, or PC under Windows™. The software allows editing of all routing, data processing, merging, and patch map parameters. Inputs and Outputs can be given real instrument names, and each of the 64 programs can be given a name. Individual programs or all 64 programs can be uploaded from or downloaded to Synapse. Files consisting of all 64 programs can be saved to and retrieved from disk, providing unlimited Synapse memory storage.

The Macintosh and Atari software comes with both an application and a desk accessory.

Synapse Remote Main Window
Notice the table of inputs and outputs. The left **IN** column displays input names, the middle column displays output numbers, and the right **OUT** column displays output names. Thus the left column corresponds to the Synapse LED display in Assignment Mode. The middle column corresponds to Synapse buttons #1 through #20.

When you first open the window, you will notice that input 1 is assigned to all twenty outputs. This represents the same thing as Synapse displaying a one above every output assignment button, that is, "1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1"

Assigning Inputs and Outputs
To assign an input to an output, first select the output by **clicking on a number in the middle column.**



Clicking on the word **IN** pulls down the Input Assignment menu. Drag down to select the desired input and release.

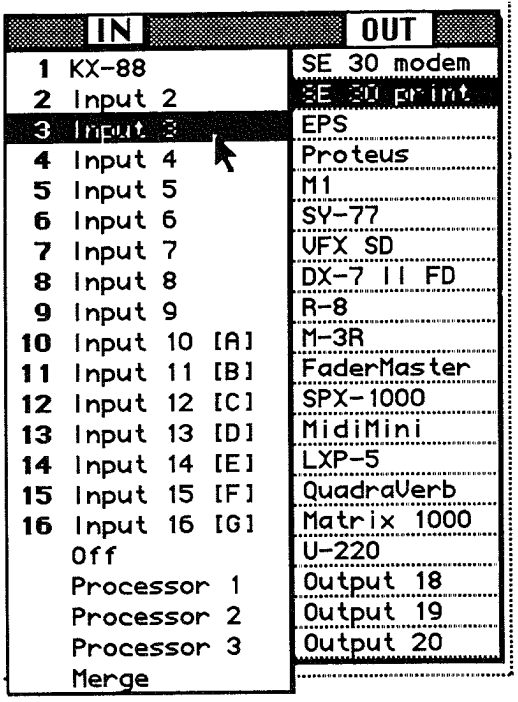
Notice that Inputs and Outputs can be given real instrument names.

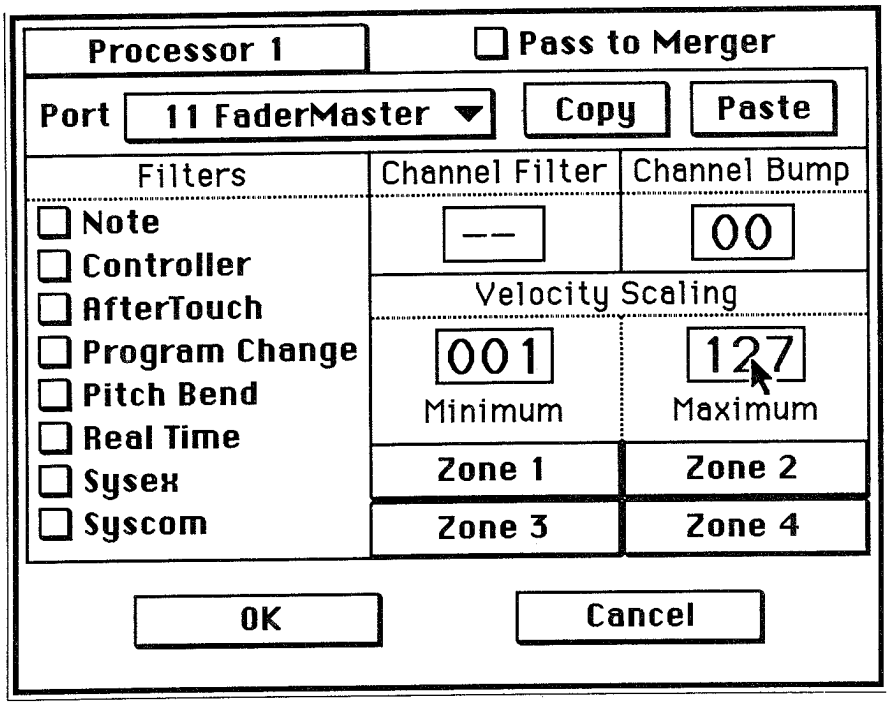
The software features many pull down and pop-up menus for speedy and simple operation.

Some of the other Synapse Remote windows are shown below.

For more information about the Synapse Remote software, refer the Synapse Remote Owners manual.

You may purchase the Synapse Remote software from your dealer, or use the enclosed order form.





The Processor Window

Clicking on the word **IN** pulls down the Input Assignment menu. Drag down to select the desired input and release.

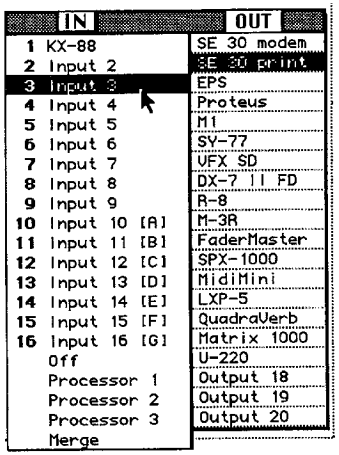
Notice that Inputs and Outputs can be given real instrument names.

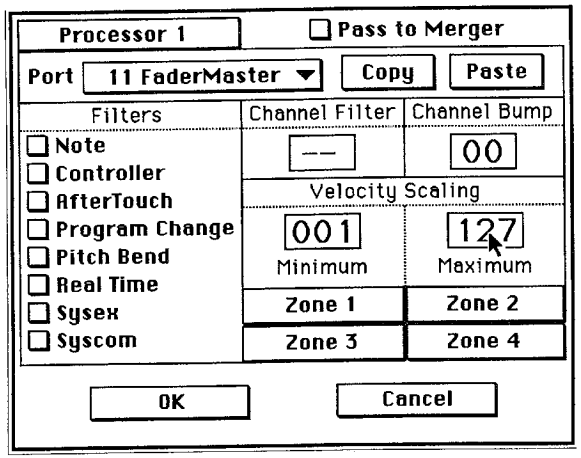
The software features many pull down and pop-up menus for speedy and simple operation.

Some of the other Synapse Remote windows are shown below.

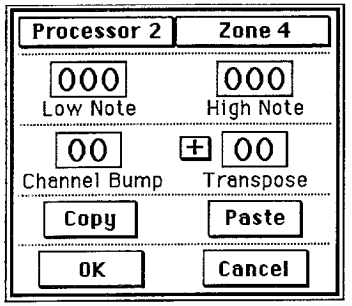
For more information about the Synapse Remote software, refer the Synapse Remote Owners manual.

You may purchase the Synapse Remote software from your dealer, or use the enclosed order form.

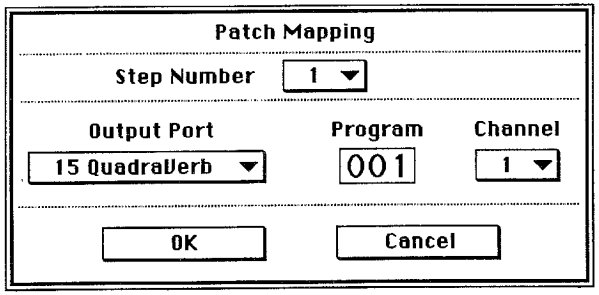




The Processor Window



The Zone Window



The Patch Map window features a pop-up menu of instrument names