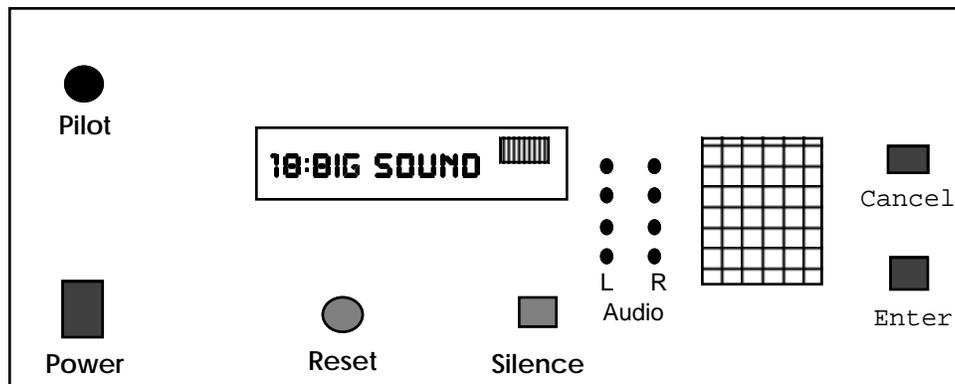


OBERHEIM 4-VOICE MIDI INTERFACE

Users Manual

- Updated 11/90
- Updated 9/91

Joe Paradiso
MIT Media Lab E15-351
Cambridge, MA 02139
617-253-8988



OBERHEIM 4-VOICE MIDI INTERFACE

Users Manual

Table Of Contents

1) Configuration	1
2) Interface Operation and Key/Button Functions.....	4
3) Fundamentals of Patch Editing	8
4) Editing the Patch Number/Title.....	10
5) The MIDI Mode Menu	11
6) The 4-Voice Set-Up Menu.....	15
7) The Set MIDI Out Menu.....	17
8) The Auto Portamento Menu	23
9) The Auto Bend Menu	25
10) The Auto Trill Menu.....	29
11) The Tone-Volume-Balance Menu	33
12) MIDI Implementation.....	34
13) System Exclusive Commands.....	42
14) Patch Editor/Librarian Program.....	44

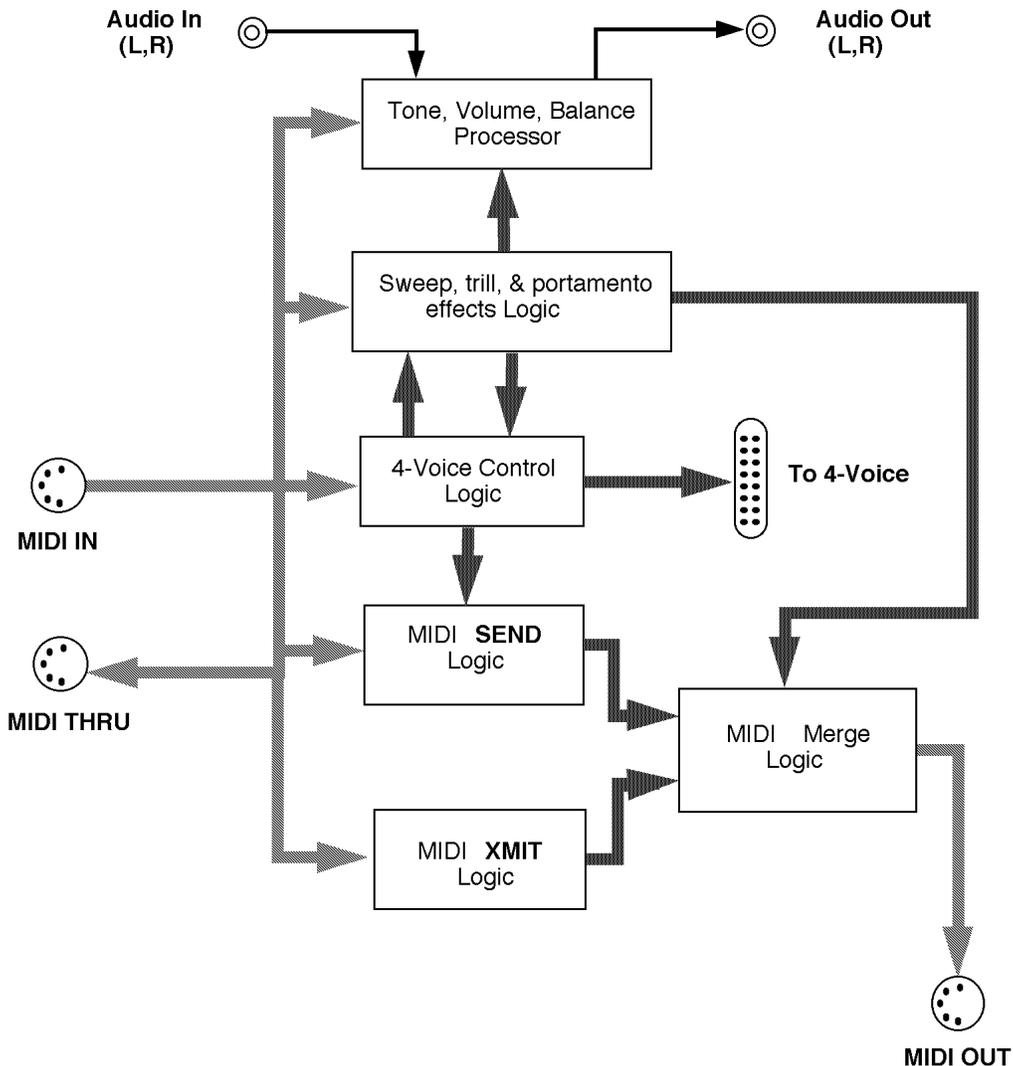
OBERHEIM 4-VOICE MIDI INTERFACE

Users Manual

J. PARADISO : joep@Media.Mit.edu

1) Configuration

This device controls the Oberheim 4-Voice synthesizer & a simple audio processor, and optionally drives a slave synthesizer through its MIDI output. A highly simplified block diagram of the Interface logic is given in the figure below (in this document, the rack-mount unit is termed the "Interface", and the circuitry built into the 4-Voice is called the "Supervisory Logic"):



One sees that this device accomplishes three purposes; it controls the Oberheim 4-Voice synthesizer (through the "4-Voice Control Logic"), acts as a powerful MIDI effects generator & interpreter by merging its own commands (generated via the "MIDI SEND Logic" and "Sweep Effects Logic") with the incoming MIDI stream (through the "MIDI XMIT Logic"), and enables MIDI control over a stereo audio signal ("Tone/Volume/Balance Processor"). The 4-Voice Control logic is configured in the MIDI MODE, 4-VOICE SET-UP, and AUTO PORTAMENTO menus. The Send and Xmit logic are configured in the SET MIDI OUT menu. The Sweep Effects are configured in the AUTO BEND and AUTO TRILL menus. The audio processor is adjusted in the TONE-VOL-BAL menu. The detailed function of these various menus, and the means by which they are edited, are explained in the succeeding chapters of this manual.

Rear panel Interface jacks include standard MIDI in/out/thru, a ¼-inch "Bend out" phone jack, a ¼-inch "VCF Ctl." phone jack, two ¼-inch "Audio in" & two ¼-inch "Audio out" jacks, and a 25-pin mini DIN connector (used typically in RS-232 devices). This DIN connector links the unit with the 4-Voice. It is not a typical RS-232 standard, however, so standard RS-232 cables may not work (since they do not connect to all of the pins in the connector). A cable must be used that connects all 25 pins (and is preferably shielded). This cable can probably be made as long as needed; since all signals are differentially driven, cable length should not be too critical. The pitch-bend and filter-control analog voltages are also routed through a balanced pair in the DIN connector. In case this causes problems (ie. injection of "digital" noise into the 4-Voice oscillators and/or filters), the bend voltage (and/or filter voltage) can be run through a standard "guitar cord" cable; merely plug into the corresponding ¼" jack on the rear of the Interface unit, and run this into the appropriately labeled ¼" jack added to the rear of the 4-Voice. Inserting a ¼" jack into the 4-Voice in this fashion automatically cuts off the pitch bend feed (or filter feed) included with the DIN cable. This feature can be used to externally inject a bend voltage (or filter voltage) into the 4-Voice (as with the alternate cable mentioned above), or to mute the bend (filter) entirely (use an unconnected ¼" jack to disengage the bend (filter) feed). The ¼" jacks added to the 4-Voice (bend, filter inputs) are actually a 3-connector balanced jacks, while the ¼" jacks on the rear of the Interface (bend, filter outputs) are 2-connector unbalanced jacks. For the most part, this doesn't matter; a standard unbalanced guitar cord should be adequate to connect between 4-Voice and Interface. In all tests, however, the connections through the DIN cable seemed to work fine, so these additional cables will probably not be needed.

When the DIN cable is not connected between the 4-Voice and Interface, the supervisory logic added to the 4-Voice is held in a reset mode. This enables the 4-Voice to work as it always has; the additional logic supervising its operation will have no effect. Upon power-up, or whenever the button added to the 4-Voice front panel is depressed, the 4-Voice supervisory logic will also be reset. If the 4-Voice has received a valid command from the Interface unit (other than a reset), the red LED added to the 4-Voice (next to the reset button) will be illuminated. This indicates that the 4-Voice may be in a commanded mode; any switch changes on the programmer or keyboard control modules may have unpredictable effect. If one desires to work directly with the switches on these 4-Voice modules, the supervisory logic should be reset first (as detailed above).

In addition, a "MIDI Control" toggle switch has been added to the 4-Voice panel (to the right of the above-mentioned LED). If this switch is up, the green LED nearby is illuminated, and the 4-Voice will not respond to incoming MIDI note commands (it will, however, still be programmed and configured under MIDI/Interface control; only note on/off's are inhibited).

In order to run the 4-Voice under external Interface control, a subset of switches on the Keyboard module should be set to the positions listed below:

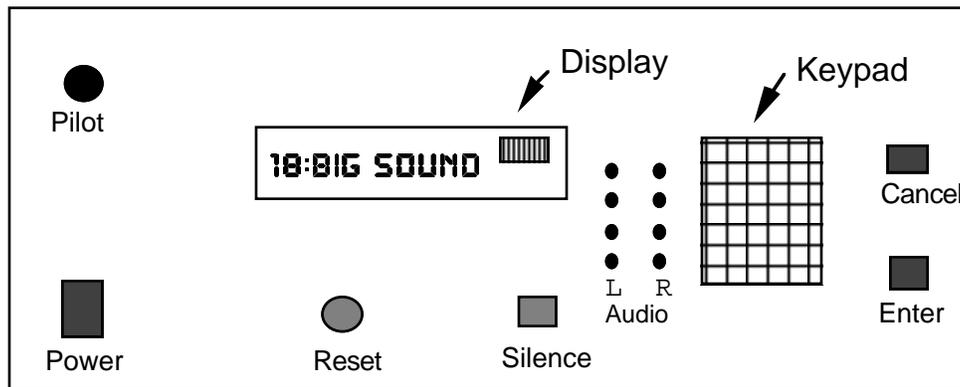
<u>Switch</u>	<u>Position for MIDI Control</u>
Split	OFF
Split Mode	2-2 (Center)
Portamento Mode	Both (Center)
Unison	OFF
Portamento	OFF
Reassign	ON
Assign	R/C (Center)

The Interface unit remotes the functions of these switches, as well as the Program selection and Program bank switches. Their settings can now be dynamically controlled through MIDI commands. In addition, the portamento time potentiometer should be fully counterclockwise (off), and the VCF control should be around center, in order to allow their reproducible adjustment through the Interface.

Note commands sent through MIDI will have access to a wider range than normally exists on the 4-Voice keyboard. 64 notes are available through MIDI, which gives roughly an extra octave and a third.

A quick note on contingencies is in order; if the supervisory logic resident in the 4-Voice ever malfunctions (ie. notes will be stuck on, the program will be unable to be changed, or the above switch settings will seem unchangeable), it may be readily disconnected from the 4-Voice electronics. Remove the Programmer and Keyboard modules. Be very careful, since cables have been added to these units. Remove the Programmer (gingerly!) first; the Keyboard module will no longer come out easily if it's attempted to be removed beforehand. The cables added to these units terminate in Molex connectors. Unplug these connections, and insert the dummy plugs provided with this unit in their place. Since these three molex connectors are entirely different, there is no possibility to cross-connect cables or dummy plugs. If the supervisory logic seems to be loading down the power supply (this isn't normally possible, since it's been designed to require under 100 mA or so), it may be disconnected from the master Oberheim power supply in the standard fashion.

2) Interface Operation and Key/Button Functions



Interface Front Panel

Upon power-up, the Interface loads in the last patch that was loaded or saved (it also programs the 4-Voice). The 12-Character display will show the patch number and title; ie.

18:BIG SOUND

The number at left is the patch number (running 1-99), and there are 9 characters at right for a user-programmable patch title. The 10-bar LED array to the right of the alphanumeric display is for diagnostic purposes; the leftmost 8 LEDs represent the 8 bit command sent to the 4-Voice, the 9th LED is the strobe sent to the 4-Voice, and the rightmost LED is the timer flag (this is perhaps the only one that is useful to an operator; ie. it is illuminated when the Interface is performing a timed wait).

The twin rows of LEDs to the right of the display show the input levels for the external audio signal. The audio level is adjusted to avoid frequent firing of the topmost red LEDs, and possible distortion/clipping. As there is no level control for these inputs on the Interface unit, they must be adjusted at the source.

The display can show two diagnostic messages if a hardware or software failure ever occurs. One is the message "**ACIA ERROR**", which indicates a major problem with the incoming MIDI stream. If this shows up, one should re-boot, check connected MIDI devices, cabling, etc. before trying again. The other message is "**MIDI OVFLOW!**", which indicates that the MIDI input stream was deluged by an incessant flood of rapid-fire MIDI data. This should never happen, however; the available buffer for MIDI input data is relatively massive, and the processor is fast enough to avoid accumulating significant backlog.

Below the display are two buttons. One is labeled "RESET"; this basically re-boots the Interface (just as the initial power on does), and should be used in extreme panic situations (ie. if the Interface locks up due to a power glitch, etc.). The other is labeled "SILENCE"; it is basically a lower level reset function that clears internal software flags and sends an "All Notes Off" to both the 4-Voice and the MIDI output (if the MIDI out is enabled in the current patch). All data is retained under this operation, and nothing is re-loaded or hard reset, thus this is the preferable way out of a bind. If anything seems to be going strange, hit this button first.

When the Interface unit is powered down, glitches will be sent through the DIN cable to the 4-Voice, potentially putting it into a random mode and turning several notes on. This may be remedied immediately by pushing the black reset button added to the 4-Voice, as outlined in Section 1. The two buttons to the right of the Interface keypad primarily control accepting and rejecting data (ie. "Enter & Cancel", as labeled).

The keypad is the primary mode of entering data. This is basically a calculator keypad. The numeric keys serve their original function. Numeric data may be entered in various ways. When the Interface is awaiting input, the cursor generally flashes over the characters that are to be changed. Numeric data may be entered by punching in a number, as on a calculator. If the number is too large, or too many keys have been hit, the quantity being changed will saturate at its maximum allowed value. If the red "C" key is depressed while editing, the number will be reset to the minimum allowed value. The "+" and "-" keys may also be hit at any time to increment and decrement the displayed value. If either of these keys are held down for a second or two, the quantity is quickly auto-incremented (or auto-decremented). This will continue until the + (or -) key is released, or when the quantity reaches its maximum (or minimum) allowed value.

Some items are not numeric in nature, but accept a choice of character options. In these cases, the numeric keypad keys will have no effect. The available items may be scrolled via the "+" and "-" keys (if held down, these keys again automatically step the values forward and reverse). The list of options is stepped through in a circular fashion; the list is not stopped at a maximum or minimum entry, but keeps repeating. The "C" key will reset the list to a simplest option (usually "OFF"). The "K" key has the same effect as the "+" key when editing a character list; it will step values forward. This can be convenient, as outlined in the "K Key" paragraph below.

Data is accepted by pushing "Enter". This causes the cursor to stop flashing, and allows one to scroll to other menu items. Note that the MIDI functions are continuously operational during the editing process. MIDI commands can be sent while a quantity is being edited in order to ascertain the change effected. The operation performed will reflect whatever is currently displayed. If one is editing the 4-Voice setup parameters, the 4-Voice is automatically re-programmed as a parameter is modified. One should not, however, hold down the "+" or "-" keys while sending MIDI data (ie. Note-On's). The auto-increment/decrement functions use timers employed by other software (and inhibit the MIDI data processing). This can't cause any real problem, but one might notice delays and/or abbreviated auto bend/trill functions (as will be described later).

Hitting "Enter" when not in edit mode scrolls downward through the current menu list (this can also be accomplished by hitting "+"). Hitting "-" is the only means of scrolling backward in this case.

The "Cancel" button functions mainly as an Undo/Exit command. While editing a parameter, hitting "Cancel" will cause the original value of the parameter to be restored, and ends the editing operation. If one hits Cancel when not in edit mode, you will be bumped up a level in the menu structure, or, if already at the top level, the patch number and title will be displayed (which is the lead item in the top level list).

The non-numeric keypad keys perform specific functions. The discussion below summarizes their abilities.

Dot key (.) This key always means "Save Patch". Whenever it is depressed, the display will flash "* OVRWRITE *" and alternate with the the patch title that is currently stored in memory. If it is desired to overwrite this patch, merely push enter. If one wants to preserve the original patch, push cancel. Another option exists at this point, however, which allows one to quickly peruse all 99 patch numbers and titles stored in memory, and pick the appropriate candidate to overwrite. If one hits either "+" or "-", the display will change. The patch number plus 1 (or minus 1 if "-" was hit) is displayed with its stored title. The colon separating patch number from patch title flashes under this option. The displayed patch is not actually overwritten at this point; only its number and title are shown. One may repeatedly hit + or - to scroll through all patches; the gap between 1 and 99 is automatically bridged whenever surpassed in either direction. If one doesn't want any of these patches, "Cancel" will abort the operation. If, however, one hits "Enter", the sequence flashing "* OVRWRITE *" will return, but the patch number displayed will be the last patch inspected. After a successful save, a confirmation message is displayed on the screen. Depressing any key will clear this message and continue operations exactly where you left off before pushing ".". If the write-protect flag was set (see description under "÷" below), the display will complain accordingly, and the save operation is intrinsically aborted.

Equals key (=) This key always means "Load Patch". Whenever it is depressed, the display will show "LOAD PGM#:", and the cursor will flash at right as it waits for an entry. The default is the current patch number, as listed. Numeric data may be entered by keying in a number, or hitting +, -, or C, as described above. "Enter" will accept the value, and cancel will abort the operation. If Enter was depressed, the display will alternate the message "**** LOAD ****" with the referenced patch number and title. If one indeed wants to load this patch, hit Enter again. Cancel will abort the operation and leave the resident patch intact. Another option exists at this point, however, which allows one to quickly peruse all 99 patch numbers and titles stored in memory. If one hits either "+" or "-", the display will change. The patch number plus 1 (or minus 1 if "-" was hit) is displayed with its stored title. The colon separating patch number from patch title flashes under this option. The displayed patch is not actually loaded; only its number and title are shown. One may repeatedly hit + or - to scroll through all patches; the gap between 1 and 99 is automatically bridged whenever surpassed in either direction. If one doesn't want any of these patches, "Cancel" will abort the operation. If, however, one hits "Enter", the sequence flashing "**** LOAD ****" will return, but the patch number displayed will be the last patch inspected. One can again hit Enter, Cancel, +, or -; as discussed above. If Enter is chosen, the patch is loaded, and a confirmation message is displayed. Hitting any key will clear this message, and return to normal operation exactly where one left off before pushing "=".

Plus Key (+) When editing a numeric value, pressing this key will increment the value by one (holding it down will produce an auto-increment function). Incrementing stops at the maximum allowed value. When editing a character value, the "+" key will scroll "upward" through the list of available items (again, holding it down will auto-increment). When not editing, pressing the "+" key will scroll the display to the next parameter in the current menu (this can also be accomplished by hitting the "Enter" key in this situation). This scrolling function will not auto-increment.

Minus Key (-) When editing a numeric value, pressing this key will decrement the value by one (holding it down will produce an auto-decrement function). Decrementing stops at the minimum allowed value. When editing a character value, the "-" key will scroll "downward" through the list of available items (again, holding it down will auto-decrement). When not editing, pressing the "-" key will scroll the display to the previous parameter in the current menu. This scrolling function will not auto-decrement.

Multiply Key (X) When at the top-level menu, hitting the X key will re-program the Oberheim 4-Voice with the currently loaded parameters. Upon depressing the X Key, the message "X TO PGM 4-V" appears on the display. Hitting "X" again will cause the 4-Voice to be programmed. Hitting any other key cancels the operation. Upon completion of the programming operation, a confirmation message is displayed, which is then cleared after hitting any key. This operation might come in handy if one needs to quickly re-load the 4-Voice (ie. if it has suffered a power loss or has otherwise been disturbed). Notes are not turned off; the "Silence" button can be used for this.

If one is in a lower level menu, this function is not available. The X key will now cause the display to show the top level item which is currently being edited. This may be useful in refreshing one's memory as to which lower level menu is under inspection (ie. the lower level items under "Auto Bend" and "Auto Trill" can appear quite similar; some confusion is possible).

Divide Key (÷) When at the top-level menu, hitting the "÷" key will allow the write protect option to be enabled and disabled. Upon depressing "÷", the write protect status is displayed, with cursor flashing. This is either "Y" or "N". If it is set to "Y", the write protect is active, and patch data can not be saved, either through the "." key scenario sketched above, or via system exclusive commands. If it is "N", data can be freely written into patch storage via either method. One can use the "+", "-", or "C" keys to edit this value. Hitting enter sets the write protect flag to its displayed value.

If one is in a lower level menu, this function is not available. The "÷" key will now cause the display to flash the current patch number and title. This may again be useful to combat cases of forgetfulness.

Clear Key (C) When at the top-level menu, hitting the "C" key will allow one to reset the 4-Voice supervisory logic. After depressing "C", the message "C TO CLR 4-V" will appear on the display. If "C" is hit again, the 4-Voice will be reset (this accomplishes the same function as hitting the black button added to the 4-Voice panel), and a confirmation message is displayed (which is cleared after hitting any key). If any other key is pressed, the operation is aborted.

This function is unavailable when in a lower level menu. If one hits "C" while editing an item, the quantity is set to its lowest allowable value (or potential "OFF" state when editing a character list).

K Key (K) This Key generally means "Go to the lower level menu" or "Edit this quantity". Assuming one is at the top-level menu, hitting "K" at the patch title display will cause it to be edited (as described below), while hitting "K" at a sub-menu title will cause one to enter the lower level scroll for that sub-menu. If one is in a lower level menu, hitting "K" will allow the displayed value to be changed (ie. the cursor will flash over the value, and it can be changed by any of the methods described above). If the quantity is non-numeric (ie. it is a list of character options), one can hold "K" down (or hit "K" repeatedly) to scroll through the list of values (this can also be done with the "+" key, although keeping one's finger on "K" may be more convenient). The opposite of "K" is the "Cancel" button; this will abort the edit operation (and restore the original value unchanged), or, if scrolling through a lower level menu, bump up to the top-level menu. The "K" key takes on additional significance when editing a patch number& title, as detailed in Chapter 4.

3) Fundamentals of Patch Editing

Upon power-up, the Interface loads the last patch either saved or accessed. The display will show the patch number and patch title, and place the patch editing software at the top level menu. This menu contains 8 entries that correspond to different aspects of Interface function. Each of these entries is a title for a submenu that contains several specific quantities that can be changed to specify Interface performance. One can scroll through the top-level menu by repeatedly pressing the "+" key or "Enter" key (forward scroll), or "-" key (backward scroll). "Cancel" will always return the display to the patch number:title. The 8 items in the top level menu are:

- 1) **Patch# : Patch Title**
- 2) **MIDI MODE**
- 3) **4-VCE SET-UP**
- 4) **SET MIDI OUT**
- 5) **AUTO PMNTO**
- 6) **AUTO PBEND**
- 7) **AUTO TRILL**
- 8) **TONE-VOL-BAL**

This list (as with all menu lists) is scrolled in a circular fashion; ie. the display will step from item 8 to item 1 on a forward scroll (or item 1 to item 8 on a backward scroll). In order to inspect and/or edit the submenu and parameters associated with any of these categories, one scrolls the display to the heading desired and presses the "K" key. The editor will then enter the appropriate submenu (or start editing the patch title if the display was on item 1), which can also be scrolled through and inspected via the "+", "Enter", and "-" keys. If one desires to modify any value displayed, again hit "K"; the cursor will flash over the current value, and it can be changed via the methods described in the last section. Pressing cancel will return the display to the top level menu, and another above-listed category can be selected (pressing cancel while actually editing, however, will abort the edit and restore the original parameter value).

Items 2 through 4 on the top-level menu specify the operation of the Interface and 4-Voice. Item 2 allows one to set the MIDI input characteristics and specify the way in which the 4-Voice will respond to incoming commands. Item 3 is essentially a remote control on the 4-Voice keyboard and programmer modules; the 4-Voice can be entirely re-moded from this menu. Item 4 dictates the way in which the MIDI output will operate, and how it will correlate with commands sent to the 4-Voice.

The next 3 items are effects which can be sent to the 4-Voice and/or MIDI output. If any of these effects are enabled, a "*" will be illuminated in the rightmost character of the corresponding display (if the "*" is not present in the top-level menu listing, the effect is deactivated). Item 4 is a velocity-sensitive portamento effect which can only be used with the 4-Voice. Items 5 and 6 are sweep effects that can be triggered via note-on, delayed note-on, note-off, and an external MIDI switch controller. Their performance is velocity-sensitive, and their effect can be applied to the 4-Voice, the MIDI output pitch bend (and/or any MIDI output controller), and the onboard audio processor. The "Auto Bend" is a

simple sweep effect, while the "Auto Trill" is an adjustable pulse-wave, triangle, or sweep modulation effect. Auto Bend and Auto Trill can both be active simultaneously, and sent to the same or different MIDI destinations (ie. they can both be assigned to pitch bend, one to pitch bend and one to any controller, both to different controllers, or both to the same controller). They can also be assigned to the 4-Voice pitch, 4-Voice filter, and the audio processor's volume, bass, treble, and balance functions.

Item 7 enables one to configure the basic operation of the audio processor. Its base parameters can be set, and their response to MIDI note velocity specified.

The next chapters will describe the functions of the submenus and procedures associated with each of the 8 top level items listed above. It must be remembered that although that any edits performed to the resident patch immediately affect the operation of the Interface and 4-Voice, none of this data is saved, and will be lost when the Interface is powered down or another patch is loaded. The "." key must be used to save any patch changes, as was discussed at length in the previous section.

4) Editing the Patch Number:Title

To change the patch number (in order to save into a different patch location) or patch title, scroll the display to show the patch number:title and press "K". The patch number will flash, and can be edited by conventional numeric means (it ranges 1 through 99). If one then presses "Enter" (or "K" is hit again), the new patch number will be accepted, and all 9 characters of patch title will begin flashing. At this stage, one has several options available, as detailed below.

Editing ASCII patch titles through a keypad can be a tiring ritual. One difficulty that I've encountered in other devices with a similar user interface is the lack of any ability to copy the title of one patch easily to another for simple modification. This is very handy for similar effects; ie. one may want to label a new patch "**COW BELL 2**", especially if it is similar to another patch called "**COW BELL 1**". It would be very convenient to copy the "**COW BELL 1**" title into the current patch (which we'll assume is called something else), and just change the "1" at the end to a "2". This can be accomplished very easily with the Interface title editor. When all title characters are flashing (as described above), one may hit the "+" and "-" keys to step through all patch titles (displacements are relative to the current patch; ie. if we're editing patch 34, pressing "+" will display the title of patch 35). If one obtains the desired title, pressing "Enter" will latch it into the current patch, or pressing "K" again will edit the individual characters, as described below. Pressing "Cancel" will restore the original patch title and abort the title edit (the patch number will start flashing, and can again be edited).

Pressing the red "C" key when the entire title is flashing will clear all patch characters; this may be useful when creating a totally different title. Again, the "Cancel" button is an "Undo" function, which will restore the original title in the event of a mistake.

If "K" is hit again, only the first patch character will flash. This character can then be changed via the "+" and "-" keys. Holding down either of these keys will cause the display to auto-increment (or auto-decrement) through all possible letters, numbers, and symbols. One can also hit the red "C" key to promptly clear the character. Hitting "K" again will advance to the next character in the title, which can be edited in a similar fashion. By repeatedly hitting "K", one can step through all nine characters in the patch title; if "K" is struck on the last character, the first character will begin flashing, and can again be edited. Hitting "Enter" will latch the modified title into the current patch, and stop the editing process. Hitting "Cancel" will restore the original character. If "Cancel" is hit twice in succession, the original patch title will be restored (regardless of any changes made), and all characters will flash, allowing any of the options described above to be repeated.

5) The MIDI Mode Menu

This menu allows the MIDI input characteristics of the Interface to be specified. This menu also contains several items which dictate the way in which the 4-Voice will respond to note-on and note-off commands; these options can change the sound and operation of the 4-Voice significantly. Each item in the menu is discussed independently below:

1) CHANNEL# This specifies the base MIDI input channel number, and ranges 1 → 16. The interpretation of this parameter varies, depending on what has been selected for the OMNI value, as discussed below. Note that the channel number is specified here as a patch-loadable parameter. This is not standard in MIDI gear, for a very good reason; ie. a program-change command can load a patch that listens to an entirely different channel, potentially cutting off communication. It was more convenient, however, to make this quantity patch loadable in the Interface software, thus it was done this way. Please be sure to check that the channel numbers in the patches used correspond to their intended values, otherwise trouble could result when patches are changed.

2) OMNI This item allows one to set the restrictions governing the Interface response to incoming MIDI data. The possible values are [**OFF**, **ON**, **X-CTRLS**, **MULTI**]; the action taken under each of these entries is detailed below:

OFF Omni mode is turned off. The Interface will only respond to MIDI commands arriving on the channel number specified above under Item #1.

ON Omni mode is turned on. The Interface will respond to all incoming MIDI commands, regardless of their channel number. The only exception to this, however, are the channel mode commands (ie. MIDI controllers with index greater than or equal to 123), including the "All Notes Off" command. In these cases, the channel number associated with the MIDI command must agree with the channel set above in Item #1.

X-CTRLS This is a more restrictive OMNI mode. The Interface will respond to note-on & note-off data arriving on any channel (as with OMNI ON), but will only accept controller data (except for sustain), pitch-bend data, and program-change data arriving on the basic channel specified in Item #1.

MULTI This is a quick-and-dirty hack at running the 4-Voice in a multi-timbral mode. As with the X-CTRLS mode specified above, all controller (except sustain), pitch bend, and program change commands must be sent over the basic channel # (specified in Item #1) in order to be accepted. Note-on & Note-off commands, however, are processed differently. Note commands sent on the basic channel will turn notes on-and-off only in the lower half of the 4-Voice keyboard (ie. the lower 24 4-Voice keys). Note commands sent on the basic channel plus one will turn notes on-and-off in the upper half of the 4-Voice keyboard (ie. 4-Voice keys 25 through 64). Commands sent on other MIDI channels will be ignored. The key boundary specified here corresponds with the split point set in the 4-Voice hardware, thus each keyboard half (ie. each set of "voices" under this option) can be made to sound entirely differently through the 4-Voice Split Mode,

and will reside on separate MIDI channels. This option may be useful in MIDI sequencing applications. A few drawbacks do exist, however; the pitch-bend commands (accepted only on the base channel) will affect both sets of voices. The range of each voice is also somewhat limited; The lower voice has 24 notes available, while the upper voice has 40 notes (the 4-Voice keyboard range is extended somewhat under MIDI). The offset is automatically added into the upper keyboard notes; the commanded note numbers for both keyboard halves will both start at the same value, as set through the transpose item (#6) discussed below. Notes sent above or below their limits are not sounded (but can be routed to the MIDI output through the Send Overflow option, as will be discussed with the SET MIDI OUT menu).

3) MONO This item allows a software-driven mono mode to be set on and off. When off, the 4-Voice will operate in a polyphonic mode (depending, as usual, on the settings of Split Mode and Unison). When on, however, the Mono option will only allow one key to be down at a time. Any previous note is turned off when a new note arrives. If several notes are held down in the MIDI input stream, the key priority can be set with Item #4, discussed below. This option may be useful if the 4-Voice is to follow a polyphonic input stream; it can be set to follow only the top, bottom, first, or last note, as defined in the key priority. Any unsounded notes can be routed to the MIDI output through the Send Overflow option in the SET MIDI OUT menu. The Interface unit can also be used to put any other MIDI synthesizer into a prioritized mono mode through this option; if the Mono mode is set on and the "Echo 4-Voice" output mode is active in the SET MIDI OUT menu, the synth. connected to the MIDI output will follow the monophonic 4-Voice response detailed above. Note that this Mono mode is independent of any settings in the 4-Voice hardware; Unison, Split Mode, etc. must be commanded separately. The Mono mode does not know about the split point; only one note will be allowed to be down over the entire 4-Voice range, regardless of the keyboard half in which it resides.

4) KEY PRTY This item allows a software key priority to be imposed on any notes that are simultaneously held down. Its allowed values are [**OFF**, **HI**, **LOW**, **FST**, **LST**]. This option knows about the 4-Voice hardware settings (ie. Split Mode &, Unison) made in the 4-VCE SET UP menu (but it doesn't know anything about actual switch settings on the 4-Voice itself, and assumes the 4-Voice has been entirely moded through this menu), thus directly accounts for the number of voices available in any keyboard half when Split is activated. It also knows about the MONO mode described above under Item #3, and will assume only one voice available across the entire keyboard if MONO is active. Overflow notes may be sent to the MIDI output (via the Send Overflow option in the SET MIDI OUT menu), or any synthesizer attached to the MIDI output can be forced into the assigned key priority through the "Echo 4-Voice" option in the SET MIDI OUT menu. Each of the values for the Key Priority are discussed briefly below:

OFF No key priority is applied. The notes are sent to the 4-Voice exactly as they are received. This is the "fastest" option. Any delay encountered in the prioritizing software (which should be fairly minimal) will be eliminated here, since all the prioritizing software will be bypassed.

HI Priority is given to the highest notes.

LOW Priority is given to the lowest notes.

FST Priority is given to the first notes that have arrived.

LST Priority is given to the most recent notes.

5) REALLOC This item allows the 4-Voice note reallocation routines to be turned ON or OFF. If ON, and assuming that all available voices are taken, new 4-Voice notes are automatically activated whenever an appropriate MIDI note has been turned off (and a voice has been freed). This option is only relevant if a key priority has been imposed via the previous item (#4); otherwise it has no effect, and the 4-Voice acts as it normally would. If on (and if subject to a full multi-voice polyphonic MIDI input stream), all 4 Voices (or less if Split, Unison, or Mono are selected) will be continually active, automatically reassigning whenever a voice frees up. This can sometimes sound fairly frenetic. If the 4-Voice is following a multi-voice polyphonic passage, it may sound better if this option is turned OFF. This will prevent voices from jumping around when several keys are held down; ie. the Interface will not instantly reallocate any voice that has been freed up. If Reallocation is active, the voices are reallocated according to the note priority in the HI and LOW case (ie. the voice is reallocated to the next highest or lowest unsounded note, as set in Key Priority). The closest unsounding note is used if the Key Priority is set to FST or LST. The MIDI output will track what happens here in the "Send Overflow" or "Echo 4-Voice" modes. The Reallocation knows about the keyboard split (if Split Mode is active), and will select notes appropriate to each keyboard half. It also is aware of the Unison and Mono settings.

6) TRNSPSE This item allows one to specify a transpose value (in note #'s; ie. semitones) that will be subtracted from the input MIDI note-on & note-off commands. This value ranges 0 → 127, and represents the base note# (corresponding to the lowest key on the 4-Voice or 4-Voice half if the Multi-timbral option is selected under Item #2). Any notes that are below this value (or above this value plus the appropriate keyboard range) will not be sent to the 4-Voice, but may be directed to the MIDI output if the "Send Overflow" option has been selected in the SET MIDI OUT" menu.

7) AUTOCRD This item may be turned ON or OFF. Its original intent was to cut off decaying notes sounding on the 4-Voice whenever a new note-on was received. It does do this, but its best application is as an independent effect; it plays a chord on the 4-Voice whenever a new key is hit. The chord is determined by Item #8 ("INTERVAL") below. It can be set by the user (i.e. played on the keyboard), recalled from a previous setting (each patch has a programmable chord assigned to it which is saved with the patch), or set to a default ("OCT" under the "INTERVAL" item below).

If "OCT" is selected, this effect sounds kind of like a polyphonic slapback "Unison" (with Reassign on, Assign set to "Cont", and Split Mode off) or gives more of a "fat" Moogish sound (with Reassign off, Assign not set to "Cont" or Split Mode on). Upon receipt of a new note-on, Autocrd will send rapid key pulses to the 4-Voice to briefly allocate all free voices. If Reassign is on (in the 4-VCE SET-UP menu), the incoming note is used to allocate all voices; rapid key-down and key-up commands are sequentially sent, as if one hit the note several times in succession. The delay between notes can be noticed, giving kind of a "slap back" effect. If Reassign is off (or Split is on, which turns the Reassign off by default in the 4-Voice hardware), the same note can not be sent sequentially, as above (otherwise the same voice would sound, and the others would remain intact). Instead, keys shifted up and down by octave intervals (relative to the new note) are hit on the 4-Voice, giving a "thick" sound. If an octave can not be used (because the latest note is too close to the edge of the available keyboard, or notes already held down are at octave intervals), the Autocrd algorithm will look for fifths (giving an even "thicker" sound). The delay needed for the Reassign-ON case is not used here; all notes will sound almost instantaneously. Autocrd is disabled if Unison is set on. It knows about the Split status, and will allocate however many voices are available per keyboard half (although only the "Reassign Off" option is intrinsically available under Split). If the Delay option is

active under the Auto Portamento menu, Autocrd may assume an extra voice to be available, thus it might sound slightly different.

The same or octave/fifth notes are played under Autocrd if the "INTERVAL" setting is set to "OCT". If it is set differently, the intervals programmed by the user (as discussed below) are used to determine the notes actually played. If the interval won't fit (because the edge of the keyboard is too close), the chord is appropriately inverted. The "Reassign On" mode of this effect will still insert a small delay between each note (thus the chord is slightly flammed), while the notes will play simultaneously under "Reassign Off".

For the full Autocrd effect, a 4-Voice program should be selected that has a quick attack & long decay, otherwise the transient voices will not sound, but only "blip" (if a sharp attack is selected). The chord will only be produced for the first note hit; if keys are held down, subsequent keys will re-allocate the Oberheim voices & be heard, but won't produce a chord. This effect is only relevant to the 4-Voice, and does not influence the MIDI output. The Autocrd effect may also be dynamically turned on and off and the INTERVAL setting changed by MIDI controller commands.

7) INTERVAL This item determines the intervals that will be played on the 4-Voice under the "Autocrd" effect discussed above. It has three possible settings; "**OCT**", "**USR**", "**SET**". "OCT" will produce the flammed unison or fifths/octaves (depending on "Reassign, Split Mode, and Assign" in the 4-Voice set-up menu), as discussed at length above. "USR" will play an additional 3 notes (whenever a new note is initially hit on the keyboard without any prior notes down) using a set of intervals that are stored with each patch. The interval is programmed via the "SET" option. When SET is selected, the first note hit is assumed to be the root of the chord, and the next three notes that are depressed are used to determine the intervals. The intervals are inverted appropriately so that all notes will be within 16 semitones of the root (the root is assumed to be the lowest note; lower notes are shifted up). After the 4 keys are hit, the INTERVAL setting will be automatically changed to "USR" (although this won't be shown on the display until it is updated). When the current patch is saved, the new chord will be stored with it, so that the "USR" setting will still play its intervals when the patch is re-loaded. If a patch is saved with the INTERVAL setting still at "SET", the first four notes played after the patch is loaded will determine the chord that is played under "AUTOCRD". This trick enables one to create a patch that will define its chord upon being loaded (i.e. via a MIDI "program change" command).

7) KEY SAFE This item may be set "ON" or "OFF". It enables a safety feature (when "ON"), where the "Soft" Pedal (MIDI controller #67) must be held down in order for key-triggered effects to occur. These effects include the Auto Trill and Auto Bend; if they are enabled to be triggered by hitting a preset key and the Key Safe option is set "ON", the Soft pedal must be down while the appropriate trigger key is hit for the effect to start. Key Safe is also used to guard against accidentally overwriting the note buffer that can be programmed into the MIDI output (see the last 4 items in the "SET MIDI OUT" menu). The note buffer may be recorded over when a pre-set key is hit or the "Sostenuto" pedal is pressed (as determined by the "REC KEY#" option in "SET MIDI OUT"). If Key Safe is set "ON", the Soft pedal must also be depressed while the preset key (or Sostenuto pedal) is pressed, otherwise notes will not be recorded and the note buffer will not be overwritten. If Key Safe is set "OFF", no check on the status of the Soft pedal is made before the bend/trill effects and/or note recording are triggered.

6) The 4-Voice Set-Up Menu

This menu allows one to configure the 4-Voice hardware. All of the contained items (except for items #1,12,13,14) refer directly to switch and pot settings on the 4-Voice's Keyboard and Programmer modules (or other hardware-related parameters). These switches must be set as listed in Section 1 for the parameters contained in this menu to be properly remoted. The 4-Voice is configured immediately when any item is changed, when a patch is loaded, or when the "X" keypad option is used (as described in Section 2). Many of these items may also be dynamically altered through MIDI controller commands. Each menu item is independently discussed below:

1) PLY NTES This option is the only item in this menu that does not refer directly to a switch setting or hardware parameter. It means "PLAY NOTES", and can be set "ON" or "OFF". When set on, the 4-Voice will respond to appropriate note commands. When set off, the 4-Voice is muted (although the hardware is still re-moded according to this menu, and MIDI controllers, program change, & pitch bend remain active). The 4-Voice can still be played manually when this item is set OFF, but no MIDI notes are sounded. This item may be useful in a patch that does not use the 4-Voice, but only employs the MIDI effects and utilities on the Interface's MIDI output. A switch has also been added to the 4-Voice that has the same effect as Play Notes (if either the switch or this item is set "off", the MIDI notes are not played).

2) PRGM # This specifies the program number active on the 4-Voice programmer. It runs 1 → 16. Programs 1 → 8 reside in the "lower" 4-Voice memory bank (without the red LED illuminated on the programmer), while programs 9 → 16 reside in the "upper" bank (with the red LED shining).

3) PBNDRNG This specifies the range of the 4-Voice response to MIDI pitch bend commands (and Auto Bend/Trill effects routed to the 4-Voice pitch bend). It runs between 0 and 255, and is essentially uncalibrated (maximum bend range is on the order of roughly ±1.5 octaves). The Interface always uses a full 12-bits of pitch bend, thus can achieve a very smooth response with MIDI controllers that send 2 bytes of pitch bend data.

4) PMNTOM This is a remote of the "Portamento Mode" switch on the 4-Voice Keyboard module. It may be set to [OFF, BTH, UPR, LWR], in correspondence with the settings on the actual switch (the OFF setting turns the portamento off, regardless of the portamento time set under Item #5 below). The other values define which half of the keyboard will possess portamento under Split mode.

5) PMNTOT This is a remote of the "Portamento Time" control on the 4-Voice Keyboard module. It ranges between 0 and 63. If the portamento has been turned "OFF" under the Portamento mode set above, there will be no effect (until it is turned "ON").

6) UNISON This is a remote of the "Unison" switch on the 4-Voice Keyboard module. It can be turned ON or OFF.

7) SPLIT M This remotes the Split functions available on the 4-Voice Keyboard module. It may be set to [**OFF, 2-2, 3-1, 1-3**]. When off, no split is active, and the keyboard is continuous. The other settings correspond to the allocation of voices available per half. The Reassign function does not work when the 4-Voice is in a split mode (this is the way the 4-Voice always functioned).

8) REASSIGN This remotes the Reassign switch on the 4-Voice Keyboard module, which determines whether recurrent notes are stolen from previously allocated voices, or new voices are sequentially allocated upon the receipt of each new key-down. It may be set ON or OFF. Under split mode, its setting is irrelevant, since reassign doesn't work with a keyboard split.

9) ASSIGN This remotes the Assign switch on the 4-Voice Keyboard module, which determines how the voices are allocated. It may be set to [**R/C, CONT, RESET**]. The meaning of these are described in the 4-Voice user's manual. "CONT" selects the cyclic reallocation of notes, R/C always starts with voice #1 (when a the first key is hit), then reallocates following notes in a cyclic fashion (as long as the first key is still held down), and "RESET" will allocate voices in a particular order (leftmost first).

10) OCTAVE This item transposes the 4-Voice oscillators up and down by octave intervals. A range of 2 octaves down and 1 octave up can be specified. The settings are [**0, -2, -1, +1**].

11) VCF BASE This item controls the base frequency of the 4-Voice VCF's. It is analogous to the "VCF" knob on the keyboard control module. This setting ranges from 0 through 255, and can produce a very wide effect (settings near zero bring the filter cutoff down so low that little audio can be heard leaking through).

12) VCF SWP This specifies the range of the 4-Voice response to MIDI VCF commands (and Auto Bend/Trill effects routed to the 4-Voice VCF). It runs between 0 and 255, and is essentially uncalibrated (full scale produces an extremely wide sweep, however). The Interface always uses a full 8-bits for VCF commands (7-bits with a MIDI controller); these bits of data are used to span a VCF frequency interval that is set through this item.

13) VCF CTLR This item specifies which MIDI controller will command the VCF cutoff frequency. It has two settings [**MOD, #3**]. "MOD" selects the modulation wheel (MIDI controller #1), thus the VCF cutoff can be swept by moving the mod wheel. "#3" selects MIDI controller #3 for the VCF cutoff, useful in cases where the modulation wheel is reserved for other purposes.

14) VCF VEL This item sets the key velocity sensitivity of the VCF cutoff. It ranges from 0 (no effect) through 127. Hitting a key harder will make the 4-Voice sound brighter when a sizable velocity sensitivity is selected. All voices are affected here at once, however; the velocity with which a particular key is hit will also change the filter cutoff for all previously sounding voices. The range of this effect is not changed by the VCF SWP setting on item #12 above.

7) The Set MIDI Out Menu

This menu allows one to specify the characteristics and operation of the MIDI output. The Interface effectively operates as a MIDI merger and re-channelizer, inserting its own commands and modifying input MIDI commands for transmission to the output. Most functions occur in parallel with 4-Voice operation (the MIDI out can't operate completely independently; there's always some correlation), but the 4-Voice can be easily muted (via Item #1 on the previous menu), and the Interface effects & functions can be used only on the MIDI output. Each menu item is discussed independently below:

1) OUT CHL# This is the channel number assigned to the MIDI output device. Internally generated commands are formed on this channel, and the interpretation of incoming MIDI commands with respect to the MIDI output will depend upon this parameter (according to the settings assigned to the items below). The Output channel can be assigned identically to the basic input channel in the MIDI MODE menu, but this might cause some confusion, thus it is recommended that different channel numbers are used.

2) XMIT M This command sets the "Transmit Mode". It sets the way in which the Interface will relay MIDI commands arriving at its input. Four values are possible: [**OFF, ONLY, RECHN, ALL**]. These values can cause filtering or modification of incoming MIDI commands before passing them on to the MIDI output. The Interface will *always* directly relay any Program Change, Real Time, System Exclusive, or System Common messages arriving at the MIDI input, regardless of the setting on this or any other menu option. The action otherwise occurring under the set of values listed above is independently discussed below:

OFF No incoming MIDI data is relayed to the MIDI output, except for the subset of moding commands listed above. The Interface, of course, still can send its own commands to the MIDI output (depending on the chosen Send Mode, as set in Item #3 below).

ONLY Incoming MIDI commands are relayed only if they arrive on the channel assigned under Item #1 above. All such MIDI commands are relayed intact; ie. note commands, controllers, pitch bend, etc. Any MIDI commands assigned to a different channel are filtered out unless they are in one of the categories that are always passed on, as listed above.

RECHN This functions like ONLY above (ie. MIDI commands assigned to the output channel are relayed). The difference, however, is that any non-note MIDI commands arriving under the Interface basic channel (set in the MIDI MODE menu) are now re-channelized to possess the output channel number (set in Item #1 above) and also relayed to the MIDI output. All other MIDI data (except for the special categories), as well as MIDI note data arriving on the basic MIDI channel, are filtered out. Program change data is not re-channelized, but passed intact, since it is in one of the aforementioned "special categories". The "RECHN" option may be useful when a synthesizer on the MIDI output is used as a "slave" that augments the 4-Voice operation; any controller, pitch bend, etc. data that is sent to the 4-Voice will also be sent to the output channel (ie. these commands do not have to be

duplicated). The slave synth. is still independently accessible through its own channel number; any commands sent under the output channel will be relayed intact without any Interface effect.

ALL All incoming MIDI data is relayed directly to the MIDI output with no modification (except perhaps pitch bend, which can be independently filtered via the "Xmit PB" command discussed in Item #4 below). This mode might be useful if several synthesizers (or a big multi-timbral job) are listening to the MIDI output for data unrelated to Interface operation (this isn't really needed, however, since the Interface also provides a MIDI "thru" connector).

3) SEND This item allows one to specify the types of MIDI note commands generated by the Interface unit. It accepts four values: [**MUTE**, **NORMAL**, **ECHO4V**, **OVFLOW**]. All note commands generated for the MIDI output are processed by the delay and filtering items specified elsewhere in this menu before being actually sent out. Each option available to the SEND command is independently discussed below:

MUTE No note commands are generated by the Interface for transmission through the MIDI output. The Interface will, as always, still relay incoming MIDI data according to the setting of the XMIT Mode command discussed above. Notice that this command only refers to generated *note* data; the pitch bend and/or controller data created through Auto Bend and Auto Trill are unaffected by this setting.

NORMAL All note commands arriving on the basic input channel are sent to the output channel (subject, of course, to the conditions selected in the other items of this menu).

ECHO4V This stands for "Echo 4-Voice", which essentially describes the action taken. All notes sent to the 4-Voice are also sent to the output channel. This allows one to have a "slave" synthesizer duplicate what the 4-Voice is playing, provided that a Key Priority (and/or Mono Mode) is selected in the MIDI Mode menu (otherwise the action will be identical to "NORMAL" above, except that notes falling outside the 4-Voice range will be ignored). The Transpose set in Item #6 of the MIDI mode menu is not subtracted from the note number, thus the incoming MIDI notes are transferred intact.

OVFLOW Only 4-Voice overflow notes (ie. notes sent but not sounding) are sent to the MIDI output. These are notes lying outside of the 4-Voice keyboard range, and/or notes in excess of the available 4-Voice voices (provided that a Key Priority or Mono Mode has been imposed via the MIDI Mode Menu).

4) XMIT PB This item specifies whether MIDI pitch bend data received at the MIDI input will be relayed to the MIDI output. If set ON, pitch bend data is relayed (subject, naturally, to the general restrictions on MIDI transfer imposed by the XMIT Mode of Item #2 above). If set OFF, all incoming pitch bend data is filtered out and not transmitted. Pitch bend data generated by the Interface (ie. in Auto Bend or Auto Trill) is not affected. This item was included in order to reduce the potential overload (or confusion) that could be caused by transmitting pitch-bend data to a slave synth. that is also bending under Auto Bend or Auto Trill.

5) FILL DLY This item may be set "OFF" or "ON". It refers to the output voice-filling option, which is described in detail under Item #7 below. When ON, the "Fill Delay" will insert a small delay in between the Note-on's and Note-off's that are sent to allocate all of the output voices. This delay may be necessary for some keyboards (like my Mirage), that can be confused by an extremely rapid stream of Note-on's followed immediately by a rapid stream of corresponding Note-off's. Of course, some keyboards may be unaffected, thus no delay is inserted when this item is turned OFF (operation of the MIDI output may be more prompt with this option OFF when the Fill mode is selected). If the Fill mode is not activated (under Item #7 below), the setting of the Fill Delay will have no effect.

6) FILL N This item allows one to specify the "Fill Note" sent under the output voice-filling option, which is described in detail under Item #7 below. It may take one of two values; ie. [SAME, SEMI]. When set to "SAME", the fill operation will repeatedly send the same note as the latest received in order to occupy all remaining output voices. This will work with some synthesizers (ie. the Mirage), which allocate different voices with each new MIDI command, and do not steal the same voice if the note is identical to one already sounding. If SAME doesn't work with the device attached to the MIDI output, the Fill Note may be set to SEMI. This will send a succession of ascending semitones (starting one above the last note received) to occupy all residual voices. The synthesizer on the MIDI output should be set to have a moderate-to-long attack if the Fill option is used (particularly in the "SEMI" mode), otherwise the brief notes sent to occupy the remaining voices may be heard.

7) #VOICES This item activates the output Fill mode, and allows one to specify the number of voices available in the output synthesizer. If one sets this value to zero (where the display will show "--" instead of "0"), the Fill mode is deactivated, and no extra Note commands will be sent to the output synthesizer. A number can be entered into this parameter (ranging up to 64), which will activate the output Fill mode, and specify the number of voices available in the output synthesizer. The Fill mode is an option that will allocate all available voices in the output synthesizer whenever a new note-on is sent. Its function is mainly to mute any sustained tones in the output synth. whenever a note is changed, allowing the "slave" synthesizer to directly follow what the 4-Voice is playing. If the ECHO4V mode is selected under the "Send" item (#3 in this menu), the Fill logic will assume that the output Synth. should normally have 4 Voices available (or only one if Mono mode or Unison are selected). The remaining voices will be rapidly allocated & de-allocated to mute excess sustaining tones when notes are changed. If ECHO4V is not selected under the "Send" item, the Fill logic will reallocate *all* free voices whenever a note is turned on, thereby always cutting off *any* sustains. One will have to experiment with this capability in order to ascertain its utility and learn how to use it effectively. In order to avoid having the wrong notes cut off (especially in the ECHO4V mode, where an attempt is made to exactly follow the 4-Voice), one should "slur" one's playing somewhat, and not let up old keys until the new keys are down. In addition, the way in which various synthesizers respond to the rapid note-on's & note-off's sent under this option will vary. The upper two items (#5, Fill delay and #6, Fill note) have been included to allow custom tailoring of this option to the perks of various synths. This option can introduce potential delay into the MIDI output, and might insist on creating "pop" sounds or "glitches" with some synths. The output Fill can be used with any of the MIDI output options in this menu.

8) DLY TRG This item (the "Delay Trigger") allows one to delay the sending of note-on's to the MIDI output, or impose a "minimum-key-down" constraint on notes played upon release. Its operation depends on the setting of Item #9, the Note Trigger, as described below. If notes are to be triggered on *attack*, a non-zero Delay Trigger will introduce a delay between the receipt of notes and their being sent to the MIDI output. If any notes are turned off before this delay has expired, they will not be included in the notes sent; only notes currently held down are sent to the MIDI output after the delay has expired. The delay is reset whenever new notes are hit; notes must be held down for the full delay interval for any MIDI output activity to occur. Note-Off's are, however, sent promptly with no delay. If notes are selected to trigger on key *release* (via Item # 9 below), the action is somewhat different. A non-zero delay trigger will now mean that notes must be held down for at least the delay interval in order to be queued for playing upon key-up. Any notes held down for a shorter interval will not be sent to the MIDI out. Again, the delay will reset with each new key hit, as above. The delay setting is allowed to range from 0 → 255. The delay scaling is on the order of 1/20'th of a second per Delay Trigger count, unless 255 is selected, which will will cause a huge delay on the order of minutes. This large delay might be handy if a MIDI controller is used to terminate the delay and send the notes to the MIDI output. This is accomplished by the "HOLD 2" MIDI controller (controller number 69). If this controller is sent with an "ON" value during a Trigger Delay, the delay will terminate, and the notes will be sent to the MIDI output or queued for playing upon note release (depending on the setting of Item #9, the Note Trigger). This can enable applications such as footswitch control over notes applied to the MIDI output. One final hint should be made here; the timer LED on the Interface front panel (rightmost in the row of diagnostic lights next to the display, labeled "T") will be illuminated while the Delay Trigger is active; this might be useful visual feedback for setting an appropriate delay. One must be careful, however, in that the delays used by the Auto Bend and Auto Trill will also illuminate this LED, thus this hint might not be accurate if these other options are simultaneously enabled & put under a delayed trigger.

9) NOTE ON This item allows the Note Trigger to be set. It dictates how notes will be sent to the MIDI output, and takes three values [**ATK**, **RLS-**, **RLS+**]. These are discussed independently below:

ATK This is the "normal" setting. Notes will be sent to the MIDI output when Note-on commands are received at the MIDI input (addressed, of course, to the basic Interface input channel, set under Item 1 of the MIDI MODE menu). Delay can be introduced as discussed above under Item #8.

RLS- Notes will be sent to the MIDI output when they are released at the MIDI input. The Delay Timer (Item #8 above), if active, will now function as a "minimum hold" timer. The attack and release velocities are preserved under this option; ie. the *release* velocity of each note will function as the *release* velocity at the MIDI output, while the original *attack* velocity of the note will function as the *attack* velocity at the MIDI output.

RLS+ Notes will be sent to the MIDI output when they are released at the MIDI input. The Delay Timer (Item #8 above), if active, will now function as a "minimum hold" timer. The attack and release velocities are "reversed" under this option; ie. the *release* velocity of each note now functions as the *attack* velocity at the MIDI output, while the original *attack* velocity of the note will now function as the *release* velocity at the MIDI output.

10) NOTE DUR This is the "Note Duration" parameter, and it may be set between 0 and 255. It is active if a Note on Release (ie. RLS) option was selected via the Note Trigger in Item #9 above. The Note Duration specifies the base duration of the notes that are triggered on release. It is in units of roughly 1/20'th of a second. A velocity increment can be added to this quantity, as detailed in Item #11 below. The note duration will begin timing after all keys have been released. All sounding notes will be cleared when either the note duration timer finishes, or when another note-on is received at the MIDI input. This item has no effect if notes are sent upon attack (ie. the ATK option in Item #9).

11) DUR VSCL This parameter applies a velocity-sensitive increment to the Note Duration defined above in Item #10. It may be set between 0 → 63. If zero, there is no velocity sensitivity, and the duration of released note-on's is solely determined by the setting of Item #10. As it is increased, velocity sensitivity is added to the value set in Item #10. The velocity sensitivity is defined such that a *slow* key release will yield a *longer* note duration than a quick key release; this protocol seems to be more intuitive than its opposite. The release velocity of the *last* note held down is used here, since the duration timer begins counting only after all notes have been released.

12) REC KEY# This parameter sets the input stimulus that starts recording the input notes for the internal note buffer. It ranges 0-129. The 0 setting is replaced by the characters "SOS" in the display. This setting selects the "Sostenuto" pedal (MIDI controller #66) to be the record trigger; i.e. the note buffer is recorded while the Sostenuto pedal is held down. If this parameter ranges between 1 and 128, a MIDI note is selected to provide the record trigger. The chosen note is this parameter minus 1 (since the MIDI note range is 0 through 127). If this note is hit once, the record option is activated. Notes are entered into the buffer as they are played (the buffer can store up to 251 notes). Recording stops when this key is hit again (although no more than 251 notes will be stored). One normally will set this key to be outside the range that sounds on the keyboard, in order to prevent accidental record toggling during a performance. The "Key Safe" item under the "MIDI MODE" menu allows one to specify a safety feature, where the Soft Pedal must be down while record mode is selected (or deselected). Occasionally, one may become confused as to whether the note buffer is recording or not; in these cases, hit the "Silence" button on the Interface panel; this always turns the note record off. If this item is set to its maximum (129), the record mode will never be activated.

13) OUT PLAY This parameter specifies the notes that are sent to the MIDI output. It may be set to [NTE, BUF]. The "NTE" setting is the standard option; it merely sends the notes that are being played to the MIDI output. The "BUF" setting sends notes from the note buffer to the MIDI output. This item may also be set through MIDI controller # 28. The selection of buffer notes is specified in item #14 below. All other items in this menu (i.e. notes on upon release, etc.) will function with this item set at either mode. The note buffer is remembered after power-off; i.e. the notes last entered still reside within the Interface once it is again turned on.

14) OUT ORD This item specifies the order in which buffer notes are played. It only has effect when "OUT PLAY" above is set to "BUF"; otherwise this setting is ignored. This option may be set to [FWD, RTRO, CRCL, NDWN, MAP]; each option is discussed separately below.

FWD Notes are played from the buffer in "forward" fashion; i.e. in the order in which they were entered. Each additional note-on will play the next note in the buffer. When the end of the buffer is reached, it wraps back to the beginning.

RTRO Notes are played from the buffer in "reverse" fashion; i.e. opposite the order in which they were entered. Each additional note-on will play the previous note in the buffer. When the beginning of the buffer is reached, it wraps up to the end.

CRCL Notes are played from the buffer in "Cyclic" fashion. At first, the notes are played as they were entered (i.e. FWD). When the end of the buffer is reached, notes are played opposite the order in which they were entered (i.e. RTRO). When the beginning of the buffer is again reached, notes are once more played in forward fashion, and so on....

NDWN Buffer notes are essentially played in the "forward" fashion, except the buffer increment is set by the number of notes being held down. If no notes are held down, successive note-ons will produce the next note in the buffer, as in "FWD". If one key is held down, every other buffer note will be played upon successive note-on's. If two keys are held down, every third buffer note will be played, and so on... When the end of the buffer is reached (or surpassed), it wraps to the beginning.

MAP The note number being played is used as an index into the note buffer. In other words, MIDI note #0 selects the first buffer note, note #1 selects the second buffer note, etc. If the note number is larger than the quantity of notes entered into the buffer, the buffer is basically "barberpoled"; i.e. it is repeated sequentially across the keyboard. This option allows one to completely re-define (in a repeatable fashion) the input-vs-output note relationship on the output synthesizer; i.e. key x corresponds to note y.

15) OUT VEL This parameter specifies a velocity increment that is added to the buffer pointer under all of the above options. If this item is zero (or the velocity of the key being played is very soft) no additional increment is added to the buffer pointer. If this parameter is set to a finite value, the key velocity will be proportionally added into the buffer pointer, to create a "jump" in the buffer playback. This parameter ranges from 0 to 127. Smaller values will produce a noticeable and controllable effect. Larger values will tend to randomize the buffer playback.

8) The Auto Portamento Menu

The "Auto Portamento" effect is a touch-sensitive portamento option for the 4-Voice synthesizer and/or the MIDI output (this effect can be quite fun to play with on other synthesizers). Under Auto Portamento, the "glissando" of each note can be made to depend on the note's attack velocity. The 4-Voice action of the Auto Portamento will depend on the setting of Item #4 of the 4-Voice Setup Menu (the Portamento Mode). If this is set either "OFF" or "BTH", the Auto Portamento will always occur in both keyboard halves. If Split is on, and the Portamento mode specifies a portamento split, then the Auto Portamento will only be active in the specified keyboard half. The Auto Portamento, when active, overrides any Portamento Time settings (Item #5 in the 4-Voice Setup Menu), although this should be left at zero so portamento will stop when & if Auto Portamento is turned OFF. A Note-on delay can also be imposed on the 4-Voice in this menu; this may have independent musical applications. All menu items are discussed in detail below:

1) AUTO PRT This item allows the Auto Portamento to be turned ON or OFF. When ON, an asterisk will appear in the right-hand corner of the display when the "AUTO PMNTO" entry is displayed at the top-level menu, and the Auto Portamento effect will function as configured in this menu. Auto Portamento must be turned ON for the delay feature to work (although it may be used independently by setting AFFECT to OFF in Item #4 below). Auto Portamento may also be turned ON & OFF through MIDI controller #19 (which is designated "General Purpose Controller #4).

2) AVEL SCL This item allows the attack velocity sensitivity to be set for the Auto Portamento effect. It ranges from zero (no effect) through 255 (maximum).

3) AVEL SNCE This item specifies how the Portamento effect will vary with attack velocity. It may take two values: [-, +]. If -, the glissando effect is diminished for notes that are hit hard. If +, the glissando effect is increased for notes that are hit hard.

4) AFFECT This item specifies which synthesizer combination is affected by the Auto Portamento. It takes 4 possibilities; [**4VCE**, **OUT**, **BOTH**, **OFF**]. "4VCE" is the 4-Voice only, "OUT" is the MIDI output, "BOTH" is both, and "OFF" produces no Auto Portamento effect (although the Delay effect can still be used; see below).

5) DELAY This item specifies a delay which is applied before each note is sent to the 4-Voice and MIDI output. It ranges from 0 → 63, and is calibrated in units of roughly 1/75'th of a second. Auto Portamento must be ON (via Item #1 above, or MIDI controller #19) for the delay to operate. This delay feature was initially introduced to compensate any lag in the 4-Voice portamento hardware; ie. if the portamento responded slowly, one would want to wait a bit in between assigning the portamento value and sending the note command. As it turned out, this was unnecessary. The 4-Voice portamento circuitry works very rapidly, and Auto Portamento will function perfectly with this Delay set to zero (ie. off). If a nonzero delay is inserted, however, weird (and *fun*) things will happen with note timing. The delay will be obvious, and a rapid stream of input notes (ie. from hitting a quick chord) will be "flammed" into a series of individual notes separated identically by the delay interval. If one "goes wild" and rapidly plays a series of notes/chords, the Interface will slowly/methodically play each note out at constant tempo (set by the delay interval). This creates a neat delay effect, where the results of one's

playing are stretched into a sequence that continues well after the playing is finished. The notes sent to the MIDI output are also delayed, and staggered with the 4-Voice notes; i.e. a 4-Voice note sounds first, followed by an output note, followed by another 4-Voice note, etc... One word of caution applies, however; when the Interface is waiting for this delay to expire, it is *not* servicing its MIDI buffer, or doing much of anything else. Incoming MIDI data can fill this buffer up in foreground, and if a large delay setting slows down its servicing, a MIDI overflow can result. Things are really not too bad... the MIDI buffer available in the Interface is fairly huge, and it's mainly things like pitch bend or mod. wheel commands that could have any chance of clogging it significantly; use these controllers judiciously if the delay is active.

9) The Auto Bend Menu

The Auto Bend is a sweep effect. Its output may be assigned to the 4-Voice pitch bend, the 4-Voice VCF, an arbitrary controller or pitch bend on the MIDI output, and/or the Tone-Volume-Balance processor. The Auto Bend is triggered by MIDI commands sent to the basic Interface input channel (as set by Item #1 in the MIDI Mode menu). It can be triggered immediately at Note On (with optional delay), at Note Off (with optional minimum-hold requirements), or via MIDI controller #82 (General Controller #7). It can also be triggered by hitting a specified key, subject to the "Key Safe" protection. Two parameters determining the Auto Bend effect are velocity sensitive; the direction of the bend (up/down) and the rate of bend. When the bend occurs on release of all notes, the bend direction can be determined by the last note's attack or release velocity (as set via Item #3 below), while the bend rate is generally determined by the last release velocity. When the bend is triggered on note attack (or delayed note attack), both parameters are determined by the attack velocity of the last note hit, unless a note has been released before the bend starts, in which case this release velocity will determine the bend rate (and bend direction if set to RLS in Item #3). The allowed range of the bend effect can be limited through the "Maximum Swing" parameter in the menu below (Item #7). The net effect of pitch bending on the 4-Voice is also dependent upon the Pitch Bend Sensitivity setting (Item #3) in the 4-Voice Set-up Menu (as the VCF range is dependent on the VCF SWP setting). After the Bend is complete, the swept parameters (ie. 4-Voice pitch/VCF, MIDI pitch bend, MIDI controller, Tone-Volume-Balance) remain at their extreme values until either a new note is hit, a new patch is loaded, or the Interface is reset via the "Silence" button or other procedures. Each item in the Auto Bend menu is described in detail below:

1) AUTO BND This is the switch to turn the Auto Bend effect ON or OFF. When OFF, the other items in this menu will have no effect. When ON, an asterisk appears at right in the AUTO BEND item of the top-level menu, and the Auto Bend will operate as configured. The Auto Bend may also be enabled and disabled via MIDI controller #80 (General Purpose Controller #5); this may be useful in sequencer situations, where the bend effect is to be dynamically turned on and off in a composition (note that another MIDI controller can be used to *trigger* the bend effect, as detailed under Item #10 below).

2) BEND DIR This quantity determines the velocity sensitivity of the bend-up or bend-down switch. It runs $0 \rightarrow 127$. A large value of this parameter will favor bend-up (ie. a very small velocity will be needed to bend down), while a small value of this parameter will favor bend-down (ie. a very fast velocity will be needed to bend up). Needless to say, an average between both effects will be at around 64, although this does depend to some extent upon one's playing style, and any velocity scaling imposed by the MIDI keyboard controller.

3) DIR ON This item determines whether the bend direction will be decided by the attack velocity of the last note hit, or the release velocity of the last note let up. This choice is only relevant when the Auto Bend is configured to occur upon last note release (as determined by the settings of Items #9 & #10 below), otherwise the attack velocity of the last note hit generally determines bend direction (unless a note has been released before the bend starts, in which case this release velocity will determine the bend direction). This item takes two possible values, ie. [ATK, RLS]. As one would assume, ATK indicates that bend direction is determined from the most recent attack velocity, while RLS indicates that bend direction arises from the last release velocity.

4) RVEL_SCL This item determines the velocity scaling on the rate of the Auto Bend effect. It ranges from 0 → 127. Values near zero will exhibit very little velocity scaling, while larger values will bend much more rapidly, depending upon the velocity value. Whenever the Auto Bend is triggered on note release, the rate scaling is determined by the release velocity of the last key let up. Otherwise, the rate scaling is determined by the attack velocity of the last key hit (unless a key was released prior to the start of the bend).

5) RVEL_SNCE This item selects the sense of the velocity scaling on the bend rate. It can be set either [-, +]. If +, larger velocities will generate a faster bend rate, while if -, smaller velocities will generate a faster bend rate.

6) SWP_BIAS This item allows one to specify a "Sweep Bias". It runs 0 → 255. This is essentially a base sweep bias for the Auto Bend effect; the velocity increment specified under Items #4 & #5 is added to this to generate the net Auto Bend sweep. A low Sweep Bias value (ie. 0 or so), will exhibit extreme velocity sensitivity (provided that the Rvel. Scale in Item #4 is set sufficiently large). A higher value of Sweep Bias will show less velocity sensitivity, and exhibit faster sweeps.

7) MAX_SWING This is a clamp on the maximum swing of the Auto Bend sweep. It runs 0 → 63. A value of 63 will allow the effect to sweep the full range of pitch bend (ie. 2 MIDI bytes worth) or controller value (ie. 1 MIDI byte, or up to a value of 127). A smaller value will allow proportionally less sweep. One uses this parameter (along with sensitivity parameters in the receiving synthesizer) to adjust the full range of the Auto Bend effect. This parameter will affect the range of Auto Bend exhibited by both the 4-Voice & MIDI Output.

8) RATE_SET This is yet another parameter that specifies the base rate of the Auto Bend. It runs 0 → 254, and is somewhat similar in effect to the Sweep Bias set under Item #6 (this parameter actually sets the clocking rate, while the Sweep Bias sets the bend increment per sweep update). It seems best to keep this at very low values, otherwise the sweep is very slow. A "quantized" stepping effect might be obtained by making the Sweep Bias (Item #6) and the Rate Set values both fairly large; this can result in large pitch (or controller) steps being taken at discretely spaced intervals, producing an ascending or descending "staircase" effect. The Rate Set value is calibrated in units of roughly 1/300'th of a second.

9) MIN_HOLD This item (Minimum Hold) allows one to impose a minimum key-down requirement on Auto Bend operation. It runs 0 → 255, and is calibrated in units of roughly 1/20'th of a second. The action taken under this setting is heavily dependent upon the setting of the Trigger Source (Item #10) below. Details are given with the Trigger Source description. The zero value of Minimum Hold is replaced with the characters (RLS), since a zero Hold value will cause the Auto Bend to trigger on last-note release if the Trigger Mode is set to NRM. Note that the "T" LED to the right of the display will be illuminated while the minimum hold delay is active; this might be useful in setting up patches (but beware, since delays in Auto Trill and the MIDI output can illuminate this LED as well).

10) TRIGGER This allows one to select a Triggering protocol for the Auto Bend effect. It can take three values, ie. [NRM, QUE, CTLR]. The action taken under each value is dependant upon the setting of the Minimum Hold (Item #9) listed above. Each possibility is discussed below. Note that the MIDI controller that triggers the Auto Bend effect is # 82 (ie. General Purpose Controller #7). A trigger source can also be derived from a specified key on the input MIDI controller; see item #14 below.

NRM This is the "Normal" mode. If the Minimum Hold (Item #9) is zero, the Auto Bend will be triggered after the last note held down is released. If the Minimum Hold is non-zero, the Auto Bend will be triggered after the notes have been held down for the minimum hold interval (the timer will reset with each new note hit; notes have to be held down for at least the full timer interval for Auto Bend to trigger). If no notes were held long enough, the Auto Bend will not be triggered. The Auto Bend will always trigger under NRM mode immediately upon receipt of a non-zero value for MIDI Controller #82 (ie. General Purpose Controller #7) or key-trigger note-on.

QUE This is the "Queue" mode. If the Minimum Hold (Item #9) is set to zero, the Auto Bend will not be triggered with any note-on or note-off activity. If, however, the Minimum Hold is non-zero, the Auto Bend will be triggered promptly after all notes have been released *only* if the set of notes were held down for at least the Minimum Hold interval. This enables a selective activation of the Auto Bend effect; notes have to be held down sufficiently long for the bend to be activated at the final note release. The Auto Bend will always trigger under QUE mode immediately upon receipt of a non-zero value for MIDI Controller #82 or key-trigger note-on.

CTLR This is the "Controller" mode. It enables the Auto Bend effect to be selectively triggered by the MIDI Controller. Auto Bend will not be triggered directly via any MIDI note activity regardless of the Minimum Hold setting. MIDI Controller #82 (and key-trigger note-on) still have effect, however. If the Minimum Hold is set to zero, the Auto Bend will trigger promptly upon receipt of a non-zero Controller #82 value (or key-trigger note-on). If the Minimum Hold is non-zero, Auto Bend will *not* trigger immediately after receipt of Controller #82 (or key-trigger note-on), but will *queue* its activity, causing the Auto Bend to trigger after all notes have been released.

11) OUT CTL# This specifies the destination of the Auto Bend effect for the MIDI output. If it is zero (causing the display to show "PB" rather than zero), the Auto Bend will command the pitch bend over the assigned MIDI output channel. If it is non-zero (it ranges up to 121), the assigned value denotes the commanded MIDI controller number addressed by the Auto Bend. If the MIDI output is deactivated through Item #12 below, this setting will have no effect.

12) CONTROL This item determines devices that are commanded with the Auto Bend effect. It can take three values: [**4VCE**, **OUT**, **BOTH**]. Each are described below:

4VCE The Auto Bend will control only the 4-Voice pitch.

OUT The Auto Bend will control only the MIDI output (as set through Item #11 above).

BOTH The Auto Bend will control both the 4-Voice and the MIDI output.

13) INIT VAL This item runs from 0 through 127, and sets the baseline value of the MIDI controller selected through item #11 above (if no MIDI controller has been selected, this item has no effect). The controller is set to this value when a key is hit, the Interface is reset (through front panel switches or MIDI commands), or a new patch is

loaded. One must beware with some synths, however; this controller value may be remembered by the synthesizer (even after it is made to load other patches!). This can create problems, especially with things like MIDI Volume, which might be set differently after a patch with Auto Bend acting on it has been loaded.

14) TRG KEY# This item allows one to select a note on the input MIDI controller to function as a trigger for the Auto Bend (the action of this trigger depends on the setting of Item 10 above). This item will run in value from 0-128. If it is set below 128, it specifies the MIDI note # which will act as the trigger source. If it is set at 128, no MIDI triggers will be produced in this fashion. The KEY SAFE option of the MIDI MODE menu (item #9) will require that the Soft pedal is depressed when the selected trigger key is hit in order to produce a valid trigger event.

15) VCF FEED This item specifies the amount of Auto Bend feed into the 4-Voice VCF (the amount of effect is also subject to the VCF SWP setting in the 4-Voice Setup Menu). It runs 0 (no effect) through 255.

16) VOL FEED This item specifies the amount of Auto Bend feed into the volume control of the on-board audio processor. It runs 0 (no effect) through 255. This option can be used to generate keyboard-driven volume sweeps.

17) BAL FEED This item specifies the amount of Auto Bend feed into the balance control of the on-board audio processor. It runs 0 (no effect) through 255. This option can be used to generate keyboard-driven dynamic panning effects.

18) BAS FEED This item specifies the amount of Auto Bend feed into the bass control of the on-board audio processor. It runs 0 (no effect) through 255. This option can be used to generate keyboard-driven bass swells.

19) TRB FEED This item specifies the amount of Auto Bend feed into the treble control of the on-board audio processor. It runs 0 (no effect) through 255. This option can be used to generate keyboard-driven brightness shifts.

10) The Auto Trill Menu

The Auto Trill is another sweep effect. Its output may be assigned to the 4-Voice pitch bend, the 4-Voice VCF, an arbitrary controller or pitch bend on the MIDI output, and/or the Tone-Volume-Balance processor. The Auto Trill is triggered by MIDI commands sent to the basic Interface input channel (as set by Item #1 in the MIDI Mode menu). It can be triggered immediately at Note On (with optional delay), at Note Off (with optional minimum-hold requirements), or via MIDI controller #83 (General Controller #8). It can also be triggered by hitting a specified key, subject to the "Key Safe" protection. Two parameters determining the Auto Trill effect are velocity sensitive; the speed at which the Trill grows, and the rate of Trilling itself. When the Trill occurs on release of all notes, the Trilling rate will be determined by the last note's attack velocity. The Trill's growth speed is generally determined by the last release velocity. When the Trill is triggered on note attack (or delayed note attack), both parameters are determined by the attack velocity of the last note hit (unless a note has been released before the trill starts, in which case this release velocity will determine the Trilling growth speed). The allowed range of the Trill effect can be limited through the "Maximum Swing" parameter in the menu below (Item #8). The net effect of pitch Trilling on the 4-Voice is also dependent upon the Pitch Bend and VCF Sensitivity settings in the 4-Voice Set-up Menu. After the trill growth is complete, the swept parameters (ie. 4-Voice pitch/VCF, MIDI pitch bend or MIDI controller, Tone-Volume-Balance) will trill at their extreme values until either a new note is hit, or the Interface is reset via the "Silence" button or other procedures. The trill waveform (triangle, pulse, sweep) and pulse width (or wave asymmetry) may be user specified or set dynamically through MIDI controllers. Each item in the Auto Trill menu is described in detail below:

1) AUTO TRL This is the switch to turn the Auto Trill effect ON or OFF. When OFF, the other items in this menu will have no effect. When ON, an asterisk appears at right in the AUTO TRILL item of the top-level menu, and the Auto Trill will operate as configured. The Auto Trill may also be enabled and disabled via MIDI controller #81 (General Purpose Controller #6); this may be useful in sequencer situations, where the Trill effect is to be dynamically turned on and off in a composition (note that another MIDI controller can be used to *trigger* the Trill effect, as detailed under Item #10 below).

2) AVEL SCL This quantity determines the velocity sensitivity of the Trilling rate. It runs 0 → 127. A large value of this parameter will favor a rapid Trill, while a small value will favor a slower Trill. Trilling rate has a different effect on a MIDI controller (unless both Auto Bend & Auto Trill are active over the same channel), since, as mentioned above, the Trill effect will generally be a simple sweep over a Controller. A faster trilling rate (in this case) will cause a faster sweep.

3) AVEL SNCE This item selects the sense of the velocity scaling on the Trilling rate. It can be set either [-, +]. If +, larger velocities will generate a faster Trill rate, while if -, smaller velocities will generate a faster Trill rate.

4) RVEL SCL This item determines the velocity scaling on the growth of the Auto Trill effect. It ranges from 0 → 127. Values near zero will exhibit very little velocity scaling, while larger values will cause the trill to grow much more rapidly, depending upon the velocity value. Whenever the Auto Trill is triggered on note release, the rate scaling is determined by the release velocity of the last key let up. Otherwise, the rate scaling is

determined by the attack velocity of the last key hit (unless a key was released prior to the start of the Trill).

5) RVEL SNCE This item selects the sense of the velocity scaling on the Trill growth. It can be set either [-, +]. If +, larger velocities will generate a faster Trill growth, while if -, smaller velocities will generate a faster Trill growth.

6) SWP BIAS This item allows one to specify a "Sweep Bias". It runs $0 \rightarrow 255$. This is essentially a base sweep bias for the Auto Trill effect; the velocity increment specified under Items #4 & #5 is added to this to generate the net Auto Trill sweep (by "sweep" here, we mean the increment in trill magnitude at each pitch change, thus this correlates with the trill "growth"). A low Sweep Bias value (ie. 0 or so), will exhibit slower trill growth and extreme velocity sensitivity (provided that the Rvel. Scale in Item #4 is set sufficiently large). A higher value of Sweep Bias will show less velocity sensitivity, and exhibit faster trill growth. This quantity can be dynamically adjusted via MIDI controller # 22.

7) RTE BIAS This item allows one to specify a Trilling "Rate Bias". It also runs $0 \rightarrow 255$. This is essentially a base rate for the Auto Trill effect; the velocity increment specified under Items #2 & #3 is added to this to generate the net Auto Trill rate. A low Rate Bias value (ie. 0 or so), will exhibit fast trills and extreme velocity sensitivity (provided that the Avel. Scale in Item #2 is set sufficiently large). A higher value of Rate Bias will show less velocity sensitivity, and exhibit slower trills. This quantity can be dynamically adjusted via MIDI controller # 20.

8) MAX SWING This is a clamp on the maximum swing of the Auto Trill sweep. It runs $0 \rightarrow 63$. A value of 63 will allow the effect to sweep the full range of pitch Bend (ie. 2 MIDI bytes worth) or controller value (ie. 1 MIDI byte, or up to a value of 127). A smaller value will allow proportionally less sweep. One uses this parameter (along with sensitivity parameters in the receiving synthesizer) to adjust the full range of the Auto Trill effect. This parameter will affect the range of Auto Trill exhibited by both the 4-Voice & MIDI Output.

9) MIN HOLD This item (Minimum Hold) allows one to impose a minimum key-down requirement on Auto Trill operation. It runs $0 \rightarrow 255$, and is calibrated in units of roughly 1/20'th of a second. The action taken under this setting is heavily dependent upon the setting of the Trigger Source (Item #10) below. Details are given with the Trigger Source description. The zero value of Minimum Hold is replaced with the characters (RLS), since a zero Hold value will cause the Auto Trill to trigger on last-note release if the Trigger Mode is set to NRM. Note that the "T" LED to the right of the display will be illuminated while the minimum hold delay is active; this might be useful in setting up patches (but beware, since delays in Auto Bend and the MIDI output [along with the Trilling operation itself] can illuminate this LED as well).

10) TRIGGER This allows one to select a Triggering protocol for the Auto Trill effect. It can take three values, ie. [NRM, QUE, CTLR]. The action taken under each value is dependant upon the setting of the Minimum Hold (Item #9) listed above. Each possibility is discussed below. Note that the MIDI controller that triggers the Auto Trill effect is # 83 (ie. General Purpose Controller #8). A trigger source can also be derived from a specified key on the input MIDI controller; see item #14 below. The operation of this logic is entirely analogous to the triggering of the Auto Bend described with the previous menu.

NRM This is the "Normal" mode. If the Minimum Hold (Item #9) is zero, the Auto Trill will be triggered after the last note held down is released. If the Minimum Hold is non-zero, the Auto Trill will be triggered after the notes have been held down for the minimum hold interval (the timer will reset with each new note hit; notes have to be held down for at least the full timer interval for Auto Trill to trigger). If no notes were held long enough, the Auto Trill will not be triggered. The Auto Trill will always trigger under NRM mode immediately upon receipt of a non-zero value for MIDI Controller #83 (ie. General Purpose Controller #8) or key-trigger note-on.

QUE This is the "Queue" mode. If the Minimum Hold (Item #9) is set to zero, the Auto Trill will not be triggered with any note-on or note-off activity. If, however, the Minimum Hold is non-zero, the Auto Trill will be triggered promptly after all notes have been released *only* if the set of notes were held down for at least the Minimum Hold interval. This enables a selective activation of the Auto Trill effect; notes have to be held down sufficiently long for the Trill to be activated at the final note release. The Auto Trill will always trigger under QUE mode immediately upon receipt of a non-zero value for MIDI Controller #83 or key-trigger note-on.

CTLR This is the "Controller" mode. It enables the Auto Trill effect to be selectively triggered by the MIDI Controller. Auto Trill will not be triggered directly via any MIDI note activity regardless of the Minimum Hold setting. MIDI Controller #83 (and key-trigger note-on) still has effect, however. If the Minimum Hold is set to zero, the Auto Trill will trigger promptly upon receipt of a non-zero Controller #83 value (and key-trigger note-on). If the Minimum Hold is non-zero, Auto Trill will *not* trigger immediately after receipt of Controller #83 (and key-trigger note-on), but will *queue* its activity, causing the Auto Trill to trigger after all notes have been released.

11) OUT CTL# This specifies the destination of the Auto Trill effect for the MIDI output. If it is zero (causing the display to show "PB" rather than zero), the Auto Trill will command the pitch bend over the assigned MIDI output channel. If it is non-zero (it ranges up to 121), the assigned value denotes the commanded MIDI controller number addressed by the Auto Trill. If the MIDI output is deactivated through Item #12 below, this setting will have no effect.

12) CONTROL This item determines devices that are commanded with the Auto Trill effect. It can take three values: [4VCE, OUT, BOTH]. Each are described below:

4VCE The Auto Trill will control only the 4-Voice pitch.

OUT The Auto Trill will control only the MIDI output (as set through Item #11 above).

BOTH The Auto Trill will control both the 4-Voice and the MIDI output.

13) INIT VAL This item runs from 0 through 127, and sets the baseline value of the MIDI controller selected through item #11 above (if no MIDI controller has been selected, this item has no effect). The controller is set to this value when a key is hit, the Interface is reset (through front panel switches or MIDI commands), or a new patch is loaded. One must beware with some synths, however; this controller value may be

remembered by the synthesizer (even after it is made to load other patches!). This can create problems, especially with things like MIDI Volume, which might be set differently after a patch with Auto Trill acting on it has been loaded.

14) TRG KEY# This item allows one to select a note on the input MIDI controller to function as a trigger for the Auto Trill (the action of this trigger depends on the setting of Item 10 above). This item will run in value from 0-128. If it is set below 128, it specifies the MIDI note # which will act as the trigger source. If it is set at 128, no MIDI triggers will be produced in this fashion. The KEY SAFE option of the MIDI MODE menu (item #9) will require that the Soft pedal is depressed when the selected trigger key is hit in order to produce a valid trigger event.

15) VCF FEED This item specifies the amount of Auto Trill feed into the 4-Voice VCF (the amount of effect is also subject to the VCF SWP setting in the 4-Voice Setup Menu). It runs 0 (no effect) through 255.

16) VOL FEED This item specifies the amount of Auto Trill feed into the volume control of the on-board audio processor. It runs 0 (no effect) through 255. This option can be used to generate keyboard-driven tremolo effects.

17) BAL FEED This item specifies the amount of Auto Trill feed into the balance control of the on-board audio processor. It runs 0 (no effect) through 255. This option can be used to generate keyboard-driven dynamic panning effects.

18) BAS FEED This item specifies the amount of Auto Trill feed into the bass control of the on-board audio processor. It runs 0 (no effect) through 255. This option can be used to generate keyboard-driven bass pulsing effects.

19) TRB FEED This item specifies the amount of Auto Trill feed into the treble control of the on-board audio processor. It runs 0 (no effect) through 255. This option can be used to generate keyboard-driven brightness modulation.

20) TRL WDTH This item specifies the duty factor (i.e. pulse width) of the Auto Trill pulse (or asymmetry of the triangle waveform). It ranges from 0 (perfectly symmetrical) through 255. The setting of this item will also affect the trill frequency; adjust items 6 and 7 to compensate, if needed. The trill width can also be set dynamically via MIDI controller #21.

21) WV SEL This item allows one to specify the trill waveform. The possibilities are [PULSE, TNGLE, SWEEP]. The meanings are obvious; a pulse wave for the standard "trill" effect, a triangle wave for a smoother change (the limited resolution can make this a bit rough and "staircase-like" if it is swept over a wide range), or a single sweep (as in the Auto Bend). The trill frequency may change when the waveform is switched between triangle and pulse; it can be compensated by items 6 and 7 if needed. The trill waveform can be dynamically changed via MIDI controller #23 (waveforms can be changed in the midst of a trill, giving interesting effects).

22) POLARITY This item specifies the polarity of the trill waveform. It has 3 possibilities; [±, UP, DWN]. The "±" setting produces a bipolar waveform that moves both up and down (excepting the sweep, which will move only up). The "UP" setting will produce a waveform that goes positive, while the "DWN" setting will produce a negative-going waveform.

11) The Tone-Volume-Balance Menu

This menu allows one to set parameters governing the action of the on-board audio processor (an external stereo audio signal [i.e. the 4-Voice audio output] can be routed through this processor via the phone jacks at the rear of the Interface unit). Baseline tone/volume/balance settings and optional key-velocity sensitivities can be specified via the items listed below. Crossfeeds into the Tone-Volume-Balance processor from the Auto Bend and Auto Trill effects can be specified in the Bend and Trill menus. Rapidly changing or wide-ranging control changes may cause a perceptible "thump" in low-level audio signals; beware! All of the below items (barring the velocity sensitivities) may be dynamically altered through MIDI controllers; see the next section for details.

1) VOLUME This parameter specifies the baseline volume. It ranges 0 through 255. The LED meters on the front panel show the input level of the audio signal, and are not affected by this volume setting.

2) VOL VSCL This parameter specifies the velocity sensitivity (i.e. last attack velocity from the input MIDI controller) on the volume. Hitting a key harder will cause a louder signal. This item runs from 0 (no effect) through 127.

3) BALANCE This item specifies the right/left channel balance. It runs 0 through 255. A value of 128 is center; 0 is hard left and 255 is hard right.

4) BAL VSCL This parameter specifies the velocity sensitivity (i.e. last attack velocity from the input MIDI controller) on the balance. Hitting a key harder will move the audio balance further right. This item runs from 0 (no effect) through 127.

5) BASS This parameter specifies the bass response. It runs 0 through 255. A value of 130 will specify flat bass response.

6) BASS VSCL This parameter specifies the velocity sensitivity (i.e. last attack velocity from the input MIDI controller) on the bass. Hitting a key harder will produce a bassier sound. This item runs from 0 (no effect) through 127.

7) TREBLE This parameter specifies the treble response. It runs 0 through 255. A value of 125 will specify flat treble response.

8) TRB VSCL This parameter specifies the velocity sensitivity (i.e. last attack velocity from the input MIDI controller) on the treble. Hitting a key harder will produce a brassier sound. This item runs from 0 (no effect) through 127.

9) INPUT This parameter specifies whether the input signals are in stereo or bridged to mono. It then logically takes two possibilities: [**STEREO**, **MONO**]. The mono bridging is handy if one is using panning effects.

10) ST ENHCE This parameter specifies the selection of a stereo enhance effect. It may be turned "ON" or "OFF". When on, a very noticeable left/right separation enhancement appears, along with a small rise in volume.

11) LD CMP This is a standard loudness compensation effect, as on a home stereo. When ON, a signal set to low volume (via item #1 and associated controllers) will be boosted in both bass and treble. The effect diminishes as the volume setting increases.

12) MIDI Implementation

The MIDI implementation adopted by the 4-Voice Interface is quite thorough. MIDI commands interpreted by the Interface are addressed to the basic MIDI channel, as set via Item #1 of the MIDI Mode menu (barring any generalities allowed via the OMNI setting; ie. Item #2 of the MIDI Mode menu). Certain MIDI controller commands can change patch menu values; ie. most settings in the 4-Voice Set-up menu, certain items in the MIDI Mode menu, and the Auto Bend, Auto Trill, and Auto Portamento enable flags are all alterable under MIDI commands. The values displayed by the patch editor will also change accordingly. This will not happen instantaneously, however; if the patch editor is displaying the item being changed via MIDI, it will not show the new value unless the display is refreshed (ie. display another menu item, then go back to the original, or use an editing function). Another important fact to keep in mind is that any altered patch values (whether changed via the patch editor *or* MIDI controller command) are saved to memory in the "." keypad save sequence outlined in Chapter 2. If one saves a patch that has been subject to controller commands that changed menu values, the altered values will be saved with the patch.

The Interface accepts MIDI commands through all phases of its operation. Regardless of the display & menu status (ie. top level menu, lower level menu, editing, etc.), the Interface will respond as commanded. There is no discrete "play" mode that is invoked to handle MIDI.

The text below goes through all accepted MIDI commands one-by-one. In some cases, little detail is given because the action taken depends on patch menu settings which were exhaustively described in the preceding chapters.

Note ON/OFF These commands play notes on the 4-Voice and/or are selectively routed to the MIDI output. The interpretation of Note commands depends *heavily* upon the setting of the MIDI Mode, 4-Voice Setup, and Set MIDI Out menus, as described in the preceding chapters. Note commands also trigger and resolve Auto Portamento, Auto Bend, & Auto Trill effects.

Pitch Bend MIDI Pitch Bend commands are automatically sent to the 4-Voice. Both bytes of the MIDI pitch bend message are used in the Interface; a 12-bit pitch bend capability exists. The pitch bend sensitivity is set through Item #3 of the 4-Voice Setup Menu (a range of roughly ± 1.5 octaves is possible). No decrease in resolution of pitch bend is caused by specifying a smaller range; the full 12 bits are always used.

Program Change Upon receipt of a program change command ranging from 0 through 98, the Interface will load the corresponding patch (these range 1 \rightarrow 99 in the display). Before the new patch is loaded, the 4-Voice, MIDI output, and various logic are cleared (an action similar to hitting the "Silence" button). When the Interface loads a patch through MIDI, the new patch number & title will be shown on the display, regardless of the state of the patch editor. This is merely a visual display (one is not blown back to the top level editor), which is cleared promptly after any keypad response. The abrupt appearance of this display can be somewhat disconcerting while editing a patch item; the cursor will continue flashing (as in normal editing), but the characters will be replaced by the patch number:title. Again, this is only a "phantom" display, and the original display re-appears after any keypad key is hit. Ideally, one will not be editing a patch when MIDI program change commands are being sent, thus this effect may be irrelevant.

Upon receipt of a program change command ranging from 100 \rightarrow 115, the selected 4-Voice program is switched to the specified value minus 99 (ie. 4-Voice programs 1 \rightarrow 16

are directly accessible). The current Interface patch is otherwise retained (although Item #2 of the 4-Voice Setup Menu is altered to agree with the new 4-Voice program).

Any MIDI program change commands specifying programs exceeding 115 are ignored (a program change of 99 is also ignored).

MIDI Controllers A host of MIDI controllers are available to augment Interface operation. Although many details were presented earlier in the discussion of the corresponding Menu items, everything dealing with MIDI controller response is summarized below:

64 - Sustain Pedal

This functions as usual; any notes currently down continue to sound (along with any new notes) until the sustain off command is issued. This is the only MIDI controller that is not restricted to be commanded over the basic input channel when the OMNI item (#2 in the MIDI Mode list) is set to X-CTRLS or MULTI (this is done because the sustain operation is correlated with note commands, which are accepted over several MIDI channels in these modes).

123 - All Notes Off

If the sustain pedal is not down, all notes on the 4-Voice (and MIDI output if selected) will be turned off. If the sustain pedal is down, the All Notes Off is queued for execution immediately upon release of the sustain. The MIDI channel transmitted with this command must *always* agree with the basic input channel (set in the MIDI Mode menu), regardless of the OMNI setting. The All Notes Off is ignored if a different channel is used.

All of the controllers listed below change items in the 4-Voice Setup menu. They are incorporated to allow a dynamic change in 4-Voice configuration and operation without the necessity of loading an entirely new patch. This may be quite useful in sequencer operation, and can save on needed patch locations. The VCF sweeps can be useful in performance. Beware, since these commands actually change the settings in the 4-Voice Setup menu; saving a patch after it has been altered will save the controller-commanded values. The Official MIDI name of each controller is given to the right of each heading (in parenthesis).

1 - VCF Cutoff Frequency (Modulation Wheel)

3 - VCF Cutoff Frequency (Undefined)

The VCF cutoff frequency may be set via either of these controllers, depending on the setting of the "VCF CTRL" item in the 4-Voice Setup Menu. If the controller is chosen to be "MOD", the modulation wheel (controller #1) will adjust the filter cutoff. If the controller is chosen to be 3, the modulation wheel will have no effect, and controller #3 will adjust the filter cutoff. The range of filter sweep is set by the VCF Sensitivity adjustment, which can be changed in the 4-Voice Setup Menu (Item #12) or through MIDI controller #9.

5 - Portamento Time (Portamento Time)

The value sent with MIDI controller #5 is loaded into the portamento time register, which runs 0 → 63 (larger values are saturated at 63). This will not have immediate effect in the 4-Voice if the portamento mode is in the **OFF** setting (this can be changed via MIDI controller #65).

9 - VCF Sensitivity (Undefined)

The value sent with MIDI controller #9 sets the range of the VCF sweep (on controllers #1 or 3, or bend/trill effects).

12 - Octave Switch (Undefined)

The value sent with MIDI controller #12 shifts the 4-Voice oscillators up and down in octave intervals. A value of 0 introduces no shift, 1 shifts down by 2 octaves, 2 shifts down by 1 octave, and 3 or greater shifts up by one octave.

13 - Assign Mode (Undefined)

The value sent with MIDI controller #13 changes the setting of the "Assign" switch on the 4-Voice keyboard module. A value of 0 sets assign to R/C, a value of 1 sets assign to CONT, a value of 2 sets assign to RESET, and any value greater than 2 sets assign to CONT.

16 - Split Mode (General Controller #1)

The value sent with MIDI controller #16 is loaded into the split mode register. The Split Mode runs 0 → 3, and corresponds to the values [**OFF, 2-2, 3-1, 1-3**] as described under Item #7 of the 4-Voice Setup Menu. Any value exceeding 3 is assumed to be 1, ie. "2-2".

17 - Unison Mode (General Controller #2)

The Unison is turned ON if the value sent with controller #17 is non-zero, or OFF if zero.

18 - Reassign Mode (General Controller #3)

The Reassign is turned ON if the value sent with controller #18 is non-zero, or OFF if zero.

65 - Portamento Mode (Portamento On/Off)

The value sent with MIDI controller #65 is loaded into the portamento mode register. The allowed values run 0 → 3, and correspond to [**OFF, BTH, UPR, LWR**], as described under Item #4 of the 4-Voice Setup Menu. Any value exceeding 3 is assumed to be 1, ie. "BTH". No change will be heard immediately if the portamento time is zero (this can be dynamically adjusted via MIDI controller #5).

94 - Pitch Bend Sensitivity (Celeste/Detune Depth)

The value sent with MIDI controller #94 is loaded into the sensitivity register for the 4-Voice pitch bend. This is allowed to range 0 → 127 (ie. the full MIDI resolution). The full range is covered, however; one unit sent via the MIDI controller is equal to 2 units changed by the patch editor (which allows this parameter to range 0 → 255).

The controllers listed below change parameters in the Audio Processor (i.e. set by the Tone-Volume-Balance menu). They are incorporated to allow a dynamic change in Audio parameters without the necessity of loading an entirely new patch. This may be quite useful in sequencer operation & performance (i.e. where a parameter is swept on a continuous controller), and can save on needed patch locations. Beware, since these commands actually change the settings in the Tone-Volume-Balance menu; saving a patch after it has been altered will save the controller-commanded values. The Official MIDI name of each controller is given to the right of each heading (in parenthesis).

4 - Treble (Foot Switch)

The value sent with MIDI controller #4 adjusts the Treble level on the audio processor.

7 - Volume (Volume)

The value sent with MIDI controller #7 adjusts the Volume level on the audio processor.

8 - Balance (Balance)

The value sent with MIDI controller #8 adjusts the Left/Right channel balance on the audio processor.

11 - Bass (Expression)

The value sent with MIDI controller #11 adjusts the Bass level on the audio processor.

24 - Loudness Compensation (Undefined)

The value sent with MIDI controller #24 activates/deactivates the loudness compensation on the audio processor. A value of 0 turns loudness off, while any other value turns it on.

25 - Stereo/Mono Input Select (Undefined)

The value sent with MIDI controller #25 sets the inputs of the audio processor to be either stereo or bridged to mono. A value of 0 selects stereo, while any other value chooses mono.

26 - Stereo Enhance (Undefined)

The value sent with MIDI controller #26 activates/deactivates the stereo enhance effect on the audio processor. A value of 0 turns the enhance off, while any other value turns it on.

The controllers listed below allow one to dynamically enable, disable, & alter several of the Interface effects. Again, these options are incorporated to allow a dynamic change in the Interface configuration and operation without the necessity of loading an entirely new patch. This may be quite useful in sequencer operation, and can save on needed patch locations. Beware, since these commands actually change the settings in the corresponding menus; saving a patch after it has been altered will save the controller-commanded values.

19 - Auto Portamento Enable (General Controller #4)

The value sent with MIDI controller #19 will turn the Auto Portamento effect ON or OFF. A zero value turns it OFF, while any non-zero value turns it ON.

80 - Auto Bend Enable (General Controller #5)

The value sent with MIDI controller #80 will turn the Auto Bend effect ON or OFF. A zero value turns it OFF, while any non-zero value turns it ON. This controller does not actually trigger the effect (see controller #82).

81 - Auto Trill Enable (General Controller #6)

The value sent with MIDI controller #81 will turn the Auto Trill effect ON or OFF. A zero value turns it OFF, while any non-zero value turns it ON. This controller does not actually trigger the effect (see controller #82).

20 - Trill Frequency (Undefined)

The value sent with MIDI controller #20 basically sets the "RTE BIAS" parameter (#7) of the Auto Trill menu. This allows one to dynamically alter the trill frequency during a trill.

21 - Trill Pulse Width (Undefined)

The value sent with MIDI controller #21 basically sets the "TRL WIDTH" parameter (#20) of the Auto Trill menu. This allows one to dynamically alter the pulse width or waveform asymmetry during a trill (although the trill rate will also be somewhat affected). A value of zero creates a symmetric waveform.

22 - Trill Amplitude (Undefined)

The value sent with MIDI controller #22 basically sets the "SWP BIAS" parameter (#6) of the Auto Trill menu. This allows one to dynamically alter the trill amplitude (although the trill frequency will be somewhat affected as well).

23 - Trill Waveform (Undefined)

The value sent with MIDI controller #23 basically sets the "WV SEL" parameter (#21) of the Auto Trill menu. This allows one to dynamically alter the waveform used during a trill. A value of 0 selects a pulse wave, 1 selects a triangle, and 2 selects a sweep.

27 - Autochord Mode (Undefined)

The value sent with MIDI controller #27 allows one to set the mode of the Autochord function (see the MIDI MODE menu). A value of zero turns Autochord off. A value of 1 turns Autochord on and sets the Interval to OCT. A larger value also turns Autochord on, but sets the Interval to USR.

28 - Play notes/Buffer (Undefined)

The value sent with MIDI controller #28 allows one to dynamically toggle the MIDI output between playing input notes or playing buffer notes (it sets the "OUT PLAY" parameter (#13) in the SET MIDI OUT menu). A value of zero selects the input notes, while any other value selects the note buffer.

The following MIDI controller commands trigger preset effects, according to the current configuration of their corresponding menus. No patch values are changed. If the effect is not enabled, these commands have no response.

66 - Load Note Buffer (Sostenuto Pedal)

If the "REC KEY" option (#12) is appropriately set in the SET MIDI OUT menu, a nonzero value sent with MIDI controller #66 will put the Interface into record mode, so that the internal note buffer is loaded. This can be inhibited by the Key Safe feature, as discussed below. Record mode is turned off when a zero value is sent over this controller.

67 - Key Safe (Soft Pedal)

The value sent with MIDI controller #67 can be used as a safety feature to inhibit spurious Bend/Trill triggering or Note Buffer recording. When selected (under the "KEY SAFE" option (#7) of the MIDI MODE menu), the Soft Pedal must be down (i.e. this controller must have transmitted a nonzero value) during the appropriate trigger event, otherwise the trigger is ignored.

69 - Note Trigger At MIDI Output (Hold 2)

A non-zero value associated with MIDI controller #69 will cause any queued notes to be routed to the MIDI output. A zero value of controller #69 will have no effect. The interpretation of this command depends *heavily* on the settings of the Delay Trigger and Note Trigger (Items #8 and #9) configured in the Set MIDI Out Menu.

82 - Auto Bend Trigger (General Controller #7)

A non-zero value associated with MIDI controller #82 will trigger the Auto Bend effect. A zero value of controller #82 will have no effect. The interpretation of this command depends *heavily* on the settings of the Minimum Hold and Trigger Source (Items #9 and #10) configured in the Auto Bend Menu.

83 - Auto Trill Trigger (General Controller #8)

A non-zero value associated with MIDI controller #83 will trigger the Auto Trill effect. A zero value of controller #83 will have no effect. The interpretation of this command depends *heavily* on the settings of the Minimum Hold and Trigger Source (Items #9 and #10) configured in the Auto Trill Menu.

The following items are all "Channel Mode" messages. They can change the protocol under which the Interface receives MIDI messages (ie. the OMNI flag can be altered), thus should be used with caution. All of these commands will cause the Interface to clear all notes, pitch bend, software flags, etc. (analogous to the "Silence" button). The MIDI channel sent with these commands must agree exactly with the basic input channel (set in the MIDI Mode menu) regardless of the OMNI value. If the channel is different, the command is ignored. Again, these options are incorporated to allow a dynamic change in the Interface configuration and operation without the necessity of loading an entirely new patch. This might be useful in sequencer operation, and can save on needed patch locations. Beware, since these commands actually change settings in the MIDI Mode menu; saving a patch after it has been altered will save the controller-commanded value.

124 - Set OMNI OFF

The OMNI parameter (Item #2 of the MIDI Mode Menu) is set to OFF. All subsequent MIDI commands must be issued over the assigned basic input channel. The value associated with this controller command is ignored.

125 - Set OMNI to X-CTRLS

The OMNI parameter (Item #2 of the MIDI Mode Menu) is set to X-CTRLS. All subsequent MIDI note commands & MIDI sustain will be accepted regardless of their transmitted channel number. All other subsequent MIDI data, however, must be issued over the assigned basic input channel for Interface recognition. The value associated with this controller command is ignored.

126 - Set MONO Mode and Unison to ON

The 4-Voice is put into MONO ON mode (see Item #3 of the MIDI Mode Menu), and the 4-Voice Unison is activated. This is a quick & dirty way of dynamically putting the 4-Voice into a mono mode (with key priority imposed) using all available voices. The value associated with this controller command is ignored.

127 - Set MONO Mode and Unison to OFF

The 4-Voice is put into MONO OFF mode (see Item #3 of the MIDI Mode Menu), and the 4-Voice Unison is deactivated. This is a quick & dirty way of reversing the above command and restoring the 4-Voice polyphony. The value associated with this controller command is ignored.

13) System Exclusive Commands

A System Exclusive protocol was designed into the Interface to enable patches to be dumped & loaded via MIDI. This is useful for several potential reasons; ie. backing up the 99 resident patches (the on-board Lithium battery is supposed to last over 20 years, but who knows what actually might happen), allowing more than 99 patches (99 is quite a few for just controlling the 4-Voice, but this box is capable of doing many other things), and eventually building a Macintosh-based patch editor/librarian which would be easier to deal with than the keypad-based software resident in the Interface. The Sys-Ex protocol was designed to resemble the usual Oberheim standard (to some extent), thus some of the more powerful "commercial" patch librarians might be coaxed into backing this device up & re-loading it when necessary.

The patch data is dumped & loaded at 4 bits per MIDI byte. The 4 low bits are sent first, followed by the 4 high bits (all patch items are assumed to be 8 bits long). The patch data begins with a byte bearing the patch number, followed by a 9-byte ASCII patch title. The contents of the 6 menus follow sequentially after this (as laid out in the text above; remember that each menu item occupies one byte). A total of 105 bytes are thus sent with each patch, expanding into 210 bytes over MIDI (not counting Sys. Ex. headers and trailers). If all 99 patches are dumped simultaneously (by addressing patch #100), 10,395 bytes will be sent (resulting in a transfer of 20,790 MIDI bytes).

The MIDI stream sent to the Interface must agree *exactly* with the protocol specified below, otherwise the command will be ignored. All numeric quantities referenced in the discussion below are in hexadecimal.

In order to request a patch dump from the Interface, the following dialog is used:

Computer requesting Dump

F0 10 77 00 yy pp F7

The hex "10" specifies an Oberheim product, the hex "77" specifies the Joe Paradiso Interface (hopefully this code isn't used by Oberheim; I picked it for good luck), and the "00" specifies that I'm *requesting* a patch dump. The "yy" can be anything; Oberheim had this as a channel number, but I ignore it. The "pp" specifies the patch number (0 through 49 here, unless the patch #100 dump-all option is used). If it likes your message, the Interface will respond:

Interface giving Dump

F0 10 77 01 yy pp [Patch Data; 210 bytes per patch, 20580 bytes for all] F7

The Interface's return message has a nearly analogous header. As a bit of additional information, however, the "yy" parameter is now the value currently assigned to the Interface's basic MIDI input channel (under the resident patch). The "pp" is the patch number that is being dumped (hopefully it agrees with the number actually requested). The patch data follows (4 bits at a time). The message finishes with an End of Sys Ex. command (ie. the F7). If the Interface didn't like the original dump request message, it will ignore it and not respond.

The messages more-or-less reverse if a patch-load command is sent to the Interface. The protocol is listed below:

Computer Commanding Memory Load

F0 10 77 01 yy pp [Patch Data; 210 bytes per patch, 20580 bytes for all] **F7**

All items are as described above; the "yy" doesn't matter. "pp" is the patch number to be written over (if "pp" is 100, the next software revision will load all 99 patches in succession). The Interface assumes that the data byte count is correct. Excess bytes will be ignored, but the patch may be only partially overwritten if insufficient bytes are sent. Again these MIDI bytes are only 4 data bits at a time, organized in low/high order.

If the Write Protect is set to "Y" (a keypad function on the ÷ key at the top-level edit), the Interface will not save the patch data into memory, bring up the write protect warning on its display, and complain with a "NAK" message:

Interface NAK Complaint

F0 7E yy 7E pp F7

Again, "yy" is the basic MIDI input channel, and "pp" is the patch number that was commanded to be overwritten. The "7E" indicates a NAK; this message is a standard in sampler dumps.

If the Interface was not under write-protect, and overwrote the patch, it responds with an "ACK" message of the form:

Interface ACK Response

F0 7E yy 7F pp F7

This message is essentially the same as the NAK, except a "7F" replaces the "7E". The Interface won't respond at all (and the patch memory will remain unchanged) if the original memory load command was found to be faulty (ie. had a bad header or lacked the "7F" End of Sys Ex. trailer).

Two additional System Exclusive Patch load commands are possible. If "pp" is set to 101, the load/dump commands will access the "currently loaded" patch buffer. In this case, all parameters of the patch will be instantly active, and no nonvolatile memory locations will be overwritten (i.e. one must manually save the patch by hitting the "." key). This option is useful for quickly debugging a patch, and playing with parameters from a patch editor program. If "pp" is set to 127 for the command requesting a dump from the Interface, a lengthy dump will indeed be provided, containing the patch data structures; i.e. the menu formats and titles, all item names, maximum/minimum limits, options, etc. This dump is used by the patch librarian to initialize and formulate its variables and options.

14) Patch Editor/Librarian Program

A handy patch editor/librarian program was concocted to enable quick backup of the Interface unit's patch memory, and aid in manipulating and experimenting with patch variables. The program is called "4-V Librarian", and it's written in MIDIBASIC on the Macintosh. Launch by double-clicking its icon. This program requires the file "4-V DS" to be resident somewhere, preferably in the same folder as "4-V Librarian" (otherwise it asks for the location). To use this routine, connect a MIDI output from the Macintosh into the Interface's MIDI input, and connect a MIDI output from the Interface to a MIDI input on the Macintosh. Make sure that the MIDI input on the Macintosh (i.e. MIDI Time Piece or JamBox) has its echo off, and doesn't route its MIDI input to the MIDI output (otherwise terrible MIDI feedback can occur, locking everything up). When the Librarian is launched, it prints a few things out, then (after a second or two) builds its menu bar. I'll blaze quickly through the menu options below...

Upon launching the Librarian program, the Macintosh assumes 99 "vanilla" patches exist, all labeled "<<BLANK>>". Real patches can be brought in by loading them from a file, uplinking them from the Interface, and/or editing the patch variables. The "File" menu enables one to access and write data files that contain Interface patches. The "Open/Load All" patch item brings up a file dialog box, requesting an input file specification. When a file containing patches is opened, a running dialog is provided as they are brought in from disk, then the "Titles" display is shown; i.e. all 99 patch names are printed in a window, with the patch considered to be the "currently edited" highlighted in bold. The patch data from the file is now in Macintosh memory and can be edited and/or installed into the Interface.

If one selects "Open/Load Patch", a similar scenario ensues; one selects an input file to open, then (after a several second delay) all patches are also displayed. Here, however, the user is prompted to enter the patch number to be loaded at the bottom of the window (patch numbers are printed to the left of patch titles). After entering a number and hitting return, the user is prompted to specify the patch number that it will overwrite in the Macintosh memory (this can be any of the 99 possible patches). After again hitting return, the window will be redrawn with the patch titles in Macintosh memory, but now the newly loaded patch is seen to be resident, wiping out what was there before. This option affords a handy way to steal separate patches from various files, and combine them into another file (or Interface load).

The "Save" option will write the patches resident in Macintosh memory into a data file; the user is prompted to specify a filename, then the patches are written out. "Quit" will stop the program and exit to the Finder.

The top items in the Edit menu function as they always do (for individual character strings), but may be of limited utility in the Librarian program. The two lower groups of items are of more interest. The "Copy Patch" and "Paste Patch" options allow one to copy the current patch to transient storage, then paste it into another file (if new data was loaded), or into another patch (if the current patch number was changed), thereby juggling patch order around. The "Copy Page" and "Paste Page" commands work at a finer level. They are only available when editing an Interface menu page, as will be outlined below. All data from the current page is copied into transient storage, and can be pasted into the analogous page of a different patch. This allows one to take a certain setup on part of a patch (i.e. the Auto Trill effect), and paste it into the Auto Trill page of another patch. The data on other patch pages is unaffected. One must be advised that these commands do not use the Macintosh clipboard (as do the standard cut/copy/paste, which are available at the top of the menu), but use special internal storage, thus the data can't be exported to the Scrapbook, etc.

The "Up/Downlink" menu controls communication with the Interface over the MIDI line. "Uplink All" will load all 99 patches from the Interface into the Macintosh memory. "Uplink Patch" will load a single patch; the user is prompted to enter the patch to load in the Interface, and specify where it will go in the Macintosh. The "Uplink Current Patch" option will load the patch that is currently active in the Interface (and has not necessarily been saved) into the currently edited patch in the Macintosh. The "Downlink" options are analogous. "Downlink All" loads the Interface with all 99 patches from the Macintosh. "Downlink Patch" will download a single patch from the Macintosh to the Interface; the user is prompted to enter the source and destination patch numbers. "Downlink Patch & Load" will downlink a single patch (as above), plus also load it as the currently active patch in the Interface. If the "Write Protect" is set (under the ÷ key) on the Interface, patches won't be able to be downloaded into the patch memory, thus the patch editor will complain bitterly, and the data transfer operation will be halted. The option "Downlink Current Patch" doesn't touch the patch memory, however (thus doesn't care about the write protect). This option will load the currently-edited patch on the Macintosh into the currently active patch in the Interface. This is the easiest way to try a patch out; merely hit the keyboard shortcut combination "⌘," answer the keyboard prompt, and the patch is active in the Interface!

The "Patch" menu provides a means of displaying and editing patch data. The "Show Title" option will bring up the window that shows the resident patch numbers and titles, as discussed above. The next three items allow one to change the designation of "currently edited" patch. If "Load Patch" is picked, the user is prompted to type in the number of the patch to be considered currently edited (its title will be made bold if the "Titles" display is selected). The "Next Patch" and "Previous Patch" items will scroll forward by one patch, or backward by one patch (the keyboard shortcuts for these options provide a quick means of perusing sequential patch data).

The following 8 items will cause data from the "currently edited" patch to be displayed. This data can be edited and changed by selecting the "Edit Item" option at the bottom of the menu (or by typing "⌘E"). The "Title/Patch #" option is the simplest; this displays the patch title, and allows it to be changed through the Edit Item option. The other 7 items are all menus from the Interface. If one of these is selected, the data from the corresponding menu page (for the currently-edited patch) is displayed. This display format gives the item name, followed by its numeric value in the patch, followed by the alphanumeric option name (provided that the menu item is one that takes options rather than a number). These quantities can be changed through "Edit Item"; a prompt will appear below the data list asking for the item number to edit. Merely enter this index, then hit return. For numeric arguments, a dialog box is produced that shows the lower and upper limits. For character arguments, a list of all possible values is presented. In both of these cases, merely type in the desired value, then hit "OK" or carriage return. The new value will be shown in the updated menu item list.

The "Print" menu allows the user to print data from a patch or group of patches. If "Titles" is selected, the Title page is printed (showing the names of all resident patches). If "Page" is selected, the currently edited menu page is printed. If "Patch" is selected, all pages of the currently-edited patch are printed. If "All Patches" is selected, the user is prompted for a start and end patch number; all patches within this range are printed. The "FF after Pages" option, when checked, will force a form feed to occur after each menu page is printed, resulting in one menu page per physical page.

The "MIDI" menu allows one to set communication defaults. The middle 3 options set the Macintosh MIDI interface data transfer rate. The "1 Mhz" setting works for most gear that I've used, and is the default. The serial port can be specified on which the MIDI interface resides (through the top 2 items); the modem port is the default (rather than the printer port, which can also be chosen). When it is selected, the last item clears all the Macintosh MIDI buffers. If a load or save was aborted, this option should be selected before attempting more such things, so the Macintosh will continue with a clear slate.