OB•Mx

User's Guide

Oberheim®

CREDITS

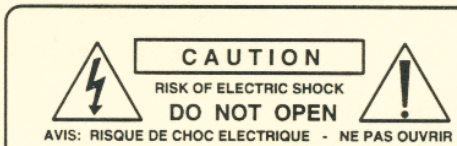
- WRITTEN BY WARREN SIROTA
- LAYOUT & DESIGN BY DENISE FORET
- OB-MX IS A PRODUCT OF OBERHEIM, A DIVISION OF GIBSON
- OB-MX DESIGNED BY DON BUCHLA
- FOR MORE INFORMATION, CONTACT

Oberheim
2230 Livigston Street
Oakland, CA 94606

Phone.....(510) 536-8600
Fax .....(510) 261-1708
Technical Support.....(800) 279-4346

WARNINGS

There are thousands of parts inside, but none are user servicable, so kindly refrain from removing the covers.
Do not take the OBMx swimming or play it in the rain.
Custom designed for Oberheim by Don Buchla
Manufactured by Oberheim, Oakland, California



OB-Mx User's Guide

Contents

SECTION 2: REFERENCE

SECTION 3: APPENDICES & CHARTS

OB-Mx User's Guide

Foreword

Remember the days when men were men and synthesizers were synthesizers? Knobs, switches, and such stuff were in abundance, imagination could run rampant, sounds were unabashedly electronic.

These days our musical lives center around four button interfaces, canned sounds, cryptic displays, and tedious instruction manuals. Interaction is just a fond memory, and the concept of actually playing a responsive electronic instrument seems to have died along with Flash Gordon.

Problem back then was that while you could get into pretty deep space, you could never get back to where you'd been yesterday—synths, with their infinite possibilities, were conspicuously short on memory.

So we had this crazy idea. What if we took the sounds of the 70's, the technology of today, combined them with controls up the wazoo, and an accessible user interface? Could be interesting.

Welcome to the OB Mx.

Thanks to the members of the development team - programming by Andy Corless and Lynx Crowe, with assistance from Greg Higgs and Chuck Carlson. Technical support from Keith Edwards, Mark Leas, Mike Lyon, Kim Flint, Huck Phomma, and Pree Boonkert. Sound design and beta testing by Jay Cloidt, Gerry Basserman, Malcolm Doak, Roger Powell, Richard Bugg, Joel Davel, Andrew Schlesinger, Mike Lee and Ezra.

The support of Zeta, G-WIZ Labs and Gibson Guitars kept this development in high gear. Special thanks to Tom Oberheim, who pioneered many of the basic concepts, Keith McMillen, who masterfully coordinated the development scene, and Warren Sirota, who skillfully implemented this vital link between designer and performer.

Don Buchla
Berkeley, 1994

OB-Mx User's Guide

Section 1

Introduction

Oberheim®

CHAPTER 1

How To Use This Manual

Welcome to the world of the Oberheim **OB-Mx** synthesizer. We'd like to make learning the instrument as fun and easy for you as possible, and enable you to discover your own sounds as quickly as possible. To this end, we've organized the manual into several sections:

- *Foreword*, a brief message from the designer
- *Quick Start*, where you'll setup the instrument and learn its basic concepts and operations
- *Explorations*, which presents a few carefully-chosen hands-on exercises to familiarize you with the most useful **OB-Mx** features
- *Reference*, where you can find detailed information on any function or parameter
- *Appendices*, full of technical information, are included at the back
- The *Front Panel Instant Index*, on the very last page of this manual, is a uniquely-convenient index into the Reference section made possible by the OB-Mx design philosophy of providing a dedicated button or knob for as many parameters and functions as possible. If you want information about how any button or knob works, just find it on the diagram and turn to the page number printed there

KEY POINT: As you traverse the *Quick Start* and *Explorations*, pay special attention to the **KEY POINTS** sections. They should make it easier for you to learn the instrument.

We recommend that everyone go through the *Quick Start* and *Explorations* sections in detail—we think you'll find it fun and well worth the short investment of time. It's not necessary that you go through all of the *Explorations* before you explore on your own, but we believe that you'll find them quite useful.

CHAPTER 2

Quick Start

2.1 Basic Setup

▶ Connections

The basic connections for using your **OB-Mx** are quite simple—power, MIDI In and Audio Out or headphones.

1. Plug the female end of the supplied power cord into the **OB-Mx** and the male end into the proper electrical outlet.
2. The MIDI In port is not absolutely necessary in order to get a sound from the synthesizer—as you'll see shortly, there are self-running patches that create music without any external inputs at all—but most people will want to play **OB-Mx** sounds from an external MIDI keyboard, drum pad controller, guitar controller or other controller. Use a MIDI cord to connect the MIDI Out port on your controller to the MIDI In port on the back of the **OB-Mx.**
3. In order to listen to the **OB-Mx**, either plug in a standard pair of stereo headphones (with a 1/4" stereo phone plug) into the **Phones** jack on the instrument's front panel, or connect the **Main Outputs** on the back panel to a mixer, amplifier, preamplifier or home stereo system using 1/4" phone cords. If you are not using the **OB-Mx** in stereo, use either one of the main outputs

   Now you're all wired up and ready to roll! There are many other connection options—see *The Back Panel* later in this chapter for more information.

▶ Testing The Basic Setup

Before you start exploring the instrument, it's a good idea to make sure that everything is working properly. This is easy to do, as follows:

1. Turn on the power to the **OB-Mx** and then to your audio system. The display should light up and display some parameters, which we'll talk about shortly.

2. Turn the **Volume** knob all the way down. In the **INSTRUMENTS** module just above the display, press the **Single** button. The cursor in the display will be blinking under the number of an Instrument. (If you've been doing some unauthorized button pushing and the Instrument parameters won't appear in the display, then good for you! Press the **Prgrm** button under the display and that should get you the Instrument display). Turn the **Data Adj** knob just under the display until the number 239 appears, with the name *Dance 2*. Press the **Trigger** button and release it—this instrument is a "self-playing" instrument that will continue sounding until another Instrument is selected. Gradually turn the **Volume** knob until you can hear some sound.

Troubleshooting

If you reach the top volume and you still don't hear anything through your amplifier or mixer, then try plugging a pair of headphones into the Phones jack. If you can't hear any sound, then something is wrong. Please call Oberheim Technical Support at the number in the front of this manual. If you *can* hear sound through the headphones, but not through your speakers, then a setup problem most likely exists in the audio path from the **OB-Mx** output to your speakers. Possible causes are:

- The **OB-Mx** is not plugged into the correct mixer channel.
- The **OB-Mx** mixer channel either is muted or has the **Volume** turned down.
- Another mixer channel is soloed, muting the **OB-Mx** channel.
- The main mixer outputs are turned down.
- The mixer or amplifier is not powered on.
- The mixer is routed to a preamplifier or home stereo system and isn't selected as the sound source.

If you check for these conditions and none of them apply, then call Oberheim Technical Support at the number in the front of this manual, or your dealer.

4. Now that you're sure the audio path is working, it's time to test MIDI triggering. With the cursor still under the Instrument number, turn the **Data Adj** knob on click clockwise to select Instrument 1, *Pie & Oh*. Turn on your MIDI controller if it's not already on and play a note. You should hear a sound.

Troubleshooting
If you do not get sound from your OB-Mx at this point, check for the following:

- A bad MIDI cable, or a cable not connected properly. Check the connection again. If it looks right, try substituting another cable.
- The controller may be transmitting on a MIDI channel that the OB-Mx is not set to respond to. The solution is to change the Mode to Omni. Press the **Cursor >** button under the **Data Adj** knob until the blinking cursor is under the word "Poly." Turn the **Data Adj** knob clockwise, and the Mode will be set to Omni. Try playing a note on your controller now.
- If you're using a MIDI switching unit, it may not be programmed to route the MIDI streams properly or may actually be malfunctioning. Connect the controller and the OB-Mx directly, bypassing the switcher for now.

If you check for these conditions and none of them apply, then contact Oberheim Technical Support at the number in the front of this manual, or your dealer.

Congratulations! Your OB-Mx setup is now up and running correctly, and you can start exploring the instrument and having fun.

2.2 Analog Synthesis Concepts

▶ *Basic Analog Synthesis*

The OB-Mx is based on the classic analog subtractive synthesis model (*see Figure 2.1*). In this model, a sound source rich in overtones (a *voltage-controlled oscillator*, or *VCO*) is put through a filter (*VCF*) to subtract unwanted frequencies. The filter's output is then put through a voltage-controlled amplifier, which changes levels rapidly to give each note a characteristic *amplitude envelope*, expressed in terms of attack, decay and release times, plus a sustain level (*see Figure 2.2*).

For instance, a drum-like sound would typically have a fast attack and then decay quickly to a residual level, and would then gradually fade-out with a fairly long release time. A pipe organ would have a fairly slow attack, no decay to speak of, would sustain as long as the key was held down, and then

would fade fairly quickly but not instantaneously when the key was released. For further information about envelopes, see the *Envelope Generators* chapter in the *Reference* section.

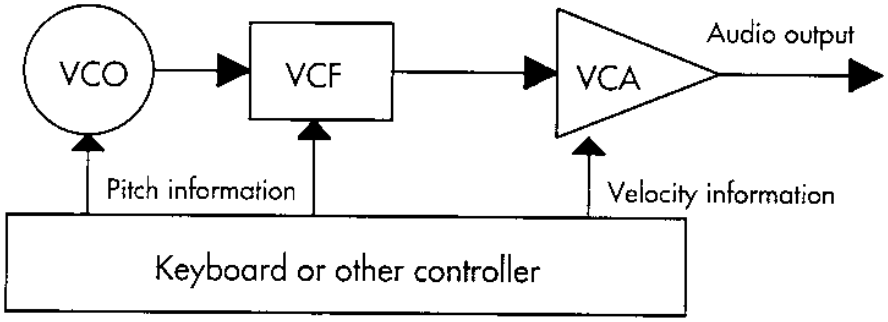


FIGURE 2.1
The basic signal path for traditional subtractive synthesis.

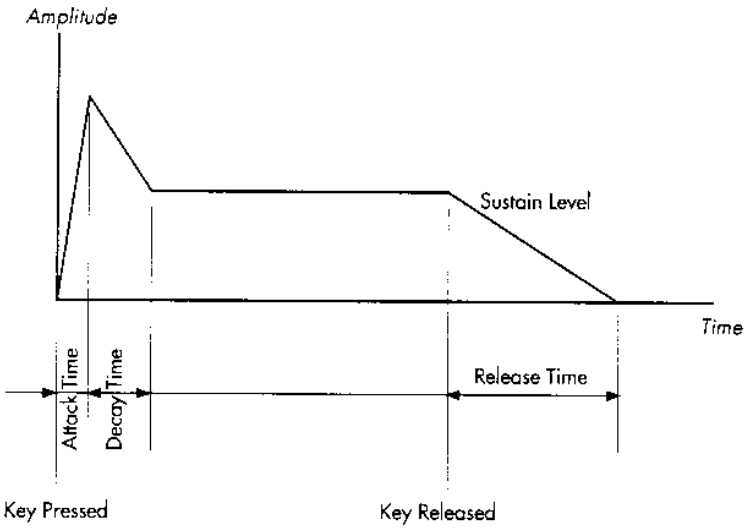


FIGURE 2.2
A simple amplitude envelope.

As shown in *Figure 2.1*, the pitch information supplied by the keyboard is most commonly used by the VCO to determine what fundamental frequency it should oscillate at, and by the VCF to affect the filter's cut-off or center frequency (more about this to follow). The velocity information from the controller is usually used to determine the sustain level of a sound, and also often to affect attack times. However, these are far from the only possibilities for using pitch and velocity information. The Matrix capabilities of the OB-Mx allow you to use this information to affect most synthesizer parameters, and also to alter the VCO, VCF and VCA parameters with other information.

▶ OB-Mx Analog Synthesis

The components shown in Figure 2.1 are the essential building blocks of sound in the **OB-Mx**, but there are more of them, they have more parameters and outputs, and they can be interconnected in convoluted networks to create thick, complex sounds with behaviors so varied that they could almost be called personalities.

The full complexity of an **OB-Mx** Instrument is shown in the *Voice Architecture* diagram on page C-3. If you don't understand it right now, don't worry—you will, and in a surprisingly short time.

To begin with, each note played on the **OB-Mx** can use two oscillators instead of just one, and each of the oscillators is capable of putting out three types of waves simultaneously, each with a different harmonic content, which you can mix and match to your liking.

Each note can also make use of two special filters. The first filter is the **MM Filter**, a low-pass filter designed to recreate the sounds of the venerable **Mini-Moog®** synthesizer. The output of this filter can be mixed with the three outputs of the **OB Filter**, a "state-variable" filter which simultaneously produces low-pass, high-pass and band-pass outputs. A **Noise Source** can create pink noise (noise which has equal power in all frequency bands) which is especially useful for creating percussive effects, and a **Pan** control in the final stage can position the note wherever you want it in the stereo spread of the main outputs. And none of this is static—just about every parameter can be varied during the course of each note by the 3 Low Frequency Oscillators and 4 Envelope Generators, as well as by external MIDI Continuous Controllers.

We'll delve into all of this quite soon, and you'll begin to realize that **OB-Mx** sounds achieve complexity and interest by connecting many simple components in complex ways.

To add to the possibilities, these Single Instrument definitions can be combined into **Multiple Instruments** (also called **MultiPrograms** or simply **Multis**). These **Multis**, which you might think of as "performance presets," layer and combine individual instruments with their own levels, panning,

transposition and the like. A single MIDI note could, for example, trigger 4 single instruments simultaneously, creating a super-thick sound from 8 oscillators with up to 24 waveforms being combined at once.

2.3 Basic Operations

▶ *Front Panel Operations*

The goal of this section is to familiarize you with the way in which the front panel of the OB-Mx works so that you can confidently push buttons, explore and modify Instruments, save your changes and just generally poke around. We'll do this in an ad-hoc, hands-on way, with no attempt to be complete—that's left for the Reference section later in this manual. Right now, the aim is to have fun and learn the basics.

The Front Panel

The front panel consist of an LCD display and a lot of knobs and switches, logically grouped into *modules*. The modules are all labeled, with names like VOLTAGE CONTROLLED OSCILLATORS, VOLTAGE CONTROLLED FILTERS, and INSTRUMENTS.

The module directly under the display controls many system-level operations like storage and editing global parameters, and is called the CONTROL module (see Chapter 6).

It's Safe To Experiment

Editing is completely safe on the OB-Mx. When you pick an Instrument, a copy of it is loaded into a temporary memory area called the *Edit Buffer*. Whenever there's a chance of losing your edit changes or overwriting existing data, you are asked whether you want to complete or cancel the operation and given the opportunity to save the Edit Buffer to any storage location you prefer. You even get to see the names of the Instruments that you might want to overwrite. And the **Compare** button lets you easily compare your edited sounds to the ones that have been saved into memory.

So poke around and make a lot of changes, and just **hit the Escape button to avoid saving anything you don't really like, and you won't hurt anything. Save changes you *do* want to keep before powering down or changing Instruments** (see *Exercise 3: Saving and Naming Your Work*, later in this chapter).

The **OB-Mx** comes with 128 Single Instruments in user-programmable RAM locations 1-128 and 128 more in permanent ROM locations 129-256. The lower 128 Instruments are duplicates of the permanent set in ROM, so you can freely overwrite them without fear of losing anything. The same situation is true for Multis.

Using The Buttons And Display

If you're used to working with other synthesizers, it may be helpful to start out by learning the differences between the convenient, sensible operations of the **OB-Mx** and the confusing, counter-intuitive operations of almost everything else (okay, so we're a little biased).

The first thing to get over, if you're used to working with other synthesizer panels, is the concept of hidden, hierarchical menus. On the **OB-Mx**, most parameters are accessed directly by pressing a front-panel button. The leftmost buttons in most modules will call a set of parameters into the display. For instance, pressing the button marked "2" in the **Voltage Controlled Oscillators** module in the upper-left part of the panel (we'll call this "the VCO 2 button" in the future for convenience, and we'll refer to other buttons similarly), brings the parameters for VCO 2 into the display. The button itself starts flashing, to let you know exactly which set of parameters is displayed. Try it.

There are 3 types of buttons to be aware of:

- *Utility Buttons*, like the **Tune** VCO's button you used while getting started
- *Selection Buttons*, like the **VCO 2** button you just pressed
- *Parameter Toggles*, which turn options on or off for selected objects

The Utility Buttons perform actions. In addition to **Tune**, they include all the buttons in the **CONTROL** module.

The Selection Buttons are the buttons that change the display contents and flash when pressed. They are located on the left in each module, and include all the numbered buttons and the buttons labeled **MM, OB, Single, Multiple** and **Disp.**

When you press a Selection Button, it makes every knob and button in its module apply to the object you've selected. For instance, if you press the **VCO 2** button, then every knob and Parameter Toggle button in the **VCO** module will affect just VCO 2.

KEY POINT: The flashing buttons tell you what is showing in the display.

The Parameter Toggles are the third type of button, and they affect (and reflect) parameters that can be either on or off. For instance, when you press the VCO 2 button, one or more of the waveform buttons in the VCO module is likely to light up. This shows which of the oscillator outputs are being used. To shut one off, simply press a lit button. The sound of most patches will probably get thinner when you do this. To turn another waveform output on, click an unlit button to light it up. The sound of most patches will probably get thicker when you do this. Parameter Toggles (with the exception of the LFO Waveform buttons) are not mutually exclusive, as Selection Buttons are. It's perfectly permissible to have all, none or any combination of the waveform outputs active at once.

One other button can be especially useful as you explore th OB-Mx—the **Trigger** button in the ENVELOPE GENERATORS module. Pressing this button has the same effect as sending a middle C (MIDI note 60) to the synthesizer from an external controller. The corresponding Note Off message is sent when you release the button. It's very convenient when you're programming and testing sounds, especially if your controller isn't right next to the OB-Mx.

KEY POINT: Press the **Trigger** button at any time to hear the current Instrument..

Exercise 1: Deconstructing An Existing Sound

In order to get warmed up and used to moving around the control panel and changing things, lets take apart an existing sound. In fact, let's start with a sound that's already familiar to you from the Setup section—Instrument 239, *Dance 2*. It's easiest to work with a self-playing sound like this, so you can hear the effects of changes as you make them.

1. First, press the **Single** button, then the **Program** button and then turn the **Data Adj** knob until Instrument 239 is selected. Press **Trigger**. You should hear it droning away. Let's try and shut it up.
2. The first thing to do is to try and shut off any waves that the oscillators are generating. Press the VCO 1 button. You'll see that the **Pulse Wave** button and the **triangle wave** button are both lit. Press each one to turn these functions off. You'll immediately hear less sound, but not silence.

3. Now let's get rid of the next component of the sound. Press the VCO 2 button. Again, you'll see that both the Triangle Wave and Pulse Wave buttons are lit. Press each one to silence that oscillator output.

4. You should still hear kind of a hissing sound. That suggests that the Noise source is being sent through one or both of the filters (see the Signal Flow diagram on page C-3). You can't shut off the Noise source, but you can set the amount of Noise that each filter accepts to 0. Here's how:

   A) Press the MM button. Note the Noise level in the display (indicated by "N:") is set to 25.

   B) Start turning the **Noise** knob slowly to the right, while watching what happens to the Noise parameter in the display. The way this knob works is typical of the way all the front panel knobs work, and it's important to understand. Turning the knob starts changing the parameter from it's current setting, rather than setting it to an absolute value corresponding to the knob's position. So, if the Noise knob is turned 3/4 of the way up before you load the Instrument and the value of the MM filter's Noise parameter is 25, then turning the knob all the way up will only result in a Noise parameter of about 40 out of 100. This allows smooth variation of any parameter from it's currently set point, but has the disadvantage that you might have to use the following technique to dial in the full range of any knob:

KEY POINT: In order to extend the range of a knob control to it's maximum, twist the knob all the way up and then all the way down. Then set it to the value you desire.

   C) Since our goal is to shut off the noise, turn the Noise knob all the way to the right and then all the way to the left. This should eliminate the hiss.

   D) Press the OB button. Note that the Noise level in the display is 0, so there's no noise leaking through that filter

5) You will still hear some sound from the **OB-Mx**. Turn the **Master Level** knob in the Final Mix module all the way off and then up to boost the sound. Since we've shut off the oscillators and silenced the noise source, what could that UFO-like sound be? If you're new to analog synthesis, then you probably have no idea of the answer. If you've been around this type of synthesis for a while, then you know that analog filters can create a ringing type of sound without any input if their resonance is set high enough.

   Indeed, pressing the **OB** and **MM** buttons and examining the display will show you that both resonances levels are set fairly high. Turning the resonance down to 0 in both of the Filters will eliminate the remaining sound.

6) Press the **Compare** button, and then hit the **Trigger** button again. You'll hear the original, unedited sound return. Press **Compare** and **Trigger** a second time to return control of the OB-Mx.

7) Leave this instrument without changing it by pressing the **Instrument** button and then turning the **Data Adj** knob one click in either direction. When prompted as to whether or not you want to save your changes, answer "No" by pressing the **Escape** button.

Exercise 2: Tweaking Blind

This is a very quick exercise to show you how parameters can be changed by turning knobs without actually first pressing a Selection Button to bring them into the display.

1. Load Instrument 239, *Dance 2*. Press **Trigger** to turn it on.
2. Grab the **Noise** knob and twist it all the way off, then all the way on. You'll notice that the amount of noise in the sound changes, even though you didn't pressed a filter button to select a filter to affect. The **Noise** knob changes the Noise parameter for the filter whose Selection Button is lit, regardless of whether it is flashing or not. This is great for fast edits—there's a lot that you can just reach right out and grab.

KEY POINT: Selection Buttons that are lit but not flashing determine which object is affected by edits (turning knobs and pressing buttons) in their modules.

Exercise 3: Saving and Naming Your Work

This is another real quickie. Now that you're getting used to making changes to Single Instruments, you might create something that you want to keep in the course of experimenting. Where and how should you save it, to be sure of not overwriting any factory patches?

The OB-Mx comes with 256 Single Instrument and 256 Multiple Instrument memory locations. Instrument locations 129-256 are factory patches stored in ROM, which cannot be overwritten.. The first 128 are copies of the ROM instruments, and can be overwritten without compunction.

Here's a quick run-through how to save your changes to a Single Instrument:

1. Load a Single Instrument. Grab a knob and turn it to make some change.
2. Press the **Store** button. The display will prompt you for a location number, asking where to save the modified Instrument. At this point, pressing **Escape** or any Selection Button will leave the display without completing the **Store** operation. Pressing **Enter** will store the changes to the Instrument location shown in the display. Don't do either of these yet.
3. Turn the **Data Adj** knob until Instrument 128, *InitialConditions*, appears in the display.
4. Press **Enter**. A message will appear confirming the save.
5. Press **Program**. Notice that the Instrument that is loaded is the one you started with, not the one that you saved to. Turn the **Data Adj** dial to call up the newly-saved instrument.

Now let's change the name:

1. Press the **Cursor >** button to move the cursor underneath the Instrument name.
2. Press the **Enter** button.
3. Turn the **Data Adj** knob to change the first letter of the name.

KEY POINT: The alphabet is stored in the order of: SPACE, capital letters, lower-case letters, numbers and special symbols. Giving the **Data Adj** knob a healthy counter-clockwise twist (towards the "off" position, if there were one) is an easy way to make a character blank.

4. Use the **Cursor** keys to move to different characters. We suggest that you change this particular name to "Test" or "Garbage" so that you won't have any compunctions writing over it later.
5. When you're done changing the name, press the **Escape** key to exit the name changing function.
6. Press **Store** and then **Enter** to save the new name.

Exercise 4: Backing Up And Restoring Via MIDI

If you have a MIDI sequencer, you can use it to record a MIDI System Exclusive dump of all or part of the data in your **OB-Mx**, as follows:

1. Make sure that a MIDI cable is connected from the **OB-Mx's** MIDI Out port to the sequencer's MIDI In.
2. Press the **MIDI** button in the **CONTROL** module. It doesn't matter whether you're in Single or Multi mode.
3. Turn the **Data Adj** knob until the Send parameter reads "Dump All."
4. Set your sequencer ready to record. Make sure that its "record filter," if it has one, is set to accept System Exclusive (Sys Ex) data. Some sequencers, like *Cakewalk*, have an independent System Exclusive area, while others record the data into a track.
5. Press the **Enter** button to start the dump.
6. Save the data to a disk file.

Restoring the data in your OB-Mx is easy:

1. Press the MIDI button.
2. Make sure that the SysEx parameter is turned On.
3. Send the dump from your sequencer to the **OB-Mx**. You'll see a series of messages indicating that the dump was successful.

   See the Reference section for details on dumping and restoring smaller chunks of data.

▶ The Back Panel

The back panel of the **OB-Mx** contains MIDI In, Out and Thru ports, the Main Outputs and individual outputs/insert points for each voice *(see Figure 2.3)*.

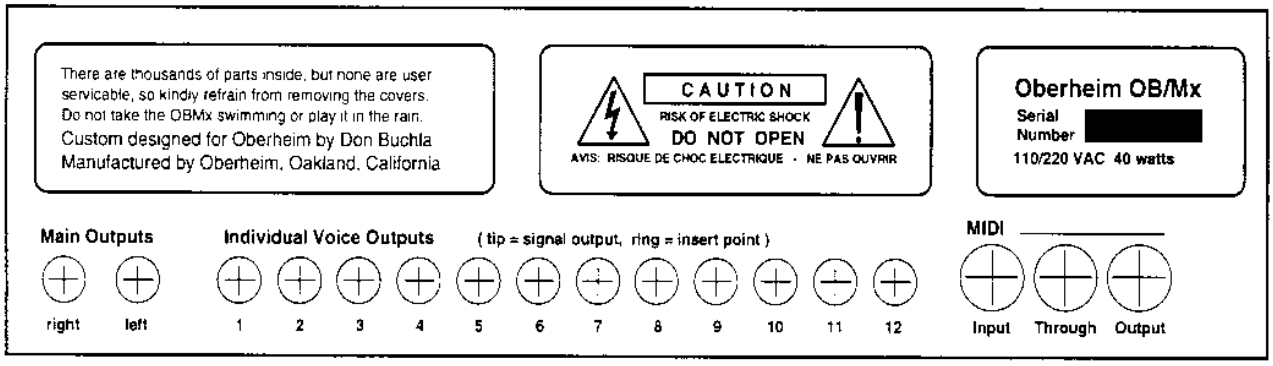


Figure 2.3
OB-Mx Back Panel.

Main Outputs

Connect these with 1/4" phone cords to your mixer, preamplifier or amplifier. If you will not be using the OB-Mx in stereo, you may use either one of the Main Outputs and still hear all the voices

Individual Outputs/Insert Points

Each voice has its own output. You can use each in one of the following ways:

- *Use it as an output.* Plug in a 1/4" cord and bring the voice's output into a mixer channel, preamplifier or amplifier. The voice output will not be mixed with other voices and will not be sent out the Main Outputs.
- *Use it as an insert point.* Plug in a TRS cord (Tip/Ring/Sleeve) and connect it to a signal processor. The voice's VCA output will be sent to the processor (*Tip*), and the processor's output will be returned to the OB-Mx (*Ring*)and sent to the Mixer module's Pan processor.
- *Ignore it.* The voice's output will be mixed with the other voices and sent out the Main Outputs.

MIDI Connections

There are 3 MIDI ports on the back panel. The *MIDI In* port is used to receive incoming MIDI messages from a controller, computer or sequencer. The *MIDI Out* port is used to send System Exclusive dumps from the **OB-Mx** to a sequencer or computer. The *MIDI Thru* port echoes all MIDI messages that come in through the MIDI In port for use by other MIDI devices.

CHAPTER 3

Explorations

In this section, we'll explore the most common and useful OB-Mx operations and concepts with short hands-on examples. The different sections in this chapter are meant to be independent of each other, so you can take them in any order that appeals to you.

3.1 Single Instrument Globals

When you press **Single** and **Program**, the name and number of an instrument appear in the top line of the display. You may have noticed that the bottom line of the display does not change from Instrument to Instrument. That's because those parameters are global, that is they apply to all Single Instruments. You can change them by using the **Cursor** < and > keys to position the cursor under then, and then turning the **Data Adj** knob to change them.

Exploration 1: The Chorus Algorithm

The **Al** setting in the Instrument display refers to the voice allocation algorithm. This determines how voices are allocated to incoming MIDI notes and, in particular, how notes are stolen if the number of simultaneously-held notes exceed the number of voices installed. In general, this is pretty technical and you should refer to the *Voice Allocation* chapter in the *Reference* section for more information. However, there is at least one value that is easy to understand—*Chorus.*

1. Load a Single Instrument that you like. Set the AI parameter to *Rotate* if it's not already there. Play and listen to the Instrument, and notice that as you play, a single button in the **VOICE STATUS** module lights for each note.
2. Change the AI parameter to "Chorus." Play some more, especially single-note lines. Notice that now *all* the lights (up to the number of voices that have been installed) in the **VOICE STATUS** module go on each time you play a note. The notes should also sound thicker (the perceived prominence of the effect varies from Instrument to Instrument). The Chorus voice allocation algorithm plays every oscillator at once for each note, all slightly detuned from one another. That's what adds the beef.
3. Notice that you can only play single lines, not chords. Every new note needs all the oscillators to sound, so it shuts off the previous note and steals its oscillators. It's a cold, cruel world.
4. If you wish, change the AI back to Rotate for more conventional musical uses.

For more information about Instrument Globals, see the list of global parameters on pages 39-40.

3.2 Using Multiple Instruments

▶ About Multiple Instruments

A *Multiple Instrument* (also called a *MultiProgram* in the display, or simply a *Multi* convenience) is a way of grouping together a number of Single Instruments into a performance preset. You can use it for setting up keyboard splits and layers, for calling up an entire ensemble complete with levels and pans with a single MIDI Program Change command or for simply determining which channels will respond to MIDI inputs.

When you press the **Multiple** button in the **INSTRUMENTS** module, the most-recently-selected MultiProgram's name appear in the display, along with the name of the Single Instrument in the MultiProgram's Part 1.

Each Multi can have up to 6 *Parts*, although there are situations when you may not have enough voices installed to use all 6 (see Chapter 4 for more information).

You can see the Single Instruments assigned to each Part by pressing the lit buttons in the **VOICE STATUS** module. You'll see the instrument names change in the display as you do so.

To see exactly what goes into each Part, press the **Parts** button. Each part is set to respond only to a specific keyboard range (Zon—the default is the full range of 0 to 127) and to respond on one MIDI channel. All the parameters are explained in detail in the *Control Module* chapter in the *Reference* section.

Exploration 2: Creating A Layered Split

Let's set a somewhat arbitrary task for ourselves: to create a Multi that plays *Instrument 129, Pie & Oh* on the lower half of the keyboard and *Instrument 157, Soft Solo* on the upper half.

Furthermore, let's transpose the left half up an octave—after all, it goes pretty low—and layer the entire keyboard with *Instrument 182, Padding Phase* (see Figure 3.1). Also, let's pan the Pie & Oh mostly to the left and the Soft Solo mostly to the right.

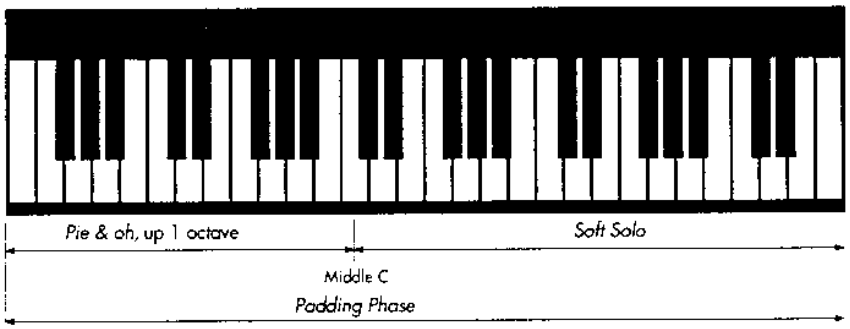


FIGURE 3.1
A layered split.

Here's how it's done:

1. Press the **Multiple** and **Program** buttons to enter Multi mode.
2. We'll need 3 Parts for our layered split, one for each Single Instrument. Each button that is lit in the **VOICE STATUS** module represents an active Part. Notice which Parts you'll need to activate or deactivate to wind up with exactly Parts 1, 2 and 3 active.

3. Follow the directions in the following KEY POINT to active and deactivate Parts until only 1, 2 and 3 are activated:

KEY POINT: To deactivate an active Part:
A) Move the cursor into the Part field.
B) Press the numbered button in the **VOICE STATUS** module to select the Part to disable
C) Press the **Escape** button to disable the Part

To activate a deactivated Part:
A) Move the cursor into the Parts field.
B) Press the numbered button in the **VOICE STATUS** module to select the Part to enable
C) Press the **Enter** button to enable the Part

4. Press 1 to examine Part 1.
5. Using the **Cursor** buttons and **Data Adj** knob, choose Instrument 129 for Part 1.
6. Press 2 to examine Part 2.
7. Using the **Cursor** buttons and **Data Adj** knob, choose Instrument 157 for Part 2.
8. Press 3 to examine Part 3.
9. Using the **Cursor** buttons and **Data Adj** knob, choose Instrument 182 for Part 3.

Now the Single Instruments are layered. Let's customize Part 1.

1. Press 1 to examine Part 1.
2. Press the Parts button to see detailed information about Part 1.
3. Move the Cursor under the 2nd number in the keyboard range (Zon), which should be set at 127.
4. Press the **Enter** button to make the number receptive to MIDI keyboard input. This is to protect you from going over to your keyboard, and unwittingly changing the keyboard range by playing a song. Stranger things have happened.
5. Play the B just below middle C on a connected MIDI keyboard, if you have one, or turn the **Data Adj** knob until the number 59 appears in the display.

6. Press the **Escape** button to turn your keyboard back into a performance instrument instead of a data entry device.
7. Using the Cursor buttons and the **Data Adj** knob, set the MIDI Channel (**Ch:**) to 1.
8. Set the Pan to -7, which is 70% over to the left.
9. Check that **Attn** is 0, Al is Rotate and TT is 10. Check the *Reference* section for more on these parameters.
10. Change Xpose to +12 (semitones) to transpose this Part up an octave.

Now work on Part 2:

1. Press **Multi part number** button 2 to examine Part 2.
2. Move the Cursor under the 1st number in the keyboard range (Zon), which should be set at 0.
3. Press the **Enter** button to make the number receptive to MIDI keyboard input.
4. Play middle C on a connected MIDI keyboard, if you have one, or turn the **Data Adj** knob until the number 60 appears in the display.
5. Press the **Escape** button.
6. Using the Cursor buttons and the **Data Adj** knob, set the MIDI Channel (**Ch:**) to 1.
7. Set the Pan to +7, which is 70% over to the right.
8. Check that **Attn** is 0, Al is Rotate, Xpose is 0 and TT is 10.

Finally, Part 3:

1. Press **Multi part number** button 3 to examine Part 3.
2. Using the Cursor buttons and the **Data Adj** knob, set the MIDI Channel (**Ch:**) to 1.
3. Set the Pan to +o, which is the center of the stereo field.
4. Check that Pan is 0, Attn is 0, Al is Rotate, Xpose is 0 and TT is 10.

Now you're ready to play. Does the MultiProgram behave as you expected?

Exploration 3: Creating a Balanced Performance

Here's where we explore the ins and outs of creating and maintaining a good balance among the three parts in our Multi. First, let's try a little experiment:

1. Examine the **Vol** parameter of any active Part. Notice the value.
2. Press the part number buttons to examine the volumes of the other two parts. You'll notice that they're all the same.
3. Go over to your MIDI controller and change the volume. If the controller sends out MIDI Volume messages—Continuous Controller 7; some do and some don't—then the **Vol** value in the display will change.
4. Look at the volumes for all 3 Parts—they've all changed together. And why shouldn't they? They're all on the same MIDI channel, receiving the same MIDI Volume message.

   This behavior, while perfectly logical, raises an interesting question. Suppose we could painstakingly balance the different Single Instruments in a Multi to our taste? Could we raise or lower the volume of the whole mix without changing the balance? If the MIDI Volume message were our only tool, the answer would be "No." At the first sign of a Volume 90 message in channel 1, all 3 instruments would assume volume 90, wrecking whatever balance you'd achieved. That's why we put in a **Attn** parameter.
5. Adjust the **Attn** parameters in all 3 Parts until you like the balance among them.
6. Turn the Volume knob on your controller, or change it with the **Data Adj** knob. Notice how the balance among the parts is preserved at different overall volumes.

KEY POINT: Use the **Attn** parameter to adjust the balance among the Single Instruments in a MultiProgram, and use the Vol parameter to change the overall volume while retaining the balance.

SAVE THE MULTIS!

It would be a good idea to save your Multi definition now. Press **Store**, locate a Multi that you can write over, and press **Enter**.

Multiple Instruments and MIDI Program Change Messages

You can select different Multis numbered from 1 to 128 remotely by using MIDI Program Change messages on the Base Channel (press the **MIDI** button to see this). You can change the Single Instrument assigned to a Part by sending a Program Change on the channel assigned to that Part. There are several points to be aware of with regard to this process:

- If a Part within a Multi is assigned to the Base Channel, then a Program Change on the Base Channel will cause a new Multi to be loaded rather than just a Single Instrument change within a Part. To avoid any possibility of ambiguity, it's a good idea to avoid assigning Parts to the Base Channel. Alternatively, you can alter this behavior by pressing the **MIDI** button and then setting the MPC parameter to **Off**. This will prevent Program Changes from affecting Multis, while still allowing them to change Single Instruments within a Multi.
- If several Parts within a Multi are assigned to the same channel, then a Program Change on that channel will change all Parts to the same Single Instrument.

3.3 Modulation & The Matrix

Modulation means change. In the **OB-Mx**, it means affecting the parameters of an oscillator, filter, or other Instrument component with the output of another component (or even the same component, creating a feedback loop). The component being changed is called the *modulation target* or *destination*, and the component doing the changing is called the *modulation source*. Here are a few of the many uses of modulation:

- The most fundamental use of modulation in synthesizers occurs with every note that you play. In most patches, every keyboard or controller note modulates the pitch of the VCOs generating the sound. This is what allows you to play most forms of music; without it, each note would result in a sound with the same pitch, or one determined by other modulators not necessarily related to the note that you played. In this example, the MIDI note number is the source, and the Pitch parameter of a VCO is the target.
- Refer back to our discussion of the classic model of analog synthesis and ***Figure 3.1***. An envelope generator (source) is responsible for controlling the output level of the VCA (target). The envelope modulates the output level parameter of the VCA constantly (except during the Sustain phase) during each note. In the **OB-Mx**, the VCA is

incorporated into the **FINAL MIX** module. Its output is controlled by Envelope Generator 4, by the **VCA Offset** knob and by any additional sources that you map to it using the Matrix.

- An LFO is often used to create vibrato in sustained notes. The LFO supplies a voltage that varies periodically (usually). This source can be used to modulate oscillator pitches, filter frequencies and note volumes (at various points in the signal path) to simulate the vibrato effect that naturally occurs in acoustic instruments when a wind or string player sustains a note. Note that a single source can have multiple targets.
- Every MIDI Continuous Controller is a potential modulation source. You can use a MIDI footpedal, for instance, to modulate the center frequency of the OB filter to create a wah-wah effect; you could simultaneously use another MIDI footpedal or a slider to change the character of the wah effect by modulating the filter's resonance.
- The time scale of a source can vary from the near-instantaneous to the very gradual. The slowest rate on an **OB-Mx** LFO is 0.010 Hz, or 1/100 cycles per second. If you press the LFO 1 selection button and then twisting the Rate knob all the way counter-clockwise, the display will tell you that the Period of this LFO is 100 seconds. While that's not exactly glacial, you could easily achieve subtle, continuous timbral variations over the course of a piece by using this LFO to modulate the pulse width of an oscillator or the resonance of a filter. In this case, you'd want to use a triangle-shaped LFO to avoid a sudden jump at the end of each cycle.
- If you use the output of one oscillator to modulate the pitch of another, you're engaging in FM synthesis. This is very fast modulation; the output of each oscillator changes continuously (these are analog, not digital oscillators, so you can take the word "continuously" literally), going through hundreds or thousands of complete cycles each second. In FM terms, the source oscillator is called "the modulator" and the target is called "the carrier." You don't actually listen to the modulator; you just hear its profound timbral effects on the carrier. We'll be doing an exploration with this technique shortly.

▶ OB-Mx Default Modulations

A number of default modulation paths have been set up in the **OB-Mx** which are active in every Single Instrument. These ease the most common tasks that musicians encounter when setting up sounds. In accordance with our philosophy of creating a synthesizer that is as close to completely configurable as possible, all of these routings can be bypassed in one way or another and alternate modulations put in their place.

TABLE 3.1
Default OB-Mx modulations. The Mod Amount column indicates how to scale, invert or shut off a given modulation.

SOURCE	TARGET	MOD AMOUNT
MIDI Pitch	VCO pitch VCF frequencies EG times	Track knob VCF Kbrd Track buttons EG Kbrd Track button
MIDI Velocity	EG levels	VSen knob
EG 1	VCO pitch MM Filter Frequency	Pitch Env 1 knob Env 1, 2 knob
EG 2	OB Filter Frequency VCO pulse width	Env 1, 2 knob P Width knob
EG 4	VCA output level	Always 100%
LFO 1	VCO pitch MM Filter Frequency	Pitch LFO 1 knob LFO 1, 2 knob
LFO 2	MM Filter Frequency VCO pulse width	LFO 1, 2 knob P Width knob
VCO 1	VCO 2	FM knob
VCO 2	VCO1	FM Knob

The default modulations are shown in **Table 3.1**. Each of the modulations in the table is explained in detail in the *Reference* section, but we'll examine a few of the key ones with hands-on explorations here.

Exploration 4: EG 4 and The VCA

KEY POINT: Envelope Generator 4 controls the VCA.

Envelope Generator 4 is hard-wired to the VCA; it's the one source that always affects its target fully, without any modulation amount parameter to dilute its effect. It's critical for playing notes.

1. Load Single Instrument 128, *InitialConditions*. Press the **Trigger** button or play a note on your keyboard to hear the sound of the patch.

2. Press the **Envelope Generator 4 (vca)** button. Turn the **Attack** button until the value shown in the display is about .3 seconds. Press the **Trigger** button again and listen—the initial impact of the note is softened considerably. Change the **Attack** value to several seconds, and then press and hold the **Trigger** button until you can hear the note fading in.

   Notice that, with the **Attack** value set to several seconds, a quick press and release of the **Trigger** button doesn't produce any sound. This is because the envelope has been interrupted before it reached an audible level. When you release a note, the envelope generator proceeds directly from its current level to the Release stage, which fades out the note gradually from its current level to 0, using the time period specified in the **Rels** parameter in the display—in this case, a little over half a second.

3. With the **Attack** value still set to several seconds, press the **Freerun** button. Now press the **Trigger** button again, and release it immediately without holding it down. You'll hear a note fade in and then out. When the **Freerun** parameter is set, the entire envelope completes regardless of whether or not the triggering key or button is held down. This is in contrast to the non-Freerun envelope, which proceeds immediately to the Release stage when a Note Off comes in, without necessarily completing the Attack or Decay. **Freerun** especially handy for use with percussion controllers, which typically generate Note Ons only.

4. Turn **Freerun** off, and reset the attack to a short time.

Now we're ready to explore velocity sensitivity:

5. If you have a velocity-sensitive keyboard (most are), then play soft and hard notes to hear the dynamic range of the Instrument. You should hear some, but not much, difference between the softest and loudest sounds.

6. Press the **Alt** button. This switches the function of the dials at the top of the ENVELOPE GENERATORS module to those indicated by the names printed under them, enclosed in curly brackets. Turn the knob marked Velo Sens all the way clockwise, until the display shows a value of 25dB. This number refers to the difference in maximum output level between a note with velocity 1 and one with velocity 127.

7. Play your keyboard again. The dynamic range should be greatly increased.

Exploration 5: Fixed-Pitch Sounds and Quartertone Keyboards

Sometimes it's important for a sound *not* to track the keyboard's pitch. The most commonly-used example of this would be for a keyboard split to play a percussive sound in an entire region of the keyboard. This frees you from

having to be very careful about which note you play. We'll learn how to do this, and, along the way, how to get at some of the notes in between the black and white keys.

1. Load Single Instrument 256, *InitialConditions*. Turn off VCO 2's output by first pressing the **VCO 2** selection button and then pressing the lit triangle wave button so that its LED goes off..
2. Play your keyboard or other controller. Notice that the pitch of the Instrument follows the notes that you play.
3. Press **VCO 1**. Notice that the track parameter (**Trck** in the display), is set to +1.00.
4. Use the **Track** knob to set the Trck value to 0. Play the controller now. You'll notice that the pitch of the notes you hear no longer depends on which key you press.

   The **Track** knob can be used for some other neat tricks, too:
5. Turn the knob all the way counter-clockwise, until the value is -1.00. Now play your keyboard—it's been turned upside down! High keys sound low notes, and vice-versa.
6. Set the **Trck** value to +0.50. Play a middle C, a C one octave above it and a C one octave above that. It now takes two octaves on the keyboard to make a one-octave change in the pitch of the Instrument. The pitch difference between adjacent keys is now a quartertone, not a semitone.

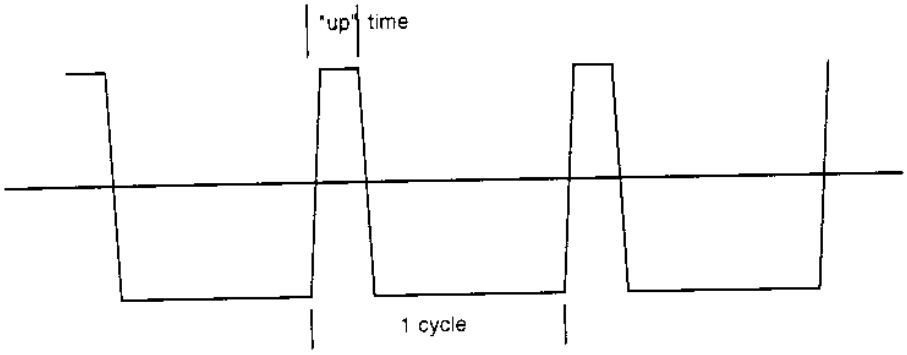
Exploration 6: Adding Vibrato

1. Load Single Instrument 256, *InitialConditions*.
2. For simplicity's sake, turn off VCO 2's output. Press **Trigger** and listen to the sound.
3. Bring VCO 1 into the display.
4. Turn the **Modulation/Pitch LFO1** knob until the value for **LF1>Pch** is + .20. Press **Trigger** and listen to the sound. If the effect is too subtle for your taste, turn the knob higher.
5. Experiment with different values for the **LFO 1 Rate**. For a flute-like sound, try 5.0 Hz.
6. Add about 1.5 seconds of initial delay. This is analogous to the way many flautists perform; their notes are often stable for several seconds before they add a vibrato effect to support sustained notes.

Exploration 7: Pulse Width Modulation

One of the wave types that the VCOs can generate is called a *Pulse Wave*. This wave consists of alternating high and low outputs, connected by very brief transitions, as shown in Figure 3.2.

FIGURE 3.2
A Pulse Wave



The pulse wave looks very choppy, but you don't hear a chopped-up sound; instead you hear a continuous tone. The pitch of the tone is determined by how many up-and-down cycles complete in a second.

The amount of time the pulse wave spends in the high position, express as a percentage of the total time for a complete cycle, is known as the *duty cycle* of the pulse wave. As the duty cycle changes, the pitch remains the same but the timbre of the note changes.

KEY POINT: Pulse Width="Duty Cycle"= "up" time / period = "up" time * frequency. The frequency determines the pitch; the pulse width determines the timbre.

1. Load Single Instrument 256, *InitialConditions*.
2. Turn off VCO 2.
3. Select VCO 1. Shut off the triangle wave output and turn on the pulse wave output. Listen to the sound.
4. Turn the **P Width** knob until the PW parameter in the display is in the 60s. Listen to the sound of the Instrument again.
5. While holding down the **Trigger** button, twist the **P Width** knob through a range of values. This simulates what Pulse Width Modulation is all about. The only thing left to do is to make an LFO do that twisting instead of us.

6. Set the PW parameter back to 0.
7. By twisting the **P Width Env/LFO** knob (below the pulse wave button) to about the 3:00 position, set the LF2>Pw parameter to about .70. Press and hold the **Trigger** button and listen to the sound.
8. Experiment with changing the Rate of LFO 2 to see how it affects the sound of the oscillator.

The Matrix

No, it's not a weird cyberspace construct or a tear in the fabric of the space-time continuum; it's just a software patch bay you can use to connect modulation sources to destinations. It's the nexus where interacting connections between simple components can transform the **OB-Mx** from a synthesizer into an audio computer. It's the place where you can get really expressive, really weird or both!

Exploration 8: Pulsing Noise

1. Load Single Instrument 256, *InitialConditions*.
2. Press the Matrix button. The parameters for Patch 1 will appear in the display, and the 1 button in the **VOICE STATUS** module will flash. A Patch is a connection between a modulation source and a modulation destination, along with a scaling multiplier to set the amount of modulation. You can program up to 12 Patches for each Single Instrument.
3. Turning the **Data Adj** knob takes you through all the possible modulation sources. Select **LFO 3**. We know from **Table 3.1** that this is not used for any default purpose, so we'll have complete freedom to play with it.
4. Set the Multiplier to the maximum value, +1.00.
5. Set the **DESTINATION** module to MM VCF. The choices in the Param: field change as you change **DESTINATION** module.
6. Set the Param to Noise In.
7. Press and hold the **Trigger** button. You should hear a pulsing noise along with the basic tone.
8. Press the **LFO 3** button and experiment with the effects of turning the **Rate** knob and choosing different waveforms.

OB-Mx User's Guide

Section 2

Reference

Oberheim®

CHAPTER 4

Voice Allocation

Your OB-Mx synthesizer has between 2 and 12 voices, each containing 2 VCOs, a noise source and the other components that make up a note in a Single Instrument. Each note played uses at least one voice, but you may decide to stack more than one for a thicker sound.

The Al (voice allocation algorithm) parameter allows you to stack sounds in various interesting ways, and also lets you control how "note stealing" is to occur. Note stealing happens when you try to play more notes than you have voices for; for example, if you play a keyboard with your entire forearm or send a thick multi-part arrangement to the synthesizer from a sequencer (or, if you have just a few voices installed, in less extreme situations).

You can watch the algorithms in action as you play; the buttons in the **VOICE STATUS** module light up to indicate each voice as it is played (unless you're in Matrix mode, with the **Matrix** button in the **CONTROL** module lit).

4.1 Single Instrument Voice Allocation Algorithms

In Single Instrument mode, the Al parameter can take the following values:

Rotate

Each new note is assigned to the next free voice. If several voices are free, the new note uses the voice that's been released for the longest time. If no voices are available, the new note steals a voice from the note that has been sounding for the longest time.

Repeat

This is an extremely useful algorithm for a number of reasons. When using the Repeat algorithm, repetitions of the same note will use the same voice, which won't be stolen by a new note until you've run out of voices.

To understand the reason for Repeat mode, think of a keyboard player playing an Instrument with a 5 or 6 second release time (or try it yourself). A chord played in the left hand and then released would sustain noticeably. In the Rotate algorithm, a 2-note trill played in the right hand during the sustain, with the same long release time, would quickly steal voices from the sustaining chord, especially in systems with less than the maximum number of voices installed. Using Repeat, however, causes only two voices to be used for the trill, and leaves the sustained chord intact.

Another reason for using Repeat mode is that the OB-Mx is an analog instrument. Therefore, the states of the various oscillators and analog modulators in different voices is likely to be slightly different at any given time. When playing notes that differ in pitch, these sonic differences will generally be of little concern; in fact, they add life to the overall sound. However, there may be some situations where you want to play the same note repeatedly. In these cases, you'll get a more consistent sound by letting the Repeat algorithm re-use the same voice for the repeated notes. You'll also avoid the unison chords that result from repeating notes with long decay times in the Rotate mode.

Combo

A curious combination of the above two algorithms. The first note sounded is assigned with the Repeat algorithm; successive notes are assigned with the Rotate algorithm until all notes are released. A succession of single notes (releases precede closures) will be assigned with the Repeat algorithm, while overlapping notes (legato) will activate the Rotate mode. Experiment with a slowly decaying sound such as *Li'l Bells* (Instrument #219) to best understand the potential of the Combo algorithm.

Double, Triple, Quad, Hex

Each new note gets 2, 3, 4 or 6 voices, respectively (if you have enough voices). This creates a louder sound than a single voice, but the most interesting thickening effects happen if you subtly modulate the pitch of the VCOs with an LFO or envelope. If, for instance, you set the LFO 1 waveform to be Random, then each Note On will trigger a new, different value in each of the voices and will modulate the oscillator pitches to different values, which should thicken the sound considerably.

Unison

Each new note gets all the voices in the instrument.

Chrs/3, Chrs/4, Chrs/6, Chorus

A fast and easy way to get thicker sounds. Each new note gets 3, 4, 6 or all of the voices, respectively. All the voices for each note are detuned slightly from one another to create a chorus effect. This is similar to the previous voice-stacking algorithms, but the detuning is static rather than controlled by a modulation source.

4.2 Multiple Instrument Voice Allocation Algorithms

In the **Multi** mode, each Part has its own voice allocation algorithm. By pressing the **Voices** button, you can determine which voices will be allocated to each Part. Setting the **Asgn** parameter to Auto will allocate the voices evenly among the Parts; setting it to User will allow you to customize

the voice allocation. For further information, see the **CONTROL MODULE** chapter.

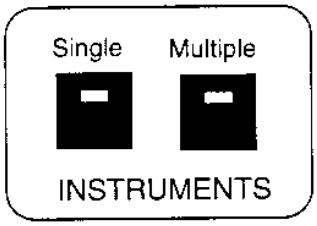
If you have one voice allocated per part, all the voice allocation algorithms will produce the same result—incoming notes will take the only voice in the part, stealing it from a sounding note if necessary. It's when there are several notes in a Part that the algorithm becomes relevant.

The Al parameter for a Multi Part is found by pressing the **Parts** button in the **CONTROL MODULE** and the number of the Part you wish to edit in the **VOICE STATUS** module. Most of the choices are identical to those in the Single Instrument definitions listed above, and supercede them. The Hex, Chrs/6, Unison and Chorus are absent, but there is a new choice:

Legato: Each note triggers all the voices assigned to the Part. When a new Note On comes in before the prior note has ended, the pitch of the currently-sounding note is changed, but the envelope is not retriggered. When that Note On is released, the pitch will change to that of the last key played that is still held down.

CHAPTER 5

Instruments Module



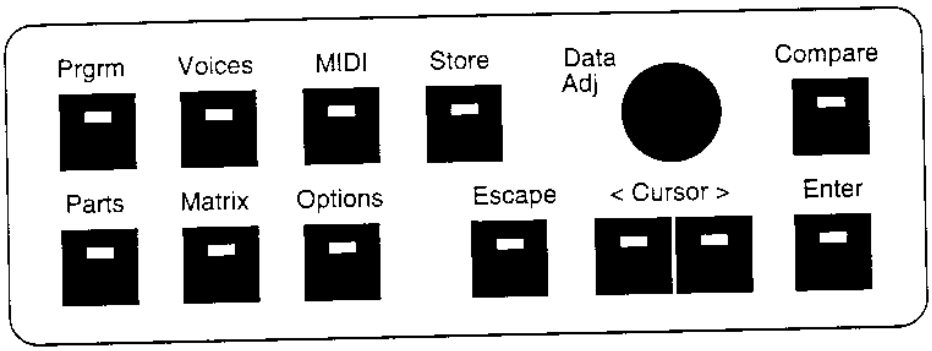
The buttons in this module select either Single Instrument or Multi mode. In Single Instrument mode, all incoming notes are played on the currently-loaded Instrument, while in Multi mode incoming notes are assigned to different Parts depending upon various parameters (explained in the **CONTROL MODULE** section following this one).

Most people will use Single Instrument mode for creating and testing individual sounds, and use Multis for creating performance presets, splits and layers and for addressing multiple sounds on different MIDI channels with sequencers.

When you press one of the buttons in this module to change modes, you are prompted first to save any Instruments—be they Single or Multi—that you have changed. You can save them into any RAM location. See the description of the **Store** button in the **CONTROL** module for more details.

CHAPTER 6

Control Module



This module controls most of the functions in the OB-Mx that are not directly related to sound creation, including storage, global parameter settings and Part assignments in Multis.

CHANGING VALUES IN THE DISPLAY

As you've no doubt already observed, many parameter values in the display are changed by first using the **Cursor** < and > buttons to position the cursor underneath the parameter and then turning the **Data Adj** knob. If you find that a parameter in the display is not accessible (i.e., you can't position the cursor under it), that means that a parameter knob in the active module (indicated by a flashing Selection button) is the only way to change that value.

PROGRAM BUTTON

This button brings the Instrument Name and some key parameters into the display. Its main functions are:

- Loading Instruments, either Single or Multiple.
- Changing Instrument names.
- Changing the Single Instruments assigned to the Parts in a Multi.

▶ *Program Parameter Descriptions, Single Instrument Mode*

Instrument : 127 *<24 Character Instrument Name>*
Pan: +0 Vol: 127 Al:Rotate Mode:Poly TT:1

Instrument

A number in the range of 1 to 256 which loads a Single Instrument. When you attempt to load a new Single Instrument, you'll be asked if you want to save any changes you've made to the current Single Instrument. If you answer "Yes" by pressing **Enter**, then you'll be able to save your Instrument to any user location. If you answer "No" by pressing **Escape**, then changes will be discarded.

Instrument Name

The name of the currently loaded Single Instrument, up to 24 characters long. To change the name, proceed as follows:

1. Position the cursor under the instrument name.
2. Press the **Enter** button.
3. Use the **Data Adj** knob to dial in a new character.
4. Use the Cursor < and > buttons to position the cursor under a new character.
5. Repeat steps 3 and 4 until the name is what you want it to be.
6. Press the **Escape** button to exit name changing mode.
7. Store the Instrument if you want to preserve your changes.

The next 5 parameters, all on the second line of the display, are global for all Single Instruments. This means that they are not saved along with the Single Instrument definition when you use the **Store** button, and they are not changed when you load a new Single Instrument. For storable versions of these settings, use a Multi.

Pan

Pans the Instrument in the stereo main outputs. This control is additive with the **Pan** parameter in the Instrument definition. Its range is -10 (full left) to +10 (full right), with 0 representing the center.

Vol

Responds to MIDI Volume (Continuous Controller 7) messages and lowers the output of the Instrument. This control is independent of the VOLUME knob in the upper-right corner of the **OB Mx**, which simply attenuates the overall output of the synthesizer.

Al

The global voice assignment algorithm for Single Instruments. It can take on the values **Rotate, Repeat, Combo, Double, Triple, Chrs/3, Quad, Chrs/4, Hex, Chrs/6, Unison** or **Chorus.** See the chapter on Voice Allocation Algorithms for a complete description.

Mode

Omni or Poly. When a Single Instrument is in Omni mode, it will respond to MIDI events coming in on any channel. In Poly mode, it will respond only to events on the Base Channel, which you can set by pressing the MIDI button.

TT

Determines which Tuning Table is in effect. The Tuning Table will only affect VCOs that have their Track parameter set to TT. See the description of the *EditTune* parameter of the **Options** button for more details.

▶ Program Parameter Descriptions, Multiple Instrument Mode

MultiProgram: 127 *<24 Character Program Name>*
Part: 1 Inst: 58 *<24 Character Instrument Name>*

MultiProgram

A number in the range of 1 to 256 which loads a Multi. When you attempt to load a new Multi, you'll be asked if you want to save any changes you've made either to the Multi or to any Single Instruments assigned to Parts in the Multiple. If you answer "Yes" by pressing **Enter**, then the Multi is saved first and you are presented with additional prompts, one by one, to save any changed Single Instruments. If you answer "No" by pressing **Escape**, then changes to both the Multi and to any Single Instruments will be discarded.

<24 Character Program Name>

The name of the currently loaded Multi, up to 24 characters long. To change the name, proceed as follows:

1. Position the cursor under the instrument name.
2. Press the **Enter** button.

3. Use the **Data Adj** knob to dial in a new character.
4. Use the **Cursor** < and > buttons to position the cursor under a new character
5. Repeat steps 3 and 4 until the name is what you want it to be.
6. Press the **Escape** button to exit name changing mode.
7. Store the Instrument if you want to preserve your changes.

PART/Part

Use this for changing the Part that is displayed, and for activating and deactivating Parts.

The Part number can be 1-6, and you change it either by using the **Cursor** < and > buttons and the **Data Adj** knob or by simply pressing a Part Number button in the **VOICE STATUS** module.

If a Part is active, the word "PART" is displayed in all-capital letters. Also, the **Escape** button will flash when the cursor is positioned in the Part field of an active Part. Pressing **Escape** will deactivate the Part.

If a Part is deactivated, the word "Part" is displayed with only the first letter capitalized. Also, the **Enter** button will flash when the cursor is positioned in the Part field of a deactivated Part. Pressing **Enter** will activate the Part.

See Explorations 2 and 3 for information about using Parts.

Inst

The Single Instrument number for the currently-displayed Part. This Instrument is the one that will be affected by any parameter changes in the other modules. Bring a different instrument into the Part by changing this number. If the currently-loaded Single Instrument has been changed, you'll be asked if you want to save changes or not.

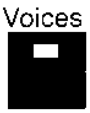
<24 Character Instrument Name>

This is the name of the Single Instrument chosen by the Inst parameter above. For convenience, it can be edited directly from this display without leaving Multi mode. Refer to the directions for changing Multi names for the exact steps.

VOICES BUTTON

This button is used to enable and disable voices in Single Instrument mode. Most of the time you'll probably leave all the voices on. Its more significant function is in Multi mode, where it is also used to assign voices to Parts.

▶ *Single Instrument Mode*



Voice	Voice:	1	2	3	4	5	6	7	8	9	10	11	12
Enable	Status:	*	*	*	*	*	*	*	*	*	*	*	*

An asterisk under a Voice number indicates that the voice is turned on, a blank space indicates that it is off. Only voice numbers that are actually installed in your system will appear in the top row.

▶ *Multiple Instrument Mode*

Multi :127	Voice:	1	2	3	4	5	6	7	8	9	10	11	12
Asgn:User	Part:	1	1	2	2	2	6	6	5	5	5	5	5

This screen is where you specify how many voices are assigned to each Part. Each Voice in the top line is assigned to the Part below it.

Asgn

In the **Auto** assign mode, the available voices are automatically distributed evenly among the active parts (the ones that are lit in the **VOICE STATUS**

module) according to the needs of the Algorithms assigned to the various parts. In the **User** assign mode, you'll be able to move the cursor into the Part area and specify your own voice assignments

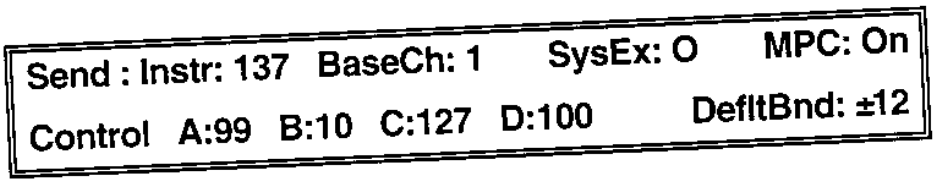
Part

Shows the Part number assigned to the voice number above it. In the example display above, Part 1 gets the first 2 voices, Part 2 gets voices 3, 4 and 5, etc. Parts are numbered 1 to 6, and you can also dial in a blank, which means that the voice will not sound in this Multi. Note that you can assign voices to Parts that are not currently active. The voices for those Parts won't sound until you activate the Parts (see the **VOICE STATUS** module chapter for more detail).

Pressing the **Voices** button a second time while in Multi mode brings up the display listed under Single Instrument mode above. It functions identically in Multi mode.

MIDI Button

Sends System Exclusive dumps and determines global MIDI settings for the OB-Mx.



Send

Sends a MIDI System Exclusive dump of all or selected parameters to a sequencer or another **OB Mx**. The **Enter** button flashes when the cursor is in this area; pressing it will start the dump. You can send a single Multi, Single Instrument or Tuning Table, a block of 64 Multis or 32 Single Instruments, all Tuning Tables, all global settings or all the data in the OB-Mx.

Later on, when a sequencer sends one of these dumps back to the OB-Mx, the synthesizer will automatically load the new data if the SysEx parameter (which refers to System Exclusive reception only, and doesn't have any effect on your ability to *send* a dump) is set to On.

Multi:nnn

A pair of parameters, the first indicating that an individual Multi will be sent, and the second indicating the number of the Multi. You can send any Multi in RAM (numbers 1-128) this way. *Note: ROM Multis (129-256)cannot be sent via MIDI. Copying to RAM locations will allow you to send these Multis via MIDI.*

Multi 1-64, Multi 65-128

Sends out a bank of 64 Multis.

Instr:nnn

A pair of parameters, the first indicating that an individual Single Instrument will be sent, and the second indicating the number of the Single Instrument. You can send any Single Instrument in RAM (numbers 1-128) this way. *Note: ROM Instruments (129-256)cannot be sent via MIDI. Copying to RAM locations will allow you to send these Intruments via MIDI.*

Ins 1-32, Ins 33-64, Ins 65-96, Ins 97-128

Sends out a bank of 32 Single Instruments.

TuneT:nn

A pair of parameters, the first indicating that an individual Tuning Table will be sent, and the second indicating the number of the Tuning Table. You can send any Tuning Table (numbers 1-19) this way.

AllTuneT

Sends all 19 Tuning Tables.

Globals

Sends the following:

- The current mode, whether Single or Multiple

- The Base Channel
- Whether SysEx reception is turned On or Off
- Whether the MPC parameter is On or Off (see page 46).
- The numbers of the MIDI Continuous Controllers that are currently assigned to modulation sources MIDI A, MIDI B, MIDI C and MIDI D.
- The pitch bend range
- The global tuning, expressed as the frequency of A, expressed in Hz (Default is 440)

If you're in Multi mode, the following parameters describe what the setup would be if you were to press the ***Single*** *button. If you're in Single Instrument mode, they simply describe the current setup:*

- The number of the currently-loaded Single Instrument
- The Single Instrument voice assignment algorithm
- The Single Instrument MIDI mode, whether Poly or Omni
- The Single Instrument Tuning Table

Dump All

Sends everything.

Base Channel

This can be set to any MIDI channel, from 1 to 16. It affects the following functions:

- Multi numbers are changed by MIDI Program Change messages coming in on the Base Channel, providing the **MPC** parameter is turned On. If the Base Channel is the same as a channel that a Part is assigned to, only the Multi will be affected by the Program Change. See *Using Multiple Instruments* in the *Explorations* chapter for a detailed explanation.
- The Base Channel will receive Global Continuous Controllers. This means that any MIDI Continuous Controller coming in on the Base Channel will be sent to all Parts, regardless of their channels. This way, a single MIDI Volume or Mod Wheel message can affect all sounds at once. The Controllers assigned to MIDI A, MIDI B, MIDI C and MIDI D will also be routed to all parts if they come in on the Base Channel.

SysEx

Turns reception of MIDI System Exclusive messages on or off. Turning this to Off can protect the RAM in the instrument from accidental overwriting by MIDI dumps embedded in sequences. Bear in mind, though, that you

can't overwrite the data in the **OB-Mx** with data from any other synthesizer model in any case; SysEx dumps from other instrument models will be ignored.

MPC

Stands for "Multiple Program Change." When this is On, MIDI Program Change messages on the Base Channel will load a new Multi. When this is Off, MIDI cannot change the current Multi. This parameter can be useful if you have Parts assigned to the Base Channel and want to change the Single Instruments assigned to them with MIDI messages.

Control A:, B:, C: and D:

This enables you to use 4 MIDI Continuous Controllers other than the most common ones to modulate parameters. If you want to use Continuous Controller 73, for instance, to modulate the cut-off frequency of the MM filter, you'd set Control A: to 73 here, then, in the **Matrix**, use **Control A** as the modulation **Source**, and set the **MM VCF**, parameter **Frequency** as the modulation **Target**.

DefaultBend

Determines the maximum number of semitones that a note can be bent with a MIDI Pitch Bend message. It can have a value from ±0 to ±12 (one octave). When the value is set to ±0, no bending will take place. When using values other than ±0, it's normally important that this value match the Bend Range set on your MIDI controller. Of course, you don't have to be normal. A guitar controller with a Bend Range of ±2 set to control a synthesizer with a Bend Range of ±12 will give you very weird, exaggerated bends. In Multi mode, default bend only affects those parts in which Def has been entered in the Bend (Bnd) field.

STORE BUTTON

Saves the currently-loaded Instrument to any RAM location, copy any Instrument to another location in RAM or swap any 2 RAM Instruments.

▶ *Saving Edited Instruments*

Whenever you take an action that would cause any changes that you've made to a Single or Multiple Instrument to be lost, or if you press the **Store** button after you've made any edits, you'll be prompted as to whether you want to save your changes. When you're in Single mode this is quite straightforward. In Multiple mode, the sequence can be a little more complex because you can choose to save some changes but not others; still, if you take the prompts literally, you won't go wrong.

If you've made changes to a Multi and/or one or more Single Instruments while in Multi mode, the **Escape** and **Store** buttons will blink when trying to load a new Multi, and the following message will be displayed:

Save Changes to MultiProgram/Instruments?
Escape:NO **Enter:YES**

You're being asked whether you want to save any changes at all here—Program refers to Multis and Instruments refers to Single Instruments.

- You don't have to take either choice—there's an implied "Cancel" function. If you press any Selection Button, like **Prgrm** or **MIDI**, for example, you'll be returned to a different display without loading anything new or losing any data.
- If you answer NO, you'll lose the data for both Multiple and Single Instruments.
- If you answer YES, you'll be given the option of saving each changed Instrument individually, beginning with the following screen:

Enter

Save Modified MultiProgram Number 4
to Multi:4 (name of MultiProgram)

At this point, the **Store** button will be lit and the **Escape** and **Enter** buttons will be flashing. Turning the **Data Adj** knob will let you choose a different memory location to store the Multi. Pressing **Enter** will store the Multi, and pressing **Escape** will skip storing the Multi and move on to the Single Instrument options, as follows:

Escape

Save Modified Instrument 12 (Part 1)
to Inst:12 (name of Instrument)

As before, you can use the **Data Adj** knob to change the target location and press **Enter** to save the Instrument, or you can press **Escape** to continue without saving.

You'll be prompted once for each Single Instrument that has been changed. If you Escape through all of the saving options, you'll be returned to the MultiProgram screen with the same Multi still loaded. The logic behind this is that if you really wanted to move to a different Multi without saving anything, you would have pressed the Escape button when you were first given the chance. Taking the safest path to protect your changes, the **OB-Mx** puts you back at the start of the process.

▶ Copying And Swapping Instruments

The **Store** button is also used to access copy and swap functions for both Single and Multis. If you press the **Store** button repeatedly when you have unedited Instruments in memory, you'll pass through the copy and swap screens. If you're in Multi mode, you'll first see the screens that apply to Multis, then the screens for Single Instruments. If you have any edited, unsaved Instruments in memory, then you'll first be given the opportunity to save them as described above—repeatedly pressing **Store** in this case will pass through those prompts and take you to the swap and copy functions.

Store

```
Copy  Multi : 127  (24 char name of program)
   to  Multi : 126  (24 char name of program)
```

Store

```
Swap  Multi : 127  (24 char name of program)
 with  Multi : 126  (24 char name of program)
```

Store

```
Copy  Inst : 73    (24 char name of instrument)
   to  Inst : 128   (24 char name of instrument)
```

Store

```
Swap  Inst :  23   (24 char name of instrument)
 with  Inst : 128   (24 char name of instrument)
```

These displays work like many of the others in this module—use the **Data Adj** knob to change the Instrument number and press **Enter** to complete the operation, or press any Selection Button to bring up a different display and exit without storing or moving anything.

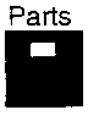
COMPARE BUTTON

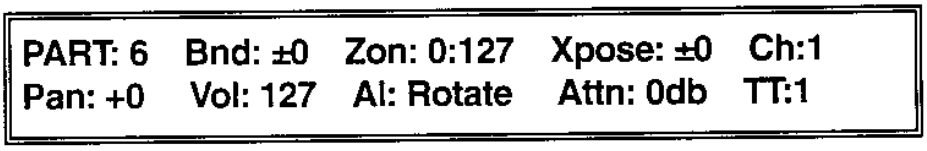
Switches the current Instruments between the stored version and the one in the edit buffer. When the button is lit, you're listening to the stored

version, otherwise you're listening to the version that you've changed. Useful for comparing the effects of changes you've made with the original program. The OB-Mx controls are inactive in the Compare mode (Compare button lit). Press the Compare button a second time to return control of the OB-Mx.

PARTS BUTTON

Only active in Multi mode. This is used to edit most of the parameters that go into a Part. You can change which Part is displayed by pressing the numbered buttons 1-6 in the VOICE STATUS module.





▶ *Parameters*

Part

Use this for changing the Part that is displayed, and for activating and deactivating Parts.

The Part number can be 1-6, and you change it either by using the **Cursor** < and > buttons and the **Data Adj** knob or by simply pressing a Part Number button in the VOICE STATUS module.

If a Part is active, the word "PART" is displayed in all-capital letters. Also, the **Esc** button will flash when the cursor is positioned in the Part field of an active Part. Pressing **Esc** will deactivate the Part.

If a Part is deactivated, the word "Part" is displayed with only the first letter capitalized. Also, the **Enter** button will flash when the cursor is positioned in the Part field of an active Part. Pressing **Esc** will deactivate the Part.

See Explorations 2 and 3 for information about using Parts.

Bnd

Determines the amount of bend for a part. It can be set to "Def" to use the the default mode in the MIDI menu.

Zon

Sets the range of incoming MIDI note numbers that the Part will respond to. It's useful for creating keyboard splits or for limiting the range of an Instrument to a particular set of notes. The number in the first field refers to the lowest note in the range, and the second to the highest. The possible values for each are from 0 to 127.

There are two ways to change either the lower or upper value for a particular Zone. The first is simply to use the **Cursor** buttons and **Data Adj** knob, as with the other parameters in this display. The other way is to move the cursor to the field you want to change, press **Enter** and then play the desired boundary note on your MIDI controller If you play the wrong note, just play the right one—the value will keep changing in response to Note Ins until you press either the **Escape** button or exit the field

Xpose

The number of semitones added to incoming note values.

Ch

The MIDI channel for the Part. The Part will only respond to MIDI messages coming in on this channel. Several Parts can share the same channel. Can be 1 to 16.

Pan

The overall panning for the part, from -10 (full left) to +10 (full right), with 0 being the center. This is added to the Pan parameter in the Part's Single Instrument to determine the final placement in the stereo field.

Vol

The overall output volume for the Part. The range is 0 to 127, and will change in response to incoming MIDI Volume (Continuous Controller 7) messages on the Part's MIDI channel. Also see the description of the Attn parameter below.

Al

See the Voice Allocation chapter at the start of this section.

Attn

Stands for *Attenuation*. Controls the relative volume of a Part. It is used to create a balance among the several Parts that may be set to respond on a single MIDI channel. In such a case, MIDI Volume messages would raise or lower the overall volume of the stacked sounds without altering the balance set by the Attn parameter. See *Exploration 3* for more details.

TT

The number of the active tuning table for this Part. This will only effect VCOs in the Single Instrument assigned to the Part which have their *Track* parameters set to TT.

MATRIX BUTTON

This allows you to map modulation sources to destinations. For an introduction to its use, see "Modulation & The Matrix" in the *Explorations* chapter. Each such mapping is called a *Patch*. Patches are stored with Single Instruments, not with Parts. Each Single Instrument can have up to 12 Patches.

When several Matrix Patches modulate a single destination parameter, their effect is additive, within limits. The destination parameter cannot, of course, exceed its range; therefore, if one controller has already pushed a

parameter to the edge (say, a Mod Wheel has pushed a filter frequency to its maximum or minimum), then any additional modulation in that direction will have no effect.

Patch : 12 Source : ModWheel Multiplier : +1.00
Destination Module : OB VCF Param : Frequency

▶ Parameters

Patch

The Patch number, from 1 to 12.

Source

The modulation source. It can take the following values:

- **NONE:** The Patch will be inactive and have no effect.

MIDI Modulation Sources

- **Pressure:** MIDI Channel Pressure, also known as Aftertouch
- **Velocity:** The velocity of each MIDI Note On message
- **Note #:** The pitch of each MIDI Note On message
- **ModWheel:** MIDI Continuous Controller 1, usually sent automatically by Mod Wheels on keyboards
- **PitchWhl:** MIDI Pitch Bend messages, usually sent automatically by Pitch Bend Wheels on keyboards
- **Breath:** MIDI Breath Controller, Continuous Controller 2
- **Pedal:** Expression pedal, Continuous Controller 11
- **Volume:** MIDI volume, Continuous Controller 7
- **MIDI A:** Any MIDI Continuous Controller that is assigned to the A modulation source in the display brought up by pressing the MIDI button
- **MIDI B, C, D:** Similar to MIDI A.

Internal Modulation Sources

- **LFO 1, 2 or 3:** The output of LFO 1, 2 or 3
- **EnvGen 1, 2, 3 or 4:** The output of envelope generator 1, 2, 3 or 4
- **Random:** A random number generator that produces a new random number every time a Note In message is received for its voice.

Multiplier

Determines the amount of modulation effect that the source will have on the destination. Takes values between -1.00 and +1.00, with 0 meaning no modulation will occur. A value of +1.00 means that the modulator will have its maximum effect on the destination, while a value of -1.00 means that the modulator will have its maximum inverse effect (i.e., as the source increases, the value of the destination parameter decreases).

Destination Module

This can be any of the voltage-producing modules. You can select a VCO, VCF, LFO, EG, or the Mixer to receive modulation. As you change modules, different choices become available in the Param field.

Param

Lets you choose which parameter to modulate in the destination module. Almost any parameter for which there is a knob may be modulated.

OPTIONS BUTTON

Sets system-wide options.

Option

Display : 4 Tune A : 440 GlobalTranspose : +0
EditTuneTable : 1 Demo : 6 SystemSize : 12v

The settings in the top line, as well as System Size are stored as Global variables.

▶ Parameters

Display

Sets the viewing angle for the display. Turn this so that the display is most easily read from your usual position.

Tune

Moves the pitch of all VCOs in the synthesizer. The number represents the pitch (in Hz) of the A below middle C. A value of 440 represents standard tuning. The Tune is variable from 415 to 466 Hz , corresponding to ± 1 semitone.

Global Transpose

The OB-Mx adds this number to all incoming MIDI note values before any other processing. From -60 to +60 semitones.

EditTuneTable

Press Enter when the cursor is in this field to edit the tuning tables. Note that simply editing a Tuning Table doesn't activate it—it also must be the chosen table for a Part or for the current Single Instrument, and the Single Instrument's VCO's must have their tracking parameters set to "TT."

There are 19 Tuning Tables in the OB-Mx. Tuning Tables 10-19 are stored in ROM and cannot be altered. Table 10 is the familiar equal temperament, with each octave divided into 12 equal parts, and is the Tuning Table for most of the ROM instruments. Tables 1-9 are stored in RAM and can be edited by users. Initially Tuning Tables 1-9 are duplicates of tables 11-19.

When you edit a Tuning Table, you can map each incoming MIDI key number to any pitch.

Tuning Table: 1 *<24 Character TT Name>*
MIDI Key #: 64 Pitch: C4 Cents Dev: 00

Tuning Table

Tells you which table you're editing. You can dial in any number from 1-19 here, but tables 10-19 can only be examined, not edited.

You can only dial in a new tuning table (without exiting to the Options menu and returning) if you haven't made any changes to the currently-displayed table. If you've made any changes and attempt to dial in a new table number, you'll be asked whether you want to save your changes. Pressing **Esc** will exit to the Options menu without saving any changes, while pressing **Enter** will save the changes and also exit to the Options menu.

- **Unit:** Tuning Table number
- **Default**: 1
- **Range:** 1 to 19

MIDI Key

This is the key number whose frequency is determined by the remainder of the display line. When this tuning table is active (both in the Part or the Single Instrument, and only if the VCO tracking parameters are set to "TT"), then an incoming MIDI note with this number will play a note with a pitch equal to the equal-tempered pitch that would be produced by the note indicated in the Pitch field offset by the number of cents (100ths of a semitone) indicated in the Cents Dev field.

- **Unit:** MIDI Key number
- **Default**: 1
- **Range:** 0 to 127

Pitch

The base pitch that will be produced by an incoming MIDI note number. Think of this as a rough or coarse pitch, which can be fine-tuned by the Cents Dev parameter.

- **Unit:** Note name and octave
- **Default**: Varies from ROM table to table. In the standard, equal-tempered Tuning Table 10, the note name and octave correspond to the values that would conventionally be called up by the MIDI Key numbers to the left.
- **Range:** 1 to 127

Cents Dev

The number of cents (1/100th of a semitone) to add to or subtract from the pitch indicated in the Pitch field when the note is triggered by the incoming note number indicated by the MIDI Key field.

- **Unit:** Cents
- **Default**: Varies from ROM table to table. In the standard, equal-tempered Tuning Table 10, all values are 0.
- **Range:** 0 to +99

Demo

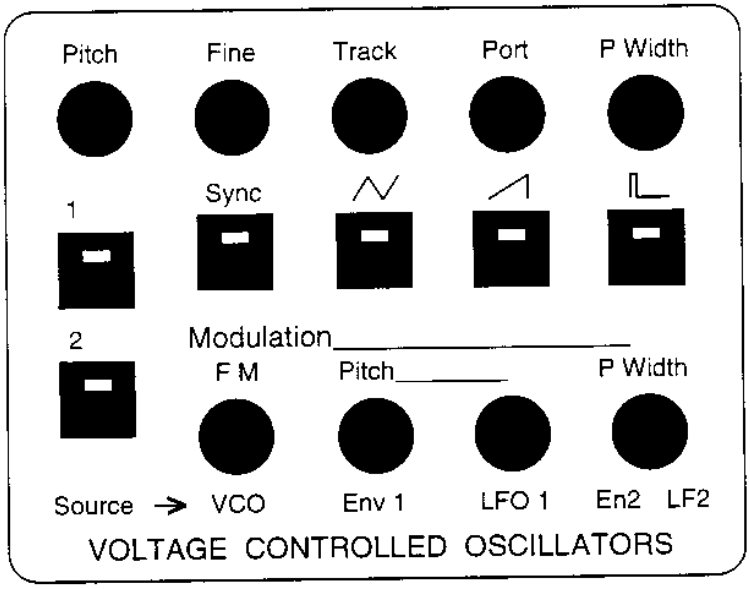
8 brief demonstration pieces may called up and listened to. Move the cursor into the **Demo** parameter area, use the **Data Adj** knob to choose one, and press **Enter** to hear it play. The Demo will continue to cycle through until you press the **Escape** button.

System Size

This is the number of voices in the system. It needs to be set manually when you install or remove voices. The value should be twice the number of voice cards (two voices per card) in the system. Refer to the "OB-Mx Two Voice Card Isntructions for Use and Installation" that came with any additional Voice Cards for more information.

CHAPTER 7

Voltage Controlled Oscillators Module



The **OB-Mx** uses 2 voltage-controlled oscillators per voice. Each of these oscillators is capable of producing a triangle wave, a sawtooth, and a variable-width pulse wave simultaneously.

SELECTION BUTTONS

The buttons numbered 1 and 2 select which VCO is being edited—in other words, the lit selection button determines which VCO will change as a result of turning a knob or pressing a Parameter Toggle button in the module. If the button is also flashing, its parameters are currently visible in the display.

PARAMETER TOGGLES

Sync

An effect that uses one oscillator's (source) output to alter the other (destination). When Sync is on in the destination oscillator, it causes the oscillator's cycle to restart every time the source oscillator's output starts a new cycle (see Figure 7.1). This results in a classic timbral effect.

A close examination of the Figure will reveal that the destination oscillator has a new cycle, with a period equal to that of the source oscillator. This is a fancy way of saying that the pitch of the source oscillator determines the fundamental pitch of the destination, although there will be overtones in the destination's output that are not exact multiples of the fundamental frequency (enharmonic overtones).

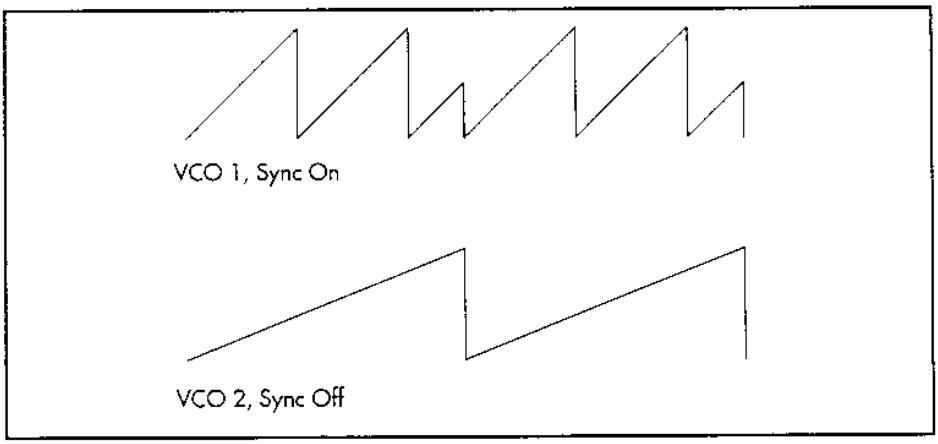


FIGURE 7.1
The effect of syncing VCO 1 to VCO 2. The third cycle in VCO 1's output is cut short when VCO 2 starts its second cycle.

Either VCO can be the destination. You can even make them *both* destinations, but the results will be unpredictable.

Triangle Wave

This turns the triangle wave output of the oscillator on and off. Triangle waves are rich in all harmonics.

Sawtooth Wave

This turns the sawtooth wave output of the oscillator on and off. Sawtooth waves are rich in even harmonics and tend to sound "reedy."

Pulse Wave

This turns the pulse wave output of the oscillator on and off. Pulse waves are exceptionally flexible for synthesis purposes, because it's easy to change their tonal content without affecting their pitch. This is done through a process called *Pulse Width Modulation*. See Exploration 7 for more information.

Continuous Parameter Knobs

Pitch

Determines the number of semitones (half-steps) to add to the fundamental frequency when the sound is triggered. When set extremely high or low, the sounds may pass out of the range of human hearing.

- **Unit:** Semitones
- **Default**: 0
- **Range:** -64 to +63

Fine

Adds or subtracts fractions of a semitone from the oscillator's pitch.

- **Units:** cents (1/100th of a semitone)
- **Default:** 0
- **Range:** -97 to +97 cents

Track

Controls the degree to which the oscillator's fundamental pitch follows the keyboard's (i.e., the incoming MIDI note numbers). When set to +1.00, the oscillator will track the keyboard exactly, yielding an equal-tempered scale with 12 semitones per octave. When set to 0, the note number will have no effect on the pitch generated, which will be fixed at middle C unless either the Pitch or Fine parameter is changed from its default value of 0.

When set to TT, each incoming MIDI note will be used as an index into the active tuning table. The oscillator's pitch will be read from the table. Changes to the VCO's Pitch parameter will cause a different table value to be read. For instance, if Pitch is set to +2 and a MIDI note 60 comes in, then the actual pitch will be read from the tuning table's value corresponding to note 62. Non-zero values in the Fine parameter will be interpreted as usual—the number of cents in the Fine parameter will be added to the pitch read from the tuning table.

There are 19 Tuning Tables in the OB-Mx. Editing instructions can be found in the Reference section relating to the Options button in Chapter 6.

The active Tuning Table is determined by a Single Instrument Global variable when in Single Instrument mode, or is set in the Parts display when in Multi mode. Each Part may use a different Tuning Table.

- **Units:** Ratio of keyboard change to oscillator pitch change (when not TT)
- **Default:** +1.00
- **Range:** -1.00 to +1.00, TT

Port

Controls the *portamento time*. This is an effect where the pitches between successive notes do not change instantly and discretely, but rather glide from the first to the second. If the value is greater than 0, the gliding effect will take place over the time period indicated. This is not an effect that you'd want to use all the time, most likely, so the default value of 0 means that there is no portamento effect, and pitch changes occur instantly when you play a note.

Portamento is essentially a monophonic effect. If you use portamento in Single Instrument mode with the Rotate algorithm in effect, you're likely to experience unforeseen behavior. When a new note is played, a voice is

allocated for it and the pitch glides from the previous pitch *in that voice* to the new pitch. Whether in a Single or Multi mode, you're best off using portamento with an assignment algorithm that allocates all the voices to each note. In a Multi, if you just assign one voice to a Part with portamento, it won't make a difference which algorithm you use in that Part.

- **Units:** seconds
- **Default:** 0 (no portamento effect)
- **Range:** .001 to 30 seconds

P Width

This only has an effect if the pulse wave output is turned on. See *Exploration 7* in Chapter 3 for a detailed explanation of its usage.

- **Units:** % of time pulse wave spends in "Up" position
- **Default:** 50
- **Range:** 0 to 100

Modulation Section

FM/VCO

The frequency of the VCO being edited can be changed continuously by the output of the other VCO. This results in timbral and pitch variations unique to FM. The ratio between the pitch of the

- **Units:** % of total modulation amount
- **Default:** 0
- **Range:** 0 to 100

Pitch/Env 1

Envelope 1 is automatically mapped to modulate oscillator pitch. If the value is 0, the envelope will have no effect. If the multiplier is 1, then the envelope can change the pitch over a range of approximately 4 octaves.

- **Units:** Multiplier
- **Default:** 0
- **Range:** -1.00 to +1.00

Pitch/LFO 1

LFO 1 is automatically mapped to modulate oscillator pitch. If the value is 0, the envelope will have no effect. If the multiplier is 1, then the LFO can change the pitch over a range of approximately 4 octaves.

Make this parameter greater than 0 to introduce a vibrato effect. Also see Exploration 6.

- **Units:** Multiplier
- **Default:** 0
- **Range:** -1.00 to +1.00

P Width/Env — LFO

This only has an effect if the pulse wave output is turned on. It controls the amount by which either envelope 2 or LFO 2 will modulate the pulse width. The center (straight up) position means that no pulse width modulation will take place. Moving the knob to the right of center will increase the amount of pulse width modulation due to LFO 2, and moving the knob to the left of center will increase the amount of pulse width modulation due to Envelope 2. Clearly, then, you can choose to modulate the pulse width with one or the other of these sources but not both, unless you also use the Matrix to achieve a second modulation.

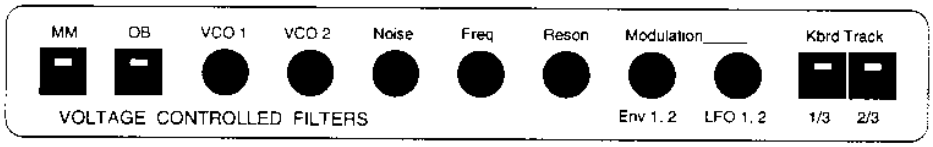
Pulse Width Modulation is additive with the basic pulse width parameter. If the value of the Pulse Width parameter is 50, a value of LF2>Pw:1.00 means that LFO 2, as it cycle from its minimum value to its maximum value, will change the pulse width from 0 to 100 (where "pulse width" means the % of time spent in the up position). If the value of the Pulse Width parameter is 50, a value of En2>Pw:0.50 means that envelope 2 will cause the pulse width to be 25 when envelope 2 is 0 and 75 when (and if) Envelope 2 reaches the maximum level of 100.

See *Exploration 7* in Chapter 3 for more details.

- **Units:** Choice of Envelope 2 or LFO 2 and scaling factor
- **Default:** 0
- **Range:** En2>0.00 to +1.00, LFO2>0.00 to +1.00

CHAPTER 8

Voltage Controlled Filters (VCF) Module



This module controls the behavior of the MM and OB filters. The **MM Filter** is a low-pass filter designed to recreate the sounds of the **Mini-Moog®** synthesizer. It's what's called a "4-pole filter," which means that it attenuates frequencies by 24 dB per octave (in other words, frequencies one octave above the cutoff frequency are 24 dB quieter when they leave than when they entered, and frequencies two octaves above cutoff are 48 dB quieter). The **OB Filter** is a "state-variable" filter which simultaneously produces low-pass, high-pass and band-pass outputs. Its slope is 12 dB per octave.

The balance among the 4 different filter outputs is controlled in the **FINAL MIX** module.

MM & OB Selection Buttons

The MM and OB buttons select which VCF is being edited. The lit selection button determines which VCF will change as a result of turning a knob or pressing a Parameter Toggle button in the module. If the button is also flashing, its parameters are currently visible in the display.

The MM button calls up this display:

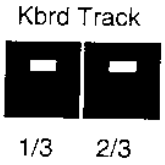

```
M M  Inputs ..... Frq Res Modulation .....
1:100 2:100 N :100 +63 100 En1:+1.00 LF1:-1.00
```

The OB button calls up a similar display:


```
OB  Inputs ..... Frq Res Modulation .....
1:100 2:100 N :100 +63 100 En2:+1.00 LF2:-1.00
```

The main difference between the displays is that EG 1 and LFO 1 are default modulators for the MM filter, while EG 2 and LFO 2 are default modulators for the OB filter.

PARAMETER TOGGLES



Kbrd Track 1/3 and 2/3

Determines how the filter's cutoff or center frequency follows the keyboard. The buttons are additive—when they're both lit, the filter's frequency will be the frequency of the MIDI Note In value. When only one button is lit, the filter's frequency will track the input pitch proportionally.

The formula is:

filterChange=<pitch of MIDI Note In - pitch of middle C><scaling factor>*

where

filterChange is the number of Hz to be added or subtracted to the filter's frequency. It adds to any other modulations that may be in effect.

pitch of middle C = 523.25 Hz

scaling factor = 1/3 or 2/3, depending on which button is lit.

CONTINUOUS PARAMETER KNOBS

VCO 1

The percentage of VCO 1's output that is input to the selected filter.

- **Unit:** Percentage
- **Default:** for the MM filter, 100; for the OB filter, 0
- **Range:** 0 to 100

VCO 2

The percentage of VCO 2's output that is input to the selected filter.

- **Unit:** Percentage
- **Default:** for the MM filter, 0; for the OB filter, 100
- **Range:** 0 to 100

Noise

The percentage of the Noise Generator's output that is input to the selected filter.

- **Unit:** Percentage
- **Default:**0
- **Range:** 0 to 100

Freq

Determines the number of semitones to add to the filter frequency. When set extremely high or low, the sounds may pass out of the range of human hearing. When set to 0, with keyboard tracking off and no other VCF modulation, the filter's frequency would be that of middle C.

- **Unit:** Semitones
- **Default:** +28.5
- **Range:** -15.5 to +48.0

Reson

Changes the filter's resonance. It boosts the filter's output in a narrow range centered at the filter's cutoff or center frequency. High values produce useful "ringing" effects (try using MIDI Velocity to modulate this parameter). Extremely high values cause the filter to oscillate, creating an additional sound source.

- **Unit:** Percentage of total boost
- **Default:** 0
- **Range:** 0 to 100

Env 1, 2

This scales the amount by which EG 1 (for the MM filter) or EG 2 (for the OB filter) will modulate the frequency of their respective filters. Among other applications, you can use this to create filter sweeps.

- **Unit:** Multiplier
- **Default:** +0.00
- **Range:** -1.00 to +1.00

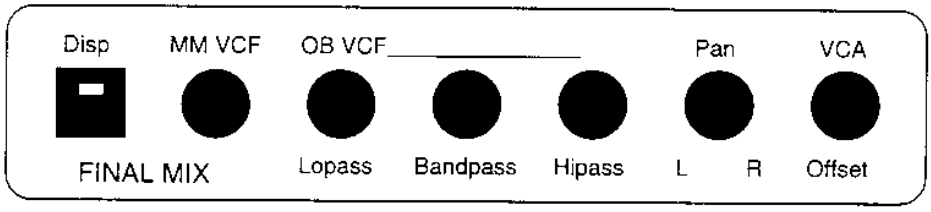
LFO 1, 2

This scales the amount by which LFO 1 (for the MM filter) or LFO 2 (for the OB filter) will modulate the frequency of their respective filters. Among other applications, you can use this to create dynamic timbral variations for sustained notes.

- **Unit:** Multiplier
- **Default:** +0.00
- **Range:** -1.00 to +1.00

CHAPTER 9

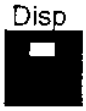
Final Mix Module



This module mixes the outputs of the VCFs to a mono signal and pans the resultant signal in the stereo output field. It's also responsible for boosting the base output level of the VCA, a function particularly useful for self-playing patches.

Disp Selection Button

Brings the Mixer parameters into the display:




```
MM  OB.....  Pan.....  VCA
100 LP:100 BP:100 HP : 99  - 8 L< |     >R  100
```

▶ Continuous Parameter Knobs

MM VCF

The percentage of the MM filter's output that is mixed into the final output signal.

- **Unit:** Percentage
- **Default:** 100
- **Range:** 0 to 100

OB VCF Lopass

The percentage of the OB filter's lowpass output that is mixed into the final output signal.

- **Unit:** Percentage
- **Default:** 100
- **Range:** 0 to 100

OB VCF Bandpass

The percentage of the OB filter's bandpass output that is mixed into the final output signal.

- **Unit:** Percentage
- **Default:** 0
- **Range:** 0 to 100

OB VCF Hipass

The percentage of the OB filter's highpass output that is mixed into the final output signal.

- **Unit:** Percentage
- **Default:** 0
- **Range:** 0 to 100

Pan L/R

Positions the final output in stereo space by sending possibly different portions of it to each of the main output jacks.

- **Unit:** -10=full pan left, 0=center, +10=full pan right
- **Default:** 0
- **Range:** -10 to +10

VCA Offset

This boosts the VCA output. Its most frequent use is probably for creating self-playing Instruments. In this case, the VCOs oscillate continually, regardless of whether the VCA is turned down to 0 or up to 100. For traditionally-played notes, a Note In triggers EG 4, which turns up the VCA output and lets the note out (a water pipe is a good analogy, with the water pressure always there, the faucet as the VCA output level and your hand as EG 4).

- **Unit:** Percentage of maximum VCA level
- **Default:** 0
- **Range:** 0 to 100

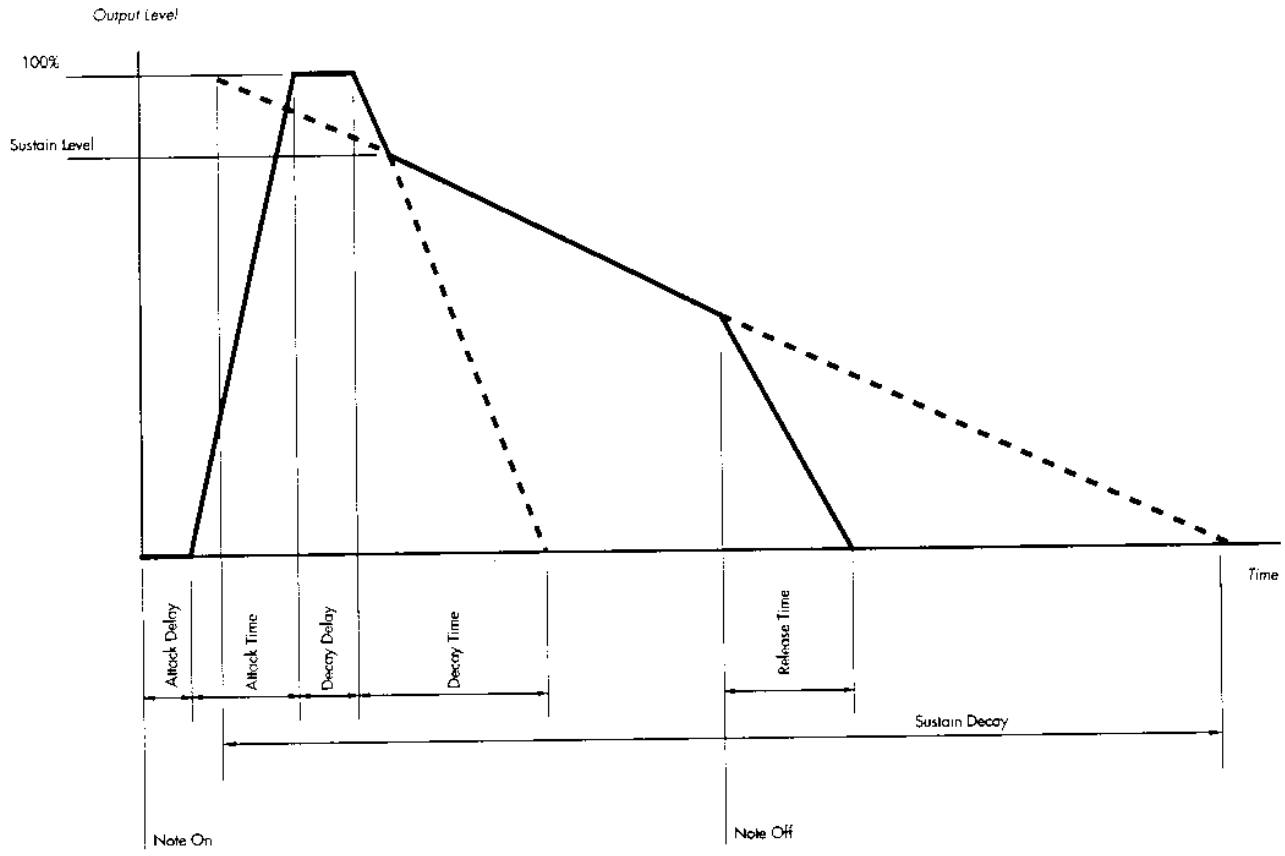
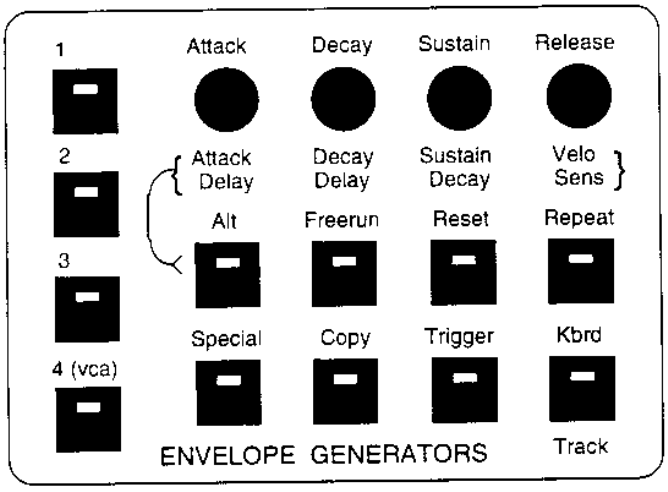


FIGURE 10.1
An OB-Mx envelope

CHAPTER 10

Envelope Generators (EG) Module



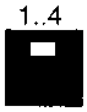
This module controls the 4 envelope generators available to each voice. It also has a a convenient **Trigger** button for playing notes from the front panel.

Envelope 4 is, by default, routed to control each voice's VCA (and hence its volume). Envelopes 1 and 2 also have default routings, which are indicated in Table 3.1 on page 25.

Figure 10.1 shows an **OB-Mx** envelope with all its numeric parameters. We'll refer often to this diagram in this section.

Envelope Selection Buttons 1–4

These buttons select which envelope generator (EG) is being edited. The lit selection button determines which EG will change as a result of turning a knob or pressing a Parameter Toggle button in the module. If the button is also flashing, its parameters are currently visible in the display.



Atck .002s	Dcay .75s	Sustain 100	Rels 4.7s
ADel .005s	DDel .033s	SDcay .75s	VSen 9 db

Parameter Toggles

Alt

This alters the functions of the knobs in the module. When the Alt button is unlit, the dials change the parameters printed above them. When the Alt button is lit, the knobs change the parameters printed below them.

Freerun

The entire envelope is performed regardless of when the Note Off occurs. This is often used for percussive sounds. When Freerun is on, if the Note Off occurs before the Sustain portion of the envelope has been entered, then the Attack and Decay portions complete normally and the envelope progresses to the Release stage immediately after the completion of the Decay. By contrast, if Freerun is off and the Note Off occurs before Sustain

has been entered, then the envelope proceeds immediately to the Release stage without completing the Attack and Decay.

Reset

Each new note restarts the envelope at the beginning of its cycle, from the value 0. If left off, each new note in a voice restarts the envelope from it's current level.

Repeat

Repeats the envelope if a Note Off message hasn't been received by the time the envelope completes its entire cycle. This cycle includes letting the Sustain Decay run the level down to 0 and then waiting through the Release stage. If the Sustain Decay is set to None while Repeat is turned on, then the Sustain stage is ignored and the envelope proceeds immediately from the end of the Decay stage to the Release.

This parameter can be used to create mandolin-like tremolo effects, among other things.

If both the Repeat and Freerun lights are lit in envelope 4, any note will repeat forever.

Special

This button is reserved for use by possible future software upgrades. Its current effect is to turn all the envelope generators into simple rectangle waves with minimum attack, decay and release times and a sustain level of 100. We advise against storing Instruments with this function active, as it may change in future releases.

Copy

Allows you to copy all the parameters of one EG to another.


```
Copy Envelope : 4 to Envelope : 1?
Escape : N0              Enter : YES
```

You can use the Cursor buttons and **Data Adj** knob to change both the source and destination envelopes. For this operation, it doesn't make any difference which EG Selection Button is lit.

Trigger

Pressing this button is equivalent to sending a MIDI Note On with pitch 60 (middle C) and velocity 64 to the OB-Mx on all channels with active Parts. Releasing the button is equivalent to sending the corresponding Note Offs.

Kbrd Track

Shortens all envelope times for higher notes and lengthens all envelope times for low notes. This simulates the properties of many acoustic instruments, such as pianos. The amount of time compression or expansion is proportional to the incoming note's distance from middle C. This function can also be accomplished by using a number of Patches in the Matrix, but this button is more convenient if you don't need to customize the time scalings.

CONTINUOUS PARAMETER KNOBS

Attack Delay

Time to wait between receiving a Note On and initiating the Attack portion of the envelope. See Figure 10.1.

- **Unit:** Seconds
- **Default:** .001
- **Range:** .001 to 30

Attack

The time it takes from the envelope to go from its starting value to its maximum value. See Figure 10.1.

- **Unit:** Seconds
- **Default:** .005
- **Range:** .001 to 30

Decay Delay

Time to wait after the end of the Attack portion before starting the Decay. See Figure 10.1.

- **Unit:** Seconds
- **Default:** .001
- **Range:** .001 to 30

Decay

The time it takes from the envelope to go from its maximum value at the end of the Attack stage to 0 *(See Figure 10.1)*. Since the Decay stage will not proceed all the way to 0 unless the Sustain level is 0, only a percentage of this time (100-Sustain Level %, to be exact) will actually elapse in the Decay stage.

- **Unit:** Seconds
- **Default:** .20
- **Range:** .001 to 30

Sustain

The level that the VCA reaches at the end of the Decay stage.

- **Unit:** Percentage of maximum level
- **Default:** 90
- **Range:** 0 to 100

Sustain Decay

The slope of the decay once the Sustain stage has been reached. Literally, this is the number of seconds that it would take the VCA to decay from 100 to 0 *(see Figure 10-1)*. If the Sustain level is near 100, think of this as the number of seconds before the sound dies out even as you hold the note. It can be used, for example, to simulate the gradual decay of a string after it has been plucked or struck but before it has been muted or damped. If the value is None, then the envelope remains at the Sustain level until the Release stage is initiated by a Note Off.

- **Unit:** Seconds
- **Default:** 11
- **Range:** .001 to 30, None

Release

The number of seconds that it takes the envelope to return to 0 after a Note Off has been received. A value of None means that the envelope will not return to 0, and will stay at the sustain level forever. If you use a Release time of None on EG 4, you'll produce a note that will sustain infinitely long. The only ways to shut the note off will be to turn off the **OB-Mx**, to load a new Instrument in place of the one that's sustaining or to steal the voice that's sounding with another Note On.

- **Unit:** Seconds
- **Default:** 0.52
- **Range:** .001 to 30, None

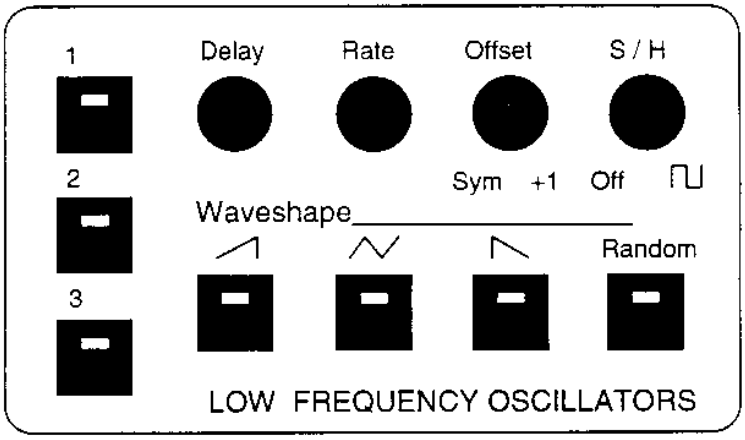
Velo Sens

Determines the envelope's responsiveness to changes in MIDI velocity. In particular, edit this parameter for EG 4 to control the way that note dynamics respond to changes in playing intensity. Specifically, this controls the amount by which the envelope's levels are attenuated (decreased) by incoming notes with velocities less than the maximum of 127. This parameter affects both the Attack level, which is 100 unless attenuated with this parameter, and the Sustain level. In precise terms, the value is the difference in decibels between the Attack level of a note with velocity 127 and that of a note with velocity 1.

- **Unit:** Decibels
- **Default:** 0
- **Range:** 0 to 25

CHAPTER 11

Low Frequency Oscillators (LFO) Module



This module controls the 3 low frequency oscillators (LFOs) available to each voice. LFOs are extremely useful modulation sources. LFOs 1 and 2 have default modulation routings, which are indicated in Table 3.1 on page 25.

Unlike the envelope generators, the LFOs are free-running; that is, they run continuously and are not triggered by incoming MIDI notes.

LFO Selection Buttons 1–3

These buttons select which LFO is being edited. The lit selection button determines which LFO will change as a result of turning a knob or pressing a Parameter Toggle button in the module. If the button is also flashing, its parameters are currently visible in the display.

LFO 2	Delay	Rate / Period		Ofst	S&H	WShape
	4.9 s	5.0Hz	.200s	100	Off	Random

Parameter Toggles

The parameter toggles in this module determine which waveshape the LFO generates. Since each LFO can only produce a single output, these buttons are mutually exclusive—pressing one shuts the others off.

Sawtooth

The LFO produces a sawtooth wave, starting from 0 and rising to 100 gradually, then abruptly dropping back to 0 and starting over again.

Triangle

The LFO produces a triangle wave, starting from 0 and rising to 100 gradually, then gradually dropping back to 0 and starting over again.

Reverse Sawtooth

The reverse of the first waveform. The LFO output starts at 100, gradually descends to 0, and then abruptly resets to 100 for the start of a new cycle.

Random

Random

The output of the LFO is constantly changing, buth with random direction and slope. The Rate knob controls the average rate of change, with higher values providing faster random change

CONTINUOUS PARAMETER KNOBS

Delay

When this is non-zero, it "fades in" the LFO effect over the number of seconds specified. See Figure 11.1.

- **Unit:** Seconds
- **Default:** .000
- **Range:** .000 to 30

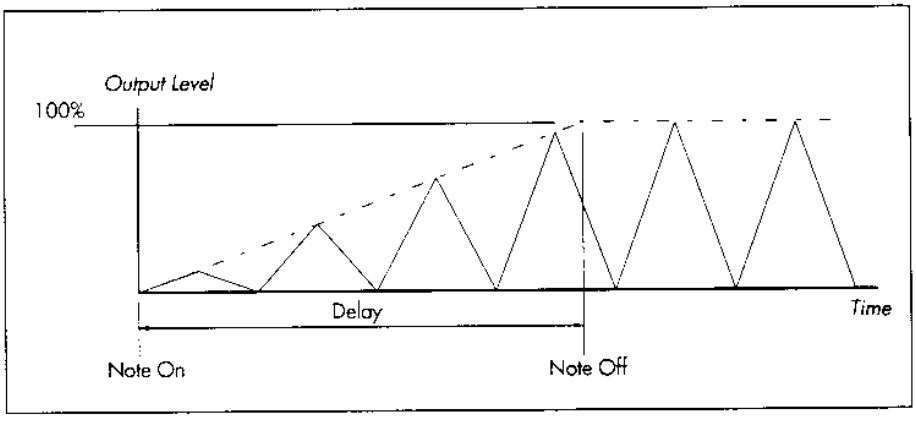


FIGURE 11.1
A non-zero value for LFO Delay creates a fade-in for the modulation effect.

Rate

The number of times the LFO goes through a complete cycle in one second, or the average rate of change for the Random waveform. Altering the Rate causes the **OB-Mx** to do a little math for you and display the Period, which is the time it takes for the LFO to complete a cycle (Period=1/Rate).

- **Unit:** Hz (cycles per second)
- **Default:** 5.0 Hz = 0.200 s period
- **Range:** .010 Hz to 30 Hz (= periods from 100 to 0.033 seconds)

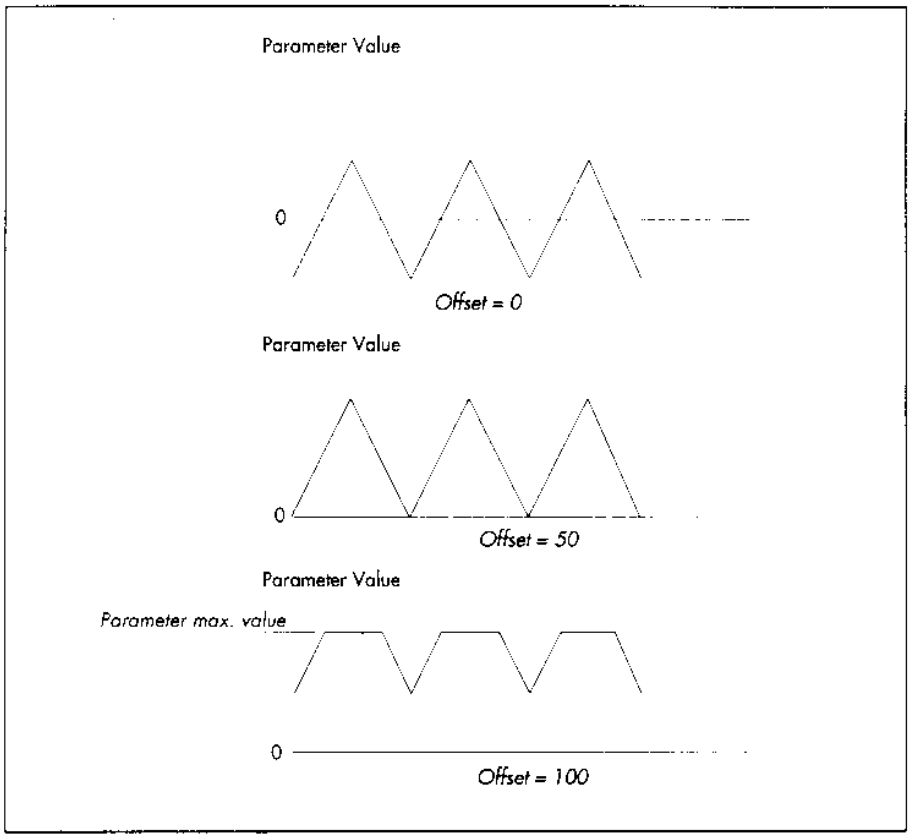


FIGURE 11.2
The effects of different offsets on LFO output.

Offset

A fixed amount to add to the LFO's output. When the Offset is set to 0, the LFO goes positive and negative; when set to 50, the LFO output ranges from 0 to its maximum value. Offsets higher than 50 will cause a "clipping" of the LFO's output when the maximum value is reached. *See Figure 11.2.*

- **Unit:** Percentage of total available offset
- **Default:** 50
- **Range:** 0 to 100

S/H

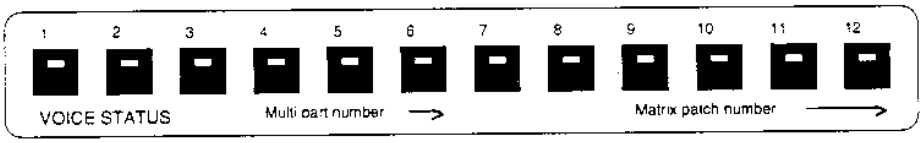
This control, when it is not turned Off, quantizes the LFO's output into a discrete number of steps. When set to 2, the quantized LFO output becomes a square wave (provided the LFO waveshape is not Random). A common usage for this parameter would be in self-playing Instruments where an LFO is used to modulate a VCO's pitch. In this application, using S/H would create discrete individual pitches rather than the sliding pitches that you would get with the S/H parameter set to Off.

S/H is named after a traditional analog synthesis technique called *sample and hold.* OB-Mx S/H produces effects similar to that of the traditional sample and hold circuits, but the way that it achieves them is quite different. A traditional sample and hold circuit has just one parameter, time between samples. What it does is to measure the LFO's output at a given time, and put out that one value until the time between samples has elapsed. Then it measures the LFO output again and repeats the process. This could be called quantization in the time domain, whereas the OB-Mx S/H function is quantization in the voltage domain, and sends out a new value whenever the LFO output crosses a quantization threshold.

- **Unit:** Number of equally-spaced values to output
- **Default:** Off
- **Range:** 32 to 2, Off

CHAPTER 12

Voice Status Module



The buttons in this module serve multiple purposes, as follows:

- To indicate voice usage by incoming MIDI notes
- To bring the parameters for different Parts into the display for editing
- To bring the parameters for different Matrix Patches into the display for editing

VOICE USAGE INDICATORS

Whenever MIDI notes come into the **OB-Mx** or the **Trigger** button is pressed, the **VOICE STATUS** buttons show, not surprisingly, the status of the voices. That is, when a MIDI Note On message comes in, the numbered button(s) for the voice(s) that it is assigned to light(s) up (if the buttons had been lit because of one of their other uses, the ones for unassigned voices shut off automatically at this time. It doesn't affect the Part or Patch being edited). When a MIDI Note Off message comes in to shut off a sounding voice, the light goes out.

There is one subtlety to be aware of here: if a note is played in a Part whose Single Instrument has a long release time, the voice for that note will actually remain in use after the Note Off message is received (until the Release stage of Envelope 4 is completed), and hence after the Voice Status light goes out. Additional incoming notes may steal a sounding note even though the Voice Status button for that voice is unlit.

PART SELECTION BUTTONS

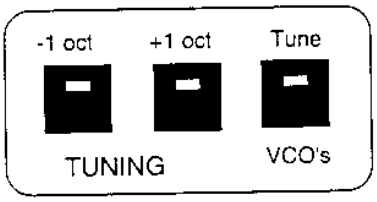
When in Multi mode, pressing any of the buttons 1-6 will bring the parameters for that Part into the display if either the **Program** or **Parts** button in the **CONTROL MODULE** is lit.

PATCH SELECTION BUTTONS

Pressing any of the buttons 1-12 will bring the parameters for that Patch into the display if the **Matrix** button in the **CONTROL MODULE** is lit.

CHAPTER 13

Tuning Module



The buttons in this module change the pitch of the VCOs.

-1 oct

Transposes all oscillators down one octave.

+1 oct

Transposes all oscillators up one octave.

Tune VCOs

Tunes all the oscillators. The OB-Mx performs this operation automatically each time it's turned on. If you suspect that the tuning of your VCOs has drifted, press this button. You shouldn't have to use it often.

OB-Mx User's Guide

Section 3

Apendices & Charts

Oberheim®

OB-Mx User's Guide

Index

▶E

▶F

▶G

▶H

▶I

▶K

▶L

▶M

▶N

▶O

▶P

▶Q

▶R

▶S

▶T

▶U

▶V

▶X

▶Z

Voltage Controlled Oscillators

1	Pitch	Fine	Track	Port	PW
	F.M.	Sync ▢	▢	▢	▢
	En1 > Pch	LF1 > Pch	LF2 ▢ En2 ▢ > PW		
2	Pitch	Fine	Track	Port	PW
	F.M.	Sync ▢	▢	▢	▢
	En1 > Pch	LF1 > Pch	LF2 ▢ En2 ▢ > PW		

Voltage Controlled Filters

	VCO1	VCO2	Noise	Freq	Reson	Modulation		Track	
MM						Env1	LFO1	1/3 ▢	2/3 ▢
OB						Env2	LFO2	1/3 ▢	2/3 ▢

Final Mix

MM	OB			Pan	VCAoffset
	LP	BP	HP		

Low Frequency Oscillators

	Delay	Rate	Offset	S & H
1	▢	▢	▢	Random ▢
2	▢	▢	▢	Random ▢
3	▢	▢	▢	Random ▢

Matrix Patches

	Source	Mult	Module	Parameter
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				
11				
12				

Envelope Generators

	Attack A Delay	Decay D Delay	Sustain S Decay	Release Vel Sens
1				
	Spec ▢ Free ▢ Rest ▢ Rept ▢ Trck ▢			
2				
	Spec ▢ Free ▢ Rest ▢ Rept ▢ Trck ▢			
3				
	Spec ▢ Free ▢ Rest ▢ Rept ▢ Trck ▢			
4				
	Spec ▢ Free ▢ Rest ▢ Rept ▢ Trck ▢			

Instrument Name (24 characters)

#	

OBMx Instrument Definition Record

Multi Program #	

PART 1	Instrument #			
BendRange:	KeyboardZone: :		Transpose:	Channel:
Pan:	MIDI Vol not stored	Algorithm:	Attenuation:	TuningTable:

PART 2	Instrument #			
BendRange:	KeyboardZone: :		Transpose:	Channel:
Pan:	MIDI Vol not stored	Algorithm:	Attenuation:	TuningTable:

PART 3	Instrument #			
BendRange:	KeyboardZone: :		Transpose:	Channel:
Pan:	MIDI Vol not stored	Algorithm:	Attenuation:	TuningTable:

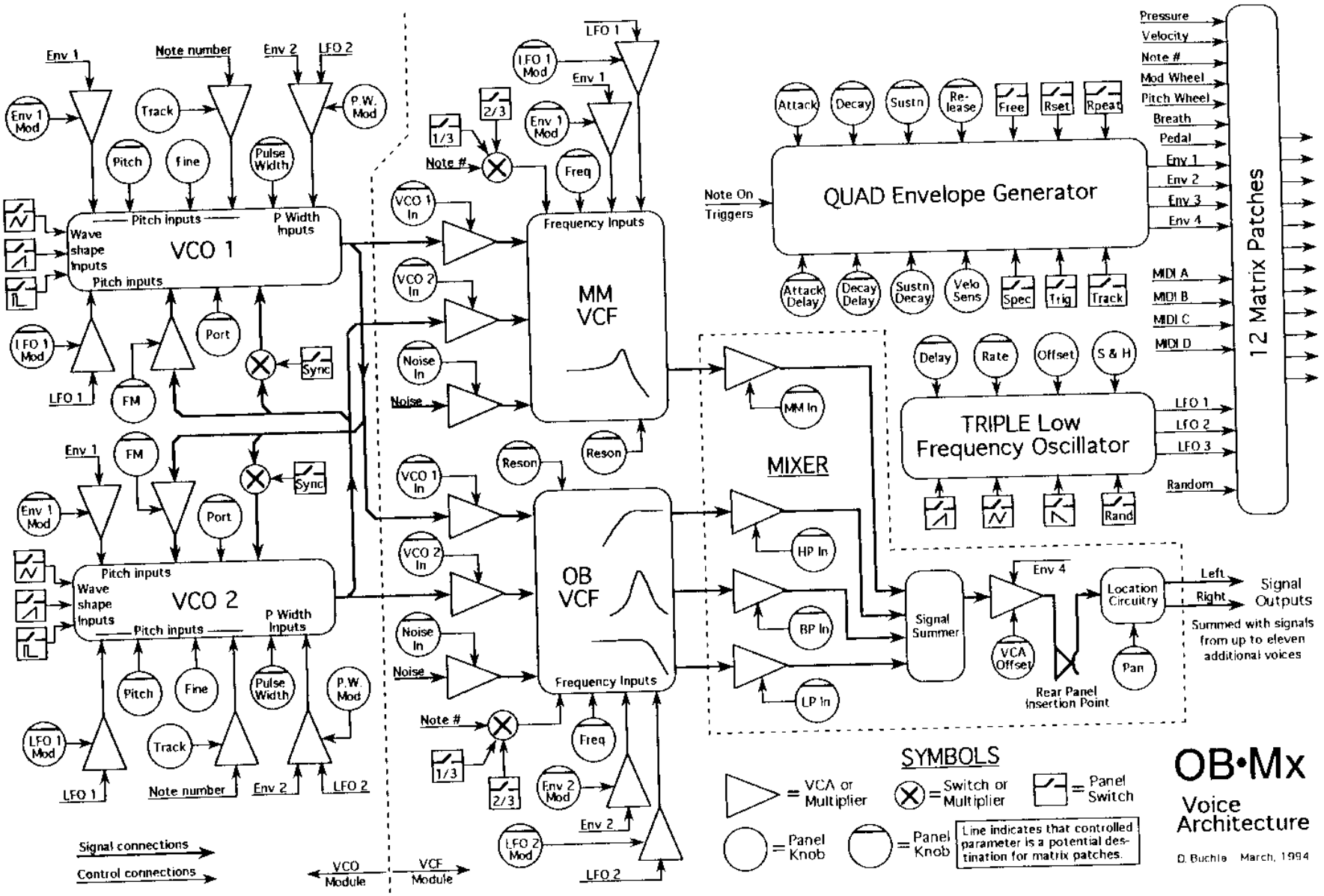
PART 4	Instrument #			
BendRange:	KeyboardZone: :		Transpose:	Channel:
Pan:	MIDI Vol not stored	Algorithm:	Attenuation:	TuningTable:

PART 5	Instrument #			
BendRange:	KeyboardZone: :		Transpose:	Channel:
Pan:	MIDI Vol not stored	Algorithm:	Attenuation:	TuningTable:

PART 6	Instrument #			
BendRange:	KeyboardZone: :		Transpose:	Channel:
Pan:	MIDI Vol not stored	Algorithm:	Attenuation:	TuningTable:

Assign:	Voice:	1	2	3	4	5	6	7	8	9	10	11	12
	Part:												

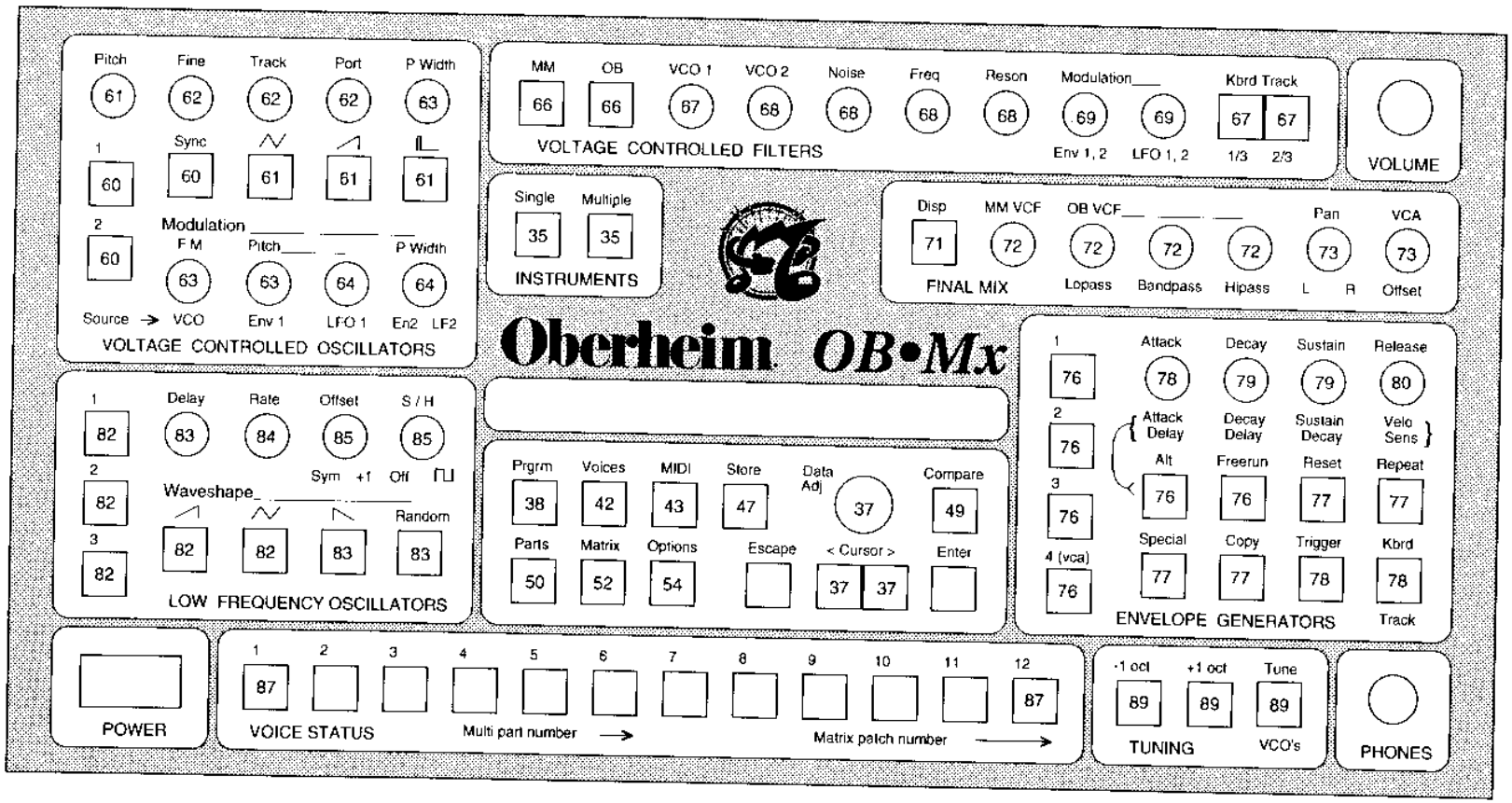
OB Mx MultiProgram Record


Env 1
Note number
Env 2
LFO 2
Env 1 Mod
Track
P.W. Mod
Pitch
Fine
Pulse Width
Pitch Inputs
P Width Inputs
Wave shape Inputs
VCO 1
Pitch inputs
LFO 1 Mod
Port
Sync
LFO 1
FM
LFO 1
1/3
2/3
Note #
Freq
VCO 1 In
VCO 2 In
Noise In
Noise
Frequency Inputs
MM VCF
Reson
MM In
MIXER
Env 1
FM
Sync
Env 1 Mod
Port
Pitch inputs
Wave shape Inputs
VCO 2
P Width Inputs
Pitch inputs
Pitch
Fine
Pulse Width
P.W. Mod
LFO 1 Mod
Track
LFO 1
Note number
Env 2
LFO 2
OB VCF
HP In
BP In
LP In
Signal Summer
Env 4
VCA Offset
Rear Panel Insertion Point
Location Circuitry
Pan
Left
Right
Signal Outputs
Summed with signals from up to eleven additional voices
Note On Triggers
QUAD Envelope Generator
Attack
Decay
Sustn
Re-lease
Free
Rset
Rpeat
Attack Delay
Decay Delay
Sustn Decay
Velo Sens
Spec
Trig
Track
Pressure
Velocity
Note #
Mod Wheel
Pitch Wheel
Breath
Pedal
Env 1
Env 2
Env 3
Env 4
MIDI A
MIDI B
MIDI C
MIDI D
12 Matrix Patches
Delay
Rate
Offset
S & H
TRIPLE Low Frequency Oscillator
LFO 1
LFO 2
LFO 3
Random
Rand
Signal connections
Control connections
VCO Module
VCF Module
SYMBOLS
= VCA or Multiplier
= Switch or Multiplier
= Panel Switch
= Panel Knob
= Panel Knob
Line indicates that controlled parameter is a potential destination for matrix patches.
OB•Mx
Voice Architecture
D. Buchla March, 1994

MATRIX SOURCES AND DESTINATIONS

Patch	Source	Module	Param	Mult
1-12	Pressure	VCO 1	Pitch	-1.00 to
	Velocity	VCO 2	Portamento	+1.00
	Note #		PulseWidth	
	ModWheel		VCO2->VCO1	
	PitchWhl		Env 1 -> Pitch	
	Breath		LFO 1 -> Pitch	
	Pedal			
	Volume	MM VCF	VCO1 In	
	MIDI A	OB VCF	VCO 2 In	
	MIDI B		Noise In	
	MIDI C		Frequency	
	MIDI D		Resonance	
	LFO 1		Env 1 -> Freq	
	LFO2		LFO 1 -> Freq	
	LFO 3			
	EnvGen 1	MIX	Lopass	
	EnvGen 2		Bandpass	
	EnvGen 3		Highpass	
	EnvGen 4		Highpass	
	Random		Pan	
			VCA Offset	
		LFO 1	Delay	
		LFO 2	Rate	
		LFO 3	S and H	
		ENV 1	AttackTime	
		ENV 2	AtackDelay	
		ENV 3	DecayTime	
		ENV 4	DecayDelay	
			Sustain	
			Release	

OB-MX Front Panel Instant Index



Turn to the page numbers shown for the best descriptions of the items' functions