

## Chapter 11

# Master Mode

Press the Master mode button to enter Master mode, which contains parameters affecting the K2500's overall performance.

### The Master Mode Page

On the Master mode page you'll find parameters for setting the overall tuning and transposition of the K2500, the MIDI channel to be used for drum programs (explained below), and for several keyboard, audio, display, and programming adjustments. You can also enter the sampler from the Master mode page

```

MasterMode: Samples:8192K  Memory:121K
Tune       :0ct
Transpose  :0ST
DrumChan   :1
VelTouch   :1 Linear
PressTouch :1 Linear
Intonation :1 Equal
Object     Delete  Util  Sample View  Reset

```

PARAMETER	RANGE OF VALUES	DEFAULT
TUNE	± 100 cents	0
TRANPOSE	± 60 semitones	0
DRUM CHANNEL	1 to 16	1
VELOCITY TOUCH	Velocity Map list	1 Linear
PRESSURE TOUCH	Pressure Map list	1 Linear
INTONATION	Intonation Table list	1 Equal
OUT A PAN MODE	Mono, Stereo	Stereo
OUT B PAN MODE	Mono, Stereo	Stereo
OUT A EFFECTS POSITION	L Only, L + R	L + R

#### Tune

Adjusting the value of this parameter will tune every program in the K2500 by the amount you specify. Tuning can be adjusted up or down 100 cents (one semitone) in one-cent increments. This parameter is useful for tuning up with recordings and acoustic instruments. Adjusting the tuning in Master mode does not change the settings on the PITCH page of individual programs, but will be added to any adjustments you make there. Master mode tuning adjustments affect only the K2500's notes.

#### Transpose

Like the Tune parameter above, Transpose will affect every K2500 program, but not those notes sent to the MIDI Out port. You can adjust the MIDI transposition on the XMIT page in MIDI mode.

#### Drum Channel

Most K2500 programs have a maximum of three layers, which is enough to create just about any sound you can think of, especially when that sound covers most or all of the keyboard. But what do you do if you want to create a program with a dozen or more different sounds across the range from C 0 to C 8, and you want to be able to run each sound independently through a set of DSP functions?

Drum programs allow you to do exactly that. They can have up to 32 layers, and each layer has all the programming features of regular programs—its own keymap, its own algorithm, etc. We call

them drum programs because a likely application for this type of program is to create drum kits with many different percussion timbres. You can assign any keymap to any layer, however, to create as complex a program as you want. All those extra layers create a lot of information for the K2500's sound engine to process. Every bit of information that's sent to the sound engine is associated with one of the MIDI channels. The K2500's sound engine has enough processing power to handle three layers' worth of information from 8 channels, and 32 layers' worth from the other 8. By default, MIDI channels 1-8 are defined as drum channels.

If you'd like, you can use the DrumChan parameter to define a channel numbered higher than 8 to be a drum channel. In this case, the channel you choose, along with channels 1-7 will be your drum channels. For example, if you set DrumChan to 13, then your eight drum channels will be 1, 2, 3, 4, 5, 6, 7, and 13. If you set DrumChan to a value of 8 or less, however, your drum channels will always be 1, 2, 3, 4, 5, 6, 7, and 8.

Drum channels are the only channels that can have drum programs (any program with more than three layers) assigned to them. So when you're in Program mode, the current MIDI channel must be a drum channel if you want to assign a drum program to it. If you're in Program mode (or in Quick Access mode with a program selected) and the current MIDI channel isn't a drum channel, the drum program's name will be in parentheses and the box at the left of the Program mode display will tell you which channels are the drum channels. If you're in Setup mode (or in Quick Access mode with a setup selected) and the current MIDI channel isn't a drum channel, the drum program's name will be enclosed in parentheses. In either case, the program will not play. Setting the DrumChan parameter to match the current MIDI channel (or vice versa) will restore the drum program to normal operation.

### **Velocity and Pressure Touch**

These two parameters are only used by the keyboard version of the K2500, and have no function in the K2500R. If you are using a K2500R, you should leave them set to their default values.

If you change the setting of the Veltouch parameter, remember that it also has an effect on the transmit velocity map (set on the MIDI XMIT page).

### **Intonation**

Most modern western music uses what is known as equal temperament. This means that the interval between each semitone of the 12-tone octave is precisely the same as every other interval. Many different intonation intervals have evolved over the centuries, however, and the K2500 supplies you with 17 different intonation "tables" to choose from. (There's also an eighteenth "table" listed, which we'll describe in a moment.) By changing the value for this parameter, you select from among the intonation tables stored in the K2500's memory. Each of these tables defines different intervals between each of the semitones in a single octave.

Scroll through the list of Intonation tables, and listen for the differences between semitones. Some of the intervals between semitones may be quite different from equal intonation, but you'll notice that all notes are precisely tuned with notes that are an octave apart. This is because the intonation tables set the intervals within a single octave, and apply those intervals to each octave. If this doesn't make sense, the explanation of the Intonation Table Editor, in Chapter 17, will help clarify things. If you're hoping to create fully microtonal tunings by editing intonation tables—sorry, that's not possible. But you *can* create microtonal tunings using the Keymap Editor; see Chapter 15. There's a list of the intonation tables in Chapter 17; descriptions of each can be found in Chapter 5 of the *Reference Guide*.

### **Determining the version number of your ROM Objects (Intonation Table 18)**

As you're scrolling through the list of intonation tables, you may notice a listing for an eighteenth intonation table with a name such as "18 Obj vn.n". This isn't really another intonation table. Rather, this is where the K2500 stores the version number of your ROM objects. If you ever need to find out what version of ROM objects you've got loaded, this is where you look. Simply go to the Master page, then scroll the Intonation parameter until 18 is displayed. And don't forget to return to your correct intonation table when you've checked the version number of your ROM objects.

## Outs A and B—Pan Mode

The parameters OutA->Mix and OutB->Mix determine the panning of the audio signal for output groups A and B at the MIX outputs. If they're set to Stereo, then whatever panning is applied to each program (as set on the OUTPUT page in the Program Editor) will show up at the MIX outputs. If either of these parameters is set to Mono, then programs routed to the corresponding audio output group will send all of their sound equally to both MIX outputs, regardless of any panning that might be applied to them on the OUTPUT page. This is a quick and convenient way to convert the MIX outputs into two mono outputs. Output groups C and D are permanently set to Stereo, and cannot be changed.

## Out A Effects Position (Out A->FX)

This parameter is intended for special cases when you want an additional separate output that doesn't go through the K2500's effects processor. A value of L Only bypasses the effects processor for all sounds assigned to the right side of Output Group A.

There's more than one way to set this up; we'll give you one possible example that should get you started. You're going to play a multi-timbral sequence. You want to send your percussion sounds through an external effects box. You want your lead sound to go through an external fuzz box. You want the rhythm section to use the K2500's internal effects. Finally, you want to send all the sounds through the K2500's MIX outputs so you'll use only two inputs on your mixing board. This is done by using stereo insert cables in the separate outputs, which enables you to loop sounds through external devices, back to the K2500, and through the MIX outputs. See Chapter 18 for more information about using stereo insert cables.

The percussion sounds are routed to the B outputs (this is done on the OUTPUT page in the Program Editor). You connect the stereo end of an insert cable in each of the K2500's B outputs. The send sides (tips carry signal) of the insert cables are connected to the inputs of your effects box. The return sides (ring receives signal) are connected to the return jacks on the effects box. The percussion sounds will now appear at the K2500's MIX outputs without going through the internal effects processor (sounds routed to Output Group B never go through the internal effects).

All the other sounds are routed to Output Group A. The lead sound is panned fully right, and the rhythm sounds are panned fully left. You connect an insert cable to the A Right output. The send side of the insert cable goes to the input of your fuzz box, and the return side goes to the fuzz box's return or output jack. The lead sound will now appear at the right side of the MIX outputs. The rhythm sounds will appear at the left side of the MIX outputs. At this point, the K2500's internal effects will be applied to the lead sound as well as to the rhythm sounds. You don't want the internal effects applied to the lead sound, so you set the value of the Out A->FX parameter to L Only. Now the lead sound will be dry and the rhythm sounds will be wet.

You want the lead and rhythm sounds to be centered in the mix, so you set the Out A->Mix parameter to Mono. This sends both sounds at equal levels to both the MIX outputs, but the lead sound is still dry.

## Contrast

Use this parameter to adjust the contrast of the display, to adapt to different lighting conditions. The K2500 will remember your setting even when you power down.

## Confirm

Confirmations are special displays that the K2500 shows you when you are about to alter memory permanently. The confirmations ask if you really want to do what you're about to do, and give you another chance to cancel the operation you're about to execute. With the Confirm parameter set to Off, these prompts do not appear. You'll still be alerted before doing something that might cause you to lose your work.

### Intonation Key (Intona Key)

This sets the tonic, or base note from which the currently selected intonation table calculates its intervals. If you select G as the intonation key, for example, and the intonation table you select tunes the minor 2nd down by 50 cents, then G# will be a quartertone flat relative to equal intonation. If you change the intonation key to D, then D# will be a quartertone flat. If you use non-standard intonations, you'll want to change the intonation key as you change the key you're playing in.

You can also set the intonation key from an external MIDI device. Note On events at C -1 through B -1 (MIDI note numbers 0 through 11) will set the intonation key at C through B, respectively.

To trigger notes in the range required to set the Intonation key, you can transpose the K2500 temporarily from its front panel, or from your MIDI controller if it has the ability. Alternatively, you could create a setup with just the lowest octave transposed down two octaves, then select it when you want to change the Intonation key. If you're driving your K2500 from a sequencer, you could simply insert the appropriate note events anywhere in the sequence to change the intonation key.

## The Soft Buttons in Master Mode

### Object

This soft button brings up the object utilities. They're described beginning on page 11-6.

### Delete

This soft button brings up the file dialog, enabling you to erase sets of objects, either entire banks or all objects, from RAM. If the Confirm parameter on the Master mode page is set to a value of On, you'll be given an extra chance to cancel before the set of objects is actually deleted. Once the deletion is complete, the objects are irretrievable, so you may want to save objects to disk before deleting them from RAM.

To delete individual objects, use the functions that are available when you press the **Object** soft button.

### Util

With this button you call up the Utility page, which gives you access to four analytic and diagnostic tools. Double pressing the two center soft buttons from any editor is another way to get to the Utility page. The Utility page looks like this:

Select what to display:

**MIDI** **Objects** **Voices** **Stealer**

**Done**

The **MIDI** soft button launches MIDIScope™, a useful subprogram that lets you monitor the MIDI messages from the K2500's keyboard and those received via MIDI. This is a good way to make sure you're receiving MIDI from MIDI masters. It's also good for making sure your controls are assigned where you want them, checking your attack velocities, etc.

The **Objects** soft button displays the entire list of objects stored in RAM. This is an easy way to check the object ID of any object you've created. You cannot manipulate objects, though, as you can with the Objects Utility (see page 11-6).

When you press the **Voices** soft button, the display shows the K2500's active voice channels as you play. Blocks of capital Xs in six columns of eight represent the 48 notes that the K2500 can play simultaneously. The Xs change to lower case xs, then to commas and periods, then finally drop out as each voice releases or decays to silence. This feature gives you an indication of the envelope level of each voice, though not necessarily the volume level. Nonetheless, this can give you a valuable

indication of how your voices are being used. For example, if all or most of the voices are being represented by capital Xs on this screen, then there's a good chance that when voice stealing takes place an audible voice will be reallocated.

Use the **Stealer** soft button to select a display that will show how the K2500 is allocating its 48 voice channels. When you trigger a note, the note number will appear in one of the display's three columns, and will remain visible while the note is sustained. The four-digit numeral you see is an internal value that has no direct significance. As long as fewer than 48 voice channels are being used, new note numbers will appear as you play additional notes, and the note numbers for notes that have decayed or have been released will disappear. When all 48 voices have been activated, the display will show which voice channels are shut off ("stolen") to enable new notes to play.

Press the **Done** soft button when you are finished with the Utility page. This is the same as pressing the Exit button.

### View

Press the **View** soft button to see the current program in large type. Press it again to return to the normal view.

### Sample

Press the **Sample** soft button to enter the K2500's sampler. Refer to Chapter 15 for complete information on the sampler.

### Reset

Press the **Reset** soft button if you want to return your K2500's memory to the state it was in when you bought it.

**Caution!** The K2500 will ask you if you want to delete everything (meaning all RAM objects), and a pair of **Yes/No** soft buttons will appear. Press **No** if you want to keep any objects you may not have saved. Press **Yes**, and everything stored in RAM will be erased. All parameters will be restored to default values. After a few seconds, the K2500 will return to the Program mode page.

## Guitar/Wind Controller Mode

If you are using a wind controller or guitar controller with your K2500, you may not always get the sound you expect. Since these controllers will sometimes send a MIDI Note On command before sending Breath or Volume data, the attack transients that characterize each instrument may be lost.

Therefore, the K2500 provides a special mode that may improve its response to your guitar or wind controller. To enter Guitar/Wind Controller mode, press both Chan/Bank buttons while in Master mode, then confirm with the **Yes** soft button.

Enable Guitar/Wind controller mode?

Yes No

Guitar/Wind Controller mode slightly delays MIDI Note On and Note Off commands, so that response to pitch bend and other expressive components of a note will be more accurate. If you are hearing a "glitch" in the attack of notes from your guitar or wind controller, you should try setting your K2500 to this mode. Keep in mind, however, that since this mode slightly changes the order in which MIDI commands are sent, it may affect the performance of the K2500 under some circumstances.

When you reboot the K2500, it will reset Guitar/Wind Controller Mode to off.

## Object Utilities

Object Utility functions are useful for moving or copying objects into various banks, naming or renaming objects, deleting objects, and dumping objects over MIDI. To access these functions, press the **Object** soft button while in Master mode. You will see the following screen:

Select database function:

**Move** **Copy** **Name** **Delete** **Dump** **Done**

The soft buttons are used to choose the various object utility functions.

**Move** Move selected objects to a new bank or a specific starting ID.

**Copy** Copy selected objects to a new bank or a specific starting ID.

**Name** Name selected objects.

**Delete** Delete selected objects.

**Dump** Dump selected objects over MIDI.

**Done** Exit from the object utilities.

Each function's multiple object selection interface is identical to the one used in the Save Objects dialog. For a complete description of this interface, see the "Saving Individual Objects" section on page 13-24. Here is what the **Move** object screen would look like (with several objects selected):

```
Func: MOVE      Sel: 14/211      Index: 1

Sample  200*Zild  20  Ride Hrd  C  4S  260K
Sample  201*Zild  20  Ride Sft  C  4S  224K
Sample  202*Zild  16  Crash   C  4S  341K
Sample  203*Zild  NewBeatOpen C  4S  198K
Select  Next    Type  Multi  OK   Cancel
```

The name of the function is displayed on the top line:

```
Func: MOVE
```

If you press **Cancel** while in one of the object utilities, you are returned back to the Object Utilities page (the "Select database function:" screen pictured above.) Any objects that were selected when you pressed Cancel will still be selected if you subsequently enter a different object utility (by pressing a different soft button such as **Name**, for instance.) The selections are reset when you exit the Object Utilities page (by pressing the **Done** button.)

All of the features of the Save Object dialog are accessible here:

**You can use the Multiple Object Selector** (described on page 13-32) to select ranges of objects according to object types, IDs, strings in the object names, or dependent relationships.

**You can quickly select or deselect all objects** using the Left/Right cursor and Up/Down cursor double-presses.

**You can audition any of the Program, Keymap, Sample, and Song objects** by pressing either the Left or Right cursor arrow button, when the desired object is highlighted. Songs will play until either cursor is pressed again. To audition a Program, Keymap, or Sample object, play a note on your MIDI controller (after pressing the Left or Right cursor button). (K2500R users also have the shortcut of holding the CANCEL button, then pressing the numeric keys to play notes.)

## Move

Pressing **Move** from the Object Utility page takes you to the Move utility. The Move utility allows you to select any group of objects and move them to a different bank. If you select several objects of a single type, then you are allowed to set a specific starting ID for the objects, of any number (0-999). For instance, you could move a group of samples from scattered IDs to a continuous range of IDs starting from ID# 354.

If you move objects that are dependent objects of other objects (such as samples that are dependents of a particular keymaps,) the parent objects are automatically relinked to their dependents that have moved. What this means is that you can move any objects to any new ID numbers, without having to worry if your programs, keymaps or songs will still play correctly.

### Moving a Single Object

If you select a single object to move, you will see a dialog similar to the Save/Replace dialog you see when saving any object.

```
Move Awesome Click to: ID#138
(rePlace Click)
```

```
RePlace Cancel
```

### Moving Multiple Objects

Select the objects that you want to move, then press **OK**. You will see the following dialog:

```
Move to bank: 200...299
               300...399
               400...499
               500...599
Append Fill Cancel
```

This is similar to the Load function, where you are asked to choose a bank and mode for loading. If you have selected more than one object, and all of the selected objects have the same type, then there will be an additional mode available, "ID":

**ID   Append   Fill   Cancel**

Here is a description of the above soft buttons:

- Append**     Try to use each object's ID offset within its current bank as the ID offset within the specified destination bank. If there is already an object at this offset in the new bank, increment the ID until a free ID slot is found. For example, if you were moving Programs 202, 209, 217, and 230 to the 400's bank, which already has a program at 409, the resulting IDs for the moved programs would be 402, 410, 417, and 430.
- Fill**         Use consecutive numbering for each object that is moved, starting from the beginning of the specified destination bank. Any object IDs that are already being used in the specified bank will be skipped over. For instance, if you were moving Songs 300, 315, 489, and 841 to the 200's bank, which already contains Songs 200 and 203, then the moved songs' ID numbers would be 201, 202, 204, and 205.
- ID**            Use consecutive numbering for each object that is moved, skipping over IDs that are in use (like Fill mode), starting from the ID that you specify. You will see the following dialog when you press the **ID** soft button:

Select starting ID: **231**

**OK   Cancel**

**Cancel**        Return to the Move object dialog.

The objects are moved as soon as Append or Fill is pressed, or when a starting ID is selected. After the Move function completes, you will still be in the Move object dialog, and you will still be scrolled to the previously highlighted object, even if it has moved to a new ID.

Keep in mind that when you only select one object to move, you can replace another object. However, when you select multiple objects for moving you cannot overwrite any objects.

## Copy

Pressing **Copy** from the Object Utility page takes you to the Copy utility. The Copy utility allows you to select any group of objects and copy them to a different bank. Only object data is copied, and not sample data. If you copy a sample object, you will end up with a "copy sample" that points to the same region of sample RAM as the original.

The operation of the Copy utility is identical to the Move utility just described.

If the objects to be copied in a single operation include any objects grouped together with any of their dependents, the new copies of the parent objects will reference the new copies of the dependent objects. As an example, suppose you select Song 400 and its three dependent programs, Program 200, 210, and 303. If you copy all of these objects at once into the 700's bank, using **Fill** mode, you will see the copies at Song 700 and Program 700, 701, and 702. Song 700 will reference the copies of the programs (at 700, 701, and 702.) In contrast, if you had only made a copy of Song 400 as Song 700, the song would reference the old programs (at 200, 210, and 303.)



## Name

The **Name** utility allows you to rename an object without entering an editor. You can also use this utility to rename one or more objects with the same name. This is much faster than renaming each object individually. A lot of times you might want several objects to have the same name except for a unique identifier at the end of the name. Using this utility function, you could assign a common name to multiple objects at once, and then quickly name each one a little differently.

When you press **OK** after selecting objects, you will see the following screen prompting you for an object name, with a suggested default:

Object Name:      Zither

Delete Insert <<< >>> OK Cancel

The default name that you are presented with is taken from the **highlighted** object in the object list, whether the highlighted object is selected or not. This makes it easy to copy the name of one object on to another.

As in all naming dialogs on the K2500, you can do a double-press of the Left/Right cursor buttons to put the naming cursor on the last character of the string. This is helpful when putting unique characters at the ends of names.

### Left/Right cursor button double-press -> Move cursor to the end of the name

The Relink-by-Name feature (described earlier) relies on there being unique names for dependent objects of the same type, so it is a good practice to make object names unique, particularly samples.

## Delete

The Delete Objects utility is very useful for reclaiming unused object and sample RAM in your K2500. This utility allows you to select any arbitrary group of objects for deleting, and audition them if necessary before getting rid of them. This can be a convenient way to delete individual or selected groups of objects. However, if you want to delete an entire bank or everything in RAM, the **Delete** soft button on the Master Mode page is quicker.

If any of the selected objects have dependents that were not selected, you will see the question:

Delete dependent objects?

Yes No

If you answer **Yes** to this question, all dependent objects of the selected objects are deleted, unless they are being used as dependents of other objects that are to remain in memory.

Answering **No** will delete only those objects that were selected.

## Dump

This utility is for dumping selected objects over MIDI. If any of the selected objects have dependents that were not selected, you will be asked the question "**Dump dependent objects?**".

Press **OK** to initiate a MIDI System Exclusive dump of the selected set of objects, one by one out the MIDI Out port of the K2500. Dumping everything can generate massive dumps, so you should know the limits of the device you're dumping to. You can cancel the dump at any time with the **Cancel** soft button.

Note that only sample objects (which contain the Start, Alt, Loop, and End points, as well as the values of all parameters found on the MISC page in the Sample Editor) are dumped by this utility, and not RAM sample data. (RAM sample data can be dumped via the MIDI Sample Dump Standard from within the EditSample page. See Chapter 10 of the *Reference Guide* for more information on the MIDI Sample Dump Standard.) Dumping the sample object of a RAM sample is not very useful because the sample memory address ranges are fixed in the object. This means that if you load the sample object back into the K2500 via MIDI, there is virtually no way it will point to and play back the same area of sample memory as when it was dumped, let alone the same sample data. However, this can lead to some interesting results.

Sample objects that reference the K2500's ROM sample area will reference the same area when you load them back in via MIDI.

## Accessing the Object Utilities from the Editor

The object utilities can also be accessed while editing any object. This is provided as a convenience, for example to be able to do certain housekeeping work such as deleting samples to free up room in your sample RAM, or making copies of objects. Access to the utilities can be done by pressing the **Object** soft button from any Save/Replace dialog in the editor:

Save Train Wreck as: **UU#412**  
(rePlace Train Wreck)

**Object**

**Rename RePlace Cancel**

You can get to this Save/Replace dialog when editing an object by either exiting after you have modified the object, or pressing the **Save** soft button.

If you try to use the Copy utility to copy the exact object you are in the process of editing, you will make a copy of the edited version. With sample objects this would be one way to save off a copy sample that references a small part of a much larger sample. You could remain in the sample editor, and continue to edit the larger sample, by pressing **Done** followed by **Cancel** after making the copy. This may be a faster way to save many "snippets" out of a sample than continually re-entering the sample editor after saving copy samples to different IDs.

When using the Object Utilities from within the editor, you must be careful not to delete any of the objects you are currently editing. This could have unpredictable results.

## Chapter 12

# Song Mode

### Getting Started with the Sequencer

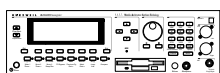
The K2500's sequencer is a powerful and versatile tool for songwriters, composers, and anyone else who needs to record and play back songs. As with any tool, however, it's best to start with the basics. This section begins with a tutorial where you will record a song, then shows some of the mixing capabilities of the sequencer. If you are familiar with other sequencers, you will have no problem using Song mode in the K2500. Read through this section, however, to learn about the features that make the K2500's sequencer unique.

#### What is a Sequencer?

A sequencer is similar in some ways to a multi-track tape recorder: you can record and play back all sorts of music and sounds, layer sounds on top of other sounds, and change or manipulate things that you've previously recorded. Unlike a tape recorder, however, you do not actually record sounds with a sequencer. Rather, you are recording commands that cause sounds to be played. Nonetheless, we will sometimes explain sequencer features by drawing analogies to familiar tape recording techniques such as splicing and overdubbing.

There are several advantages to recording a song by sequencing. For one thing, sequencer commands take up much less disk space than digitally recorded music would, so you can get a lot of information (i.e., music) on a single floppy disk. Furthermore, you can easily make changes to your sequences. For example, you can change individual notes, transpose parts, or change instrumentation. Lastly, you can share the sequences you create with other musicians.

#### A Word about the Local Keyboard Channel



Before you begin sequencing, we'd like to remind you about the Local Keyboard Channel parameter on the MIDI Receive page (described on page 10-6). Local Keyboard Channel is especially important for sequencing with the rack mountable K2500R, since it enables you to record on different tracks without constantly switching transmit channels on your controller. Therefore, you should do the following before you begin sequencing with the K2500R (or any K2500 model, when using an external controller):

- Set the Local Keyboard Channel on the MIDI Receive page to a specific channel (1-16).
- Set your controller (e.g., keyboard) to transmit on the same channel.

Performing the above two steps means that you'll be able to hear the individual channels (each of which is assigned by default to a separate record track) as you scroll through the different record tracks in the K2500's Song mode. Local Keyboard Channel performs a "re-channelizing" function that makes this happen.

#### Patch Through

There's one more use for Local Keyboard Channel. With any model of the K2500, the Local Keyboard Channel parameter lets you "patch through" (also known as "soft through") to external sound modules. When Local Keyboard Channel is enabled, the K2500 takes the rechannelized information and sends it out the MIDI port. This lets you hear an external module while you are recording a track assigned to that module.

## Tutorial: Recording a song

In this tutorial, we'll record a song by using the steps described below. Bear in mind, though, that this is just one approach to sequencing a song. This example includes:

- Assigning programs to channels
- Recording a drum loop
- "Unlooping" the drum track and adding the remaining instruments
- Mixing the song

### Assign Programs to Channels

Start by deciding what instruments you want to use in the song. Suppose you want to record a bass / drums / organ rhythm track with a lead instrument on top. You've decided to use:

- Jazz Kit (Program No. 52)
- Mix Bass (Program No. 80)
- Gospel Organ (Program No. 97)
- Brt Saxy Lead (Program No. 46)

Set up your K2500 so that each of these instruments is on a separate MIDI channel. Since Song mode automatically assigns each channel to a separate sequencer track (1-16, consecutively), you'll then be all set when you start laying down tracks, and won't have to go scrolling through the program list. Don't worry about changing your mind later, though, since you can always make changes after you've recorded your initial tracks.

It will be most logical to assign Jazz Kit to one of the drum channels, such as channel 1. If you want to put the drums on a channel that isn't a drum channel, though, you'll have to change the drum channel on the Master page (see page 11-1). For the sake of simplicity, however, we'll put the drums on channel 1 and assign Mix Bass, Gospel Organ, and Brt Saxy Lead to channels 2, 3, and 4.

Follow these steps to assign the programs to separate channels:

1. **Press the Song mode button to enter Song mode. The screen will look something like this:**

Record track is set to track 1

```

SongMode:MAIN  Events:186K  STOPPED
CurSong:1 NewSong  Tempo:120
RecTrk :1  Vol:127 Pan:64  Mode :Merge
Program:1 Acoustic Piano  Locat: 1:1

Track :R - - - - -
Channel:1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Record Play Stop Erase MISC MIXER

```

Notice that the sequencer is ready for you to record a new song, and the record track (RecTrk) is set to track 1. If "1 NewSong" doesn't appear in the CurSong field, press the MISC soft button, then press the New soft button on the MISC page. You will be returned to the MAIN page, and CurSong will say "1 NewSong."

2. Use the down arrow button to move the cursor to the Program field.

When this field is highlighted, type "52" then press the ENTER button. You've now assigned "Jazz Kit" to channel 1.

```

SongMode:MAIN  Events:188K  STOPPED
CurSong:1  NewSong          Tempo:120
RecTrk :1      Vol:127  Pan:64  Mode :Merge
Program:52 Jazz Kit          Locat: 1:1

Track  :R - - - - -
Channel:1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Record  Play  Stop  Erase  MISC  MIXER

```



**NOTE:** If "52 Jazz Kit" appears in parentheses, it means that the current channel is not a drum channel. You can either go to the Master page to change the drum channel, or assign the drums to a current drum channel in Song mode.

3. Now use the up arrow button to highlight the RecTrk field.

Change RecTrk to "2". You can use either the alpha wheel or the keypad.

4. Return to the Program field, and set it to "80 Mix Bass".

```

SongMode:MAIN  Events:188K  STOPPED
CurSong:1  NewSong          Tempo:120
RecTrk :2      Vol:127  Pan:64  Mode :Merge
Program:80 Mix Bass          Locat: 1:1

Track  :- R - - - - -
Channel:1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Record  Play  Stop  Erase  MISC  MIXER

```

Notice that the "R" on the Track line, which stands for "Record", has moved to track 2. Also notice how each of the sixteen tracks has a default channel associated with it. You could change this if you wanted to, but most people find it easiest to associate track 1 with channel 1, track 2 with channel 2, etc.

5. Repeat the above two steps to assign "97 Gospel Organ" to channel 3 and "46 Brt Saxy Lead" to channel 4.

You've now chosen the programs for your first sequence. It's important to realize, though, that you have not recorded them yet. They will be there when you need them, but they have not yet been included in a song. Also, don't forget that you can change the program assignments any time before or after you record the song.

## Record a Drum Loop

Our song will be based around a four measure drum loop that we'll record now. Then, a little later on, we'll "unloop" the drum track and really start jamming.

The length of the drum loop is determined by the current *endpoint*, so we'll start by recording 4 measures of silence to set the endpoint.

1. Set RecTrk to "1" then press the Record soft button. The Song Status indicator (top right hand corner of the display) flashes "REC READY".

song status indicator

```

SongMode:MAIN Events:188K REC READY
CurSong:1 NewSong          Tempo:120
RecTrk :1 Vol:127 Pan:64 Mode :Merge
Program:52 Jazz Kit Locat: 1:1

Track  :R - - - - -
Channel:1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Record  Play  Stop  Erase  MISC  MIXER
  
```

Locat parameter  
shows current measure  
and beat number

2. Press the Play soft button. The Song Status indicator now reads "RECORDING".

The K2500's built-in metronome will begin clicking, and the Song mode LED will also flash at the current tempo.

Notice the "Locat" parameter on the right side of the display, which shows the current measure and beat number. When you begin recording, the K2500 will provide you with a four beat count-off, during which time Locat's measure and beat number will be preceded by a minus sign.

Since we are recording four measures of silence, press the Stop soft button as soon as Locat reads "4:4".



**NOTE:** The sequencer will truncate to the nearest downbeat, so as long as you press Stop before Locat reads "5:2" (but after it reads 4:4) you'll be alright. Don't worry about this too much, though, since in the next step we'll show you how to check (and change, if necessary) the endpoint.

When you press Stop, you'll be asked to confirm and name this song. Even though we've just recorded four measures of silence, when you save it and give it a name it's officially a song.

For the purposes of this example, we'll assume that you pressed Stop a few beats too late. You'll see how easy it is to correct this sort of thing in the Event Editor.

3. On the Song mode MAIN page, make sure that the cursor is highlighting any field other than the Program field, then press the EDIT button.

If the Program field is highlighted when you press EDIT, you'll enter the Program Editor, which is not what you want to do right now. The EditSong: COMMON page appears:

```

EditSong:COMMON <>Track:1
Tempo      :120      StartStep  :1
TimeSig    :4/4      TempoControl:Song
EffectChan :1        TriggerChan :1
ChainTo    :0 None   TriggerCtl  :ON
TrackDest  :-----
DrumTrack  :-----
<more> EVENT TRACK STEP ARRANG <more>
  
```

- Now press the EVENT soft button to bring up the Event Editor, which will look something like this:

```

EditSong:EVENT (Ch 1) <>track:1

1:1 1:1.000 CTRL BANK 0
1:1 1:1.000 PCHG 52
1:1 1:1.000 CTRL VOL 127
1:1 1:1.000 CTRL PAN 64
Cut Copy Paste View AllOn Done

```

As you can probably decipher, the Event Editor gives you access to an editable list of all note, controller, and other MIDI events that Song mode uses to describe your sequence. While you're looking at the Event Editor, notice the data that the sequencer keeps track of, even when no notes are played.

- Use the Alpha wheel to scroll to the bottom of the event display.

The last event listed is your endpoint, which should be the first beat of the measure following the last measure you want to record. For our four measures of silence, then, the endpoint should be 5:1. The display below, however, shows that five measures have been recorded:

```

EditSong:EVENT (Ch 1) <>track:1
1:1 1:1.000 CTRL VOL 127
1:1 1:1.000 CTRL PAN 64
6:1 6:1.000 END

Cut Copy Paste View AllOn Done

```

Fortunately, it's quite easy to change the endpoint from "6:1" to "5:1" to remove the extra measure that's been accidentally recorded. (If your endpoint is at 5:1, you won't need to change it now. If, however, your endpoint is less than or greater than 5:1, you should proceed with the next step.)

- Check the endpoint, and change it if necessary.

To change the endpoint from 6:1 (or any other incorrect value) to 5:1, press the right arrow button to position the cursor in the Bar:Beat:Tick column ("6:1.000" in the above example). Type "51000" then press the ENTER button. The endpoint is changed, and the song is now four measures long.

- Press EXIT to return to the SongMode:MAIN page.

If you've made changes in the Event Editor, confirm them now.

- Make sure that record mode and play mode are set to "Loop."

From the SongMode:MAIN page, press the MISC button. Set the RecMode parameter on the SongMode:MISC page to "Loop".

The PlayMode parameter should already be set to "Loop". If it isn't, turn the alpha wheel until "Loop" is highlighted.

## RecMode &amp; PlayMode parameters

## Quantize parameters

SongMode:MISC		Events:188K	STOPPED
RecMode : Loop	Quant:Off	CountOff:1	
PlayMode:Loop	Grid :1/8	Click :Rec	
KeyWait :Off	Swing:0%	ClickCh :16	
Locate : 1:1	Sync :Both	ClickPrg:198	
AutoIn : 1:1	Clock:Int	ClickKey:C 4	
AutoOut : 1:1	Tempo:Auto	ClickVel:90	
Record	Play	Stop	New
		In/Out	MAIN

## 9. Turn on Input Quantization.

While you're on the MISC page, take a look at the quantize parameters (Quant, Grid, and Swing) in the middle column of the page. Quantization is a very useful feature, especially for the rhythmically challenged. When you quantize a track, the sequencer moves the elements of that track closer to a grid based on the time signature of the tune. You can use quantization to subtly tighten up a rhythm track or to create a precise, unwavering mechanical rhythm.

For our drum loop, we'll try the total quantize experience, so position the cursor over the Quant parameter, then turn the alpha wheel until the value is set to "100%". Move the cursor down to the Grid parameter. The default value "1/8" indicates that quantization will move the notes you play to the closest eighth note. Try double-pressing the increment/decrement buttons below the alpha wheel to move through a range of useful grid values. Note that some of the values have "tr" or "t" appended to them. These are grid settings that allow you to maintain a triplet feel. We'll use a setting of "1/16", however, so set this value, then press EXIT to return to the SongMode:MAIN page.



**NOTE:** The K2500's sequencer also provides a full range of advanced quantization features that you can apply to previously recorded tracks. To learn about these, check out the *Quantize and Reference Quantize* functions on the EditSong:TRACK page.

## 10. Make sure you are in Merge Mode.

On the SongMode:MAIN page, check the Mode parameter, and make sure it is set to "Merge" rather than "Erase". Being in Merge Mode means that while you are recording this track you will be able to overdub more sounds with each successive loop.

## 11. Begin recording drums.

Press the Record soft button, observe the "REC READY" indicator on the top line, then press the Play soft button when you're ready to begin. Remember to wait for the four beat count off before you start to play.

Since you are in Merge Mode, you don't need to do everything at once. A common approach to making drum loops is to record a different instrument each time the loop comes around. For example, on the first loop you could record snare hits on the back beats (1:2, 1:4, 2:2, 2:4, etc.). Then you could add kick drum to the snare when the loop comes around again; you'll be able to hear the previously recorded part, as well as the new part. On the third pass you might record ride cymbal, followed by hi-hat or other percussive accents. Keep it simple at first, because you can always save the part while it's



basic (but correct), then make additions later. To keep track of where you are, watch the flashing Song mode LED or the Locat parameter on the SongMode:MAIN page.

**12. Press the Stop soft button when you've finished recording the drums.**

Confirm that you want to keep the changes to your song by pressing the Yes soft button followed by the Replace soft button.

### Record a Bass Line

When you are satisfied with your drum loop, you can begin using it as the foundation for a song. What we'll do here is set Record Mode to "unloop" while leaving Play mode set to loop. This means that the drum loop will keep playing while we record new material of any length. The endpoint of the song will change to reflect the length of the newly recorded material.

1. Press the MISC soft button to bring up the SongMode:MISC page.
2. Set the RecMode parameter to "Unloop". Leave the PlayMode parameter set to "Loop".

Depending on the type of song you are recording, you may also want to turn quantization off before you record your bass part.

```

SongMode:MISC  Events:186K  STOPPED
RecMode :UnLoop Quant:100%  CountOff:1
PlayMode:Loop  Grid :1/16   Click  :Rec
KeyWait  :Off    Swing:0%   ClickCh :16
Locate   : 1:1   Sync :Both  ClickPrg:198
AutoIn   : 1:1   Clock:Int  ClickKey:C 4
AutoOut  : 1:1   Tempo:Auto ClickVel:90
Record   Play   Stop   New   In/Out MAIN

```

3. Press the MAIN soft button to return to the SongMode:MAIN page.
4. Set the record track (RecTrk) to track 2.

This track already has material recorded on it.

```

SongMode:MAIN  Events:186K  STOPPED
CurSong:200*NewSong  Tempo:120
RecTrk :2    Vol:127 Pan:64  Mode :Merge
Program:80 Mix Bass   Locat: 1:1
□
Track  :P R - - - - - - - - - -
Channel:1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Record Play Stop Erase MISC MIXER

```

Track status indicators: track 1 is set to "Play", track 2 is set to "Record". Tracks 3 through 16 are empty.

Since you previously assigned "Mix Bass" to channel 2, it should appear in the Program field when you set track 2 as the record track. Note, too, that the track status indicator for

track 1 changes to "P" (for "Play") when you select track 2 for recording. The small square above the track status indicator tells us that material is contained on that track.

5. **Press the Record soft button to enter "REC READY" mode.**
6. **Press the Play soft button, then begin laying down a bass track.**

Remember that by default there is a four beat count off, during which time the Locat value will be preceded by a minus sign (-). No material is recorded during the count off, though anything you play during the countoff will be quantized to the first note of the song. As you are recording the bass track, your drum loop will keep playing. Play for as long as you want; the sequencer will lengthen the song as needed.

7. **Press the Stop soft button when you are done recording the bass.**

You will be given the usual save options. To keep what you've just recorded, press the Yes soft button followed by the Replace soft button.

Since you unlooped the drum track when you recorded the bass, you've changed the endpoint of the song to be wherever you stopped the bass track. You can check the endpoint (and change it, too, if you want) using the Event Editor, as described earlier.

### Record the Remaining Instruments in Your Song

Now that you've defined your song with the bass and drum tracks, you can put the organ and Brt Saxy Lead (or whatever instruments you've chosen) into your song.

1. **Set the record track (RecTrk) on the SongMode:MAIN page to track 3.**

Notice the small squares above the track status indicators for tracks 1 and 2, reminding you that you've now got material on two tracks.

```

SongMode:MAIN  Events:186K  STOPPED
CurSong:200*NewSong  Tempo:120
RecTrk :3  Vol:127 Pan:64  Mode :Merge
Program:97 Gospel Organ  Locat: 1:1
  □ □
Track  :P P R - - - - - - - - - -
Channel:1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Record  Play  Stop  Erase  MISC  MIXER

```

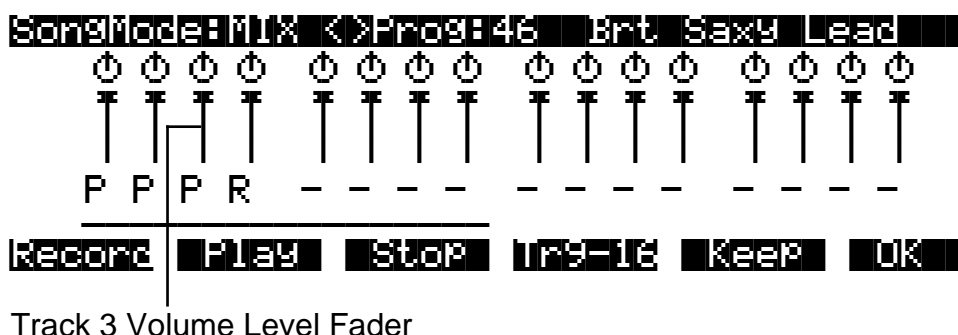
2. **Press the MISC soft button to bring up the SongMode:MISC page.**
3. **Set RecMode to "FixLen".**  
 Since you've defined the length of your song with the bass track, setting RecMode to "FixLen" means that the song will just play through once each time you record a new part.
4. **Record the organ in the same way that you recorded the bass track in the previous section.**  
 Notice that you can do this from the MISC page, without returning to the MAIN page.
5. **Continue recording instruments until you have played all the parts of your song.**

## Mix Your Song

The SongMode:MIX page lets you change the panning and volume levels for the tracks in your song. Needless to say, this is one of the most important steps in the completion of your song production, and potentially one of the most creative. This example will keep things simple by showing you how to change the volume level of one of your instruments. We'll also take a quick look at the track mute feature.

1. **Press the MIXER soft button to bring up the SongMode:MIX page.**

The icons that represent pan position knobs and volume level faders resemble the controls on a traditional mixing board. Manipulating them should be quite intuitive. Simply position the cursor over a pan position knob or volume level fader, then turn the alpha wheel to set the level you want.



For example, suppose you want to turn down the organ on track 3:

2. **Use the arrow keys to position the cursor over track 3's volume level fader on the SongMode:MIX page.**

Although the tracks aren't numbered on the SongMode:MIX page, they are laid out logically: left-to-right, from 1 through 16 consecutively. Track 3, then, is the third track from the left.

3. **Use the alpha wheel to turn down the volume of the track by changing the position of the volume slider.**
4. **Press the Keep soft button and confirm the change.**

Now when you play back the song, track 3's volume will start playing at the newly set level.

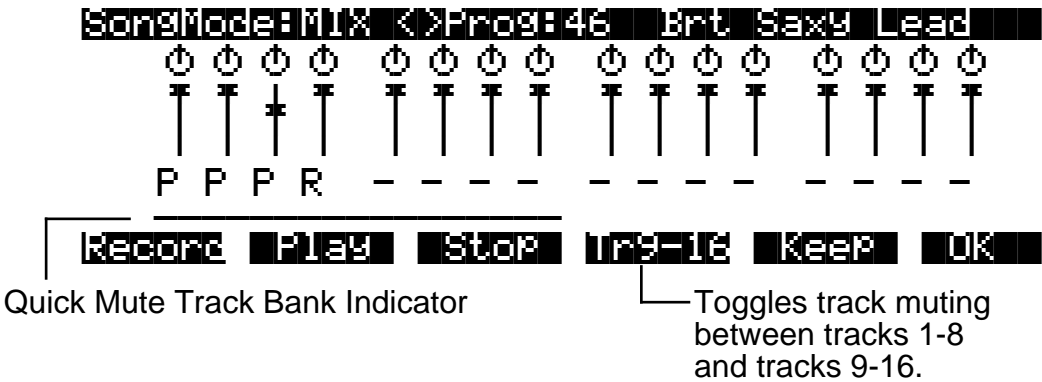
You can also record real-time volume and pan changes from the MIX page or enter numeric values for these parameters on the MAIN page.

## Using the Mode Buttons to Mute a Track

Finally, we'll take a quick look at the track mute feature, which lets you use the K2500's mode buttons to mute individual tracks. This can be invaluable during mixdown.

You may have noticed a horizontal line underneath the sliders for tracks 1 through 8 on the SongMode:MIX page. As the illustration below shows, this is the "Quick Mute Track Bank Indicator", showing which bank of eight tracks will respond to the eight track mute buttons on the front panel of the K2500. If you're looking for the track mute buttons on the front panel of your K2500, they're the eight mode buttons (Program, Setup, etc.); press one of these buttons

while you are on the SongMode:MIX page and playing a song, and the associated track will be muted.



Use the M 1-8 / M 9-16 soft button to toggle between banks of eight tracks, either 1 through 8 or 9 through 16. When you press this soft button, the horizontal bar will reposition itself below the affected tracks. The table below shows the K2500's mode buttons, and which tracks they will mute when you are playing a song:

Available Buttons	1-8	9-16
Program	1	9
Setup	2	10
Quick Access	3	11
Effects	4	12
MIDI	5	13
Master	6	14
Song	7	15
Disk	8	16

Let's listen to our song with and without drums:

- From the SongMode:MIX page, press the Play soft button.**  
The song, with all of its instruments, begins to play.
- Press the Program mode button on the front panel of the K2500.**  
The Program mode LED will light, and track 1 (the drum track) will be muted.
- Press the Program mode button again to unmute the drum track.**  
Using your K2500's front panel buttons, you can mute one or more tracks at once, or even mute eight tracks at a time.

With any luck, you've just completed your first song. You should now be ready to take on the more advanced features of K2500 Song Mode that are described in the remainder of this chapter.

## MAIN Page

The Song Mode MAIN Page allows real time recording and playback, song and track selection. From this page you can view and edit the tracks' channel, program, volume and pan settings, as well as other useful items.

```

SongMode:MAIN  Events:317K  STOPPED
CurSong:1  NewSong  Tempo:120
RecTrk :1  Vol:127  Pan:64  Mode :Merge
Program:2  Stereo Grand  Locat: 1:1

Track  :R - - - - -
Channel:1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Record  Play  Stop  Erase  MISC  MIXER

```

PARAMETER	RANGE OF VALUES	DEFAULT
CURRENT SONG	SONG ID & NAME	1 NEWSONG
RECORD TRACK	1 – 16, NONE, MULT	1
PROGRAM	PROGRAM ID & NAME	CURRENT PROGRAM
SETUP	SETUP ID & NAME	CURRENT SETUP
TRACK STATUS	-(EMPTY), R, M, P	-(EMPTY)
CHANNEL	1 – 16	1 – 16
VOLUME	0 – 127	127
PAN	0 – 127	64
TEMPO	1 – 255 BPM	120 BPM
MODE	MERGE, ERASE	MERGE
LOCATION	1:1 – 9999:9	1:1

The **Events** field on the top line displays the amount of free RAM space. 317 K in the above example represents maximum available memory with P/RAM installed. The maximum is 60 K without P/RAM. This field is replaced by the **Used** field, as shown below, when the Song Status is REC READY or RECORDING, to display the percentage of the record buffer (not the RAM memory) being used.

```

SongMode:MAIN  Used: 0%  REC READY

```

**Song Status**, also on the top line of the display, will be one of the following:

- **STOPPED** is displayed whenever the sequencer is not in another status, or when the **Stop** or **Pause** button is pressed.
- **PLAYING** is displayed when the **Play** button is pressed (assuming **Record** has not been previously pressed; and Key Wait is Off and the Clock set to Internal)
- **REC READY** is displayed when the **Record** button is pressed while the sequencer is STOPPED. It flashes, indicating that it is waiting to start recording.
- **RECORDING** is displayed when the **Play** button is pressed while REC READY is flashing (unless Key Wait is On or the Clock is set to Ext). RECORDING will also be displayed if the **Record** button is pressed while the status is PLAYING.
- **KEY WAIT** is displayed when the **Play** button is pressed, if the KeyWait parameter on the MISC page is set to On. It will go this status from both the STOPPED and REC READY status. It flashes, indicating it is waiting for a key to be struck to start recording or playing.

- **EXT. CLOCK** is displayed when the **Play** button is pressed, if the Clock parameter on the MISC page is set to Ext. It will go this status from both the STOPPED and REC READY status. It flashes, indicating it is waiting for an external MIDI clock message to start recording or playing.

### Current Song (CurSong)

Current Song ID# and 16 character name selected for recording, playback or editing. When a Song is selected, Program Change, Volume, and Pan information is sent to all MIDI channels assigned to tracks that have data on them, and the internal clock is set to the Tempo.

### Tempo

Controls tempo for the selected song. You can make temporary changes, record real time tempo changes, or set an initial tempo for the current Song.

Whatever the tempo is set to when you record your first track will be the Song's initial tempo. Temporary changes may be made during playback, but the tempo will reset to the initial tempo when the sequencer is STOPPED.

To change a Song's initial tempo, press **Record** (the Song Status will change to REC READY), set the tempo desired, then press **Stop**. The initial tempo can also be changed with the Tempo parameter in the EditSong : COMMON page. The Song will always start playback at the initial tempo, even though this tempo marker does not get recorded as a tempo event on any track.

If the sequencer is RECORDING, any tempo value changes will be recorded in real time. Unlike the special case of setting the initial tempo, any tempo changes recorded in real time are recorded as tempo events.

### Fractional Tempos

You can use fractional tempos (120.5, etc.) in your sequence. However, the initial tempo can not be fractional, and you cannot enter a fractional number in the tempo parameter on the MAIN or COMMON pages. You must first record a real time tempo event, then go to the EVENT editor and change it to a fractional amount.

To do this, press **Record**, then **Play**. The sequencer starts recording. Use any data entry method to choose a tempo. The value is unimportant since you will be changing it in the EVENT editor. Next press **Stop** and save the Song. Now when you go to the EVENT editor (see page 12-28), you will see a tempo event. You can now edit the value to a fractional amount. To have the song start immediately with the fractional tempo, edit its location to 1:1:000.

### Record Track (RecTrk)

Determines which track is record enabled. Set the record enabled track to Multi to record more than one channel simultaneously or to use a Setup in your Song.

When RecTrk is set to a single track ( 1 - 16 ), Record (R) is displayed for that track in the Track Status Indicator field (above the Track Channels). Conversely, with one exception, when any Track's Status Indicator is changed to Record (R), that Track is shown as the value for the RecTrk parameter.

The exception is when RecTrk is already set to Multi, you can select the record enabled tracks by toggling the Track Status Indicator to Record (R), and the RecTrk will remain set to Multi.

When Multi is initially selected, all of the empty tracks will be record enabled. Tracks containing data will remain set to play (P), but you can manually set them to record (R).

The parameter(s) below RecTrk will change according to the value of RecTrk and whether or not you select a setup. If RecTrk is set to a single track (1-16), Program is displayed and you can select the program to be assigned to that track. If you change RecTrk to Mult or None, the display changes to show the Channel parameter followed by the Program parameter (although the the word "Program" no longer appears). An example of this is shown in the screen below. If you switch through the channels, the program will also change, showing the program currently on that channel. Finally, if RecTrk is set to Mult, then if you go directly to Setup Mode and then back to Song mode, the parameter changes to Setup, allowing you to choose a setup with which to record.

### Program

Scroll through the Program objects in memory to select the Program before initially recording each track of your Song. Any MIDI program changes on the current RecTrk or Chan will cause the ID# and name of the track's program to change during playback.

This parameter's name, Program, is not present on the display when the RecTrk is set to None or Multi (to make room for the Chan parameter), but the value is still displayed.

The parameter is replaced with Setup when you select a Setup for playback or recording. The Setup parameter functions similarly to Program.

Programs selected in Program Mode or from a Quick Access bank will be selected as the program on the current RecTrk when you return to Song Mode.

```

SongMode:MAIN  Events:188K  STOPPED
CurSong:1 NewSong          Tempo:120
RecTrk :MULT Vol:127 Pan:64  Mode :Merge
Chan:4   2 Stereo Grand     Locat: 1:1

Track  :R R R R R R R R R R R R R R R R
Channel:1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Record  Play  Stop  Erase  MISC  MIXER

```

### Channel (Chan)

This parameter determines the control channel and is only available when the RecTrk is set to None or Mult. The Program parameter value remains on the display and can be edited, even though the parameter's name, "Program", no longer appears.

### Setup

Displays the ID# and name of Setup to be recorded. This parameter is available by setting RecTrk to Mult, then entering Setup mode momentarily by pressing the SETUP button followed by the SONG button to return to Song mode. The screen below shows an example of the Song mode page with the Setup parameter displayed.

```

SongMode:MAIN  Events:188K  STOPPED
CurSong:1 NewSong          Tempo:120
RecTrk :Mult Vol:127 Pan:64  Mode :Merge
Setup  :4 Total Orchestra   Locat: 1:1

Track  :R R R R R R R R R R R R R R R R
Channel:1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Record  Play  Stop  Erase  MISC  MIXER

```

The Chan parameter will be replaced with Setup. It is important to know how many channels, and consequently how many tracks, are needed for recording a particular Setup. Each Setup can have up to three Zones that can respond to your playing differently, depending on what range of the keyboard is being played, or if certain velocity and/or controller values determines when a particular Zone will respond. Be aware of the behavior of each Setup you intend to record so that you can allocate the proper tracks and channels needed in your Song.

### Volume (Vol)

You can set an initial volume level for the playback and recording of each track as a value between 0 and 127. If the channel of the RecTrk (or the control channel, if RecTrk is set to Multi or None) contains any recorded volume change (controller code 7), the change will be reflected as the Vol parameter's value in real time.

### Pan

You can set an initial pan position (the balance between the Left and Right audio channels) for the playback and recording of each track as a value between 0 and 127. A value of 64 is center. If the channel of the RecTrk or the control channel contains any panning data (controller code 10), the Pan parameter's value is modified in real time.

### Mode

If Mode is set to Merge you will be able to overdub when recording on a track containing previously recorded data. You'll usually want to set Mode to Merge when RecMode (on the MISC page) is set to Loop. Otherwise, each time through the loop, the previously recorded information will be erased.

If you set Mode to Erase, the previously recorded data on the record enabled track will be replaced with the new data only during the Bars and Beats you are actually recording, and the previously recorded data before and after the newly recorded Bars and Beats will be preserved.

### Location (Locat)

The Bar and Beat displayed as the Locate value changes relative to current location of the Song during playback and recording. You can set this to a negative Bar and Beat location to start playback a set length of time before the beginning of the Song.

Whenever you set the Locate point, that location will be used as the return point when **Stop** is pressed. Simply press **Stop** again to reset the Song to the top (1 : 1).

```

      □ □ □ ×
Track  : P P M R - - - - - - - - - -
Channel: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

```

### Mode Indicators (+ and x):

Mode Indicators only appear when there is pre-existing data on a track(s).

A plus sign (+) appears above the Track Status Indicator of a track set to record (R) when the Mode parameter is set to Merge.

An (x) appears above the Track Status Indicator of a track set to Record (R) when the Mode parameter is set to Erase.

### Activity Indicators ( □ )

A small square (□) above the Track Status Indicator of a track set to Play (P) or Mute (M) means there is data contained on that track.



During playback and recording, the indicators above tracks containing any MIDI data will flash a small, filled-in square when any MIDI activity is detected. The filled-in square also flashes over a track any time that there is incoming MIDI data on that track's channel, even while the sequencer is STOPPED.

### Track Status Indicators

Using the Up, Down, Left, and Right cursor arrows to position the cursor onto a Track Status Indicator, you can toggle an empty track (-) into Record (R) with the Alpha Wheel or Increment/Decrement buttons.

Once a track contains data, it will have a (P) as a Track Status Indicator, and it will be played during playback. You now will be able to toggle between Play (P), Mute (M), and Record (R).

The track selected as the RecTrk will have an (R) in this field, designating it as the record enabled track. If the RecTrk is set to Mult, initially all empty tracks will have Record (R) as a Track Status Indicator, any of which can be switched back to empty (-) if at any time recording on specific tracks is not desired.

If there isn't a track with an (R) in this field, the RecTrk parameter's value will be None. (The exception is when the RecTrk is set to Mult and you have switched all of the tracks out of record enable.)

### Track Channels

Each track has a MIDI Channel that it uses to receive and transmit data. By default, tracks 1 through 16 of a new Song are assigned to channels 1 through 16 respectively, although a track can play or record on any channel and the same channel can be assigned to more than one track. However, keep in mind that only one program can be assigned to a channel at a time, so if you have more than one track assigned to the same channel, they will be playing the same program.

## Soft Buttons on the MAIN Page

Although these soft buttons may look similar to the transport controls on a tape deck that might require that you hold down play and record simultaneously to begin recording, they do not operate in the same fashion. It is important that you only press one of these transport style soft buttons at a time to insure proper record start points, and to always be sure of the current sequencer status.

**Record** **Pause** **Stop** **Erase** **MISC** **MIXER**

**Record** changes the Song Status to REC READY if the current Song Status is STOPPED. If the current Song Status is PLAYING, it will be switched to RECORDING when you press **Record**.

If the RecTrk is set to None, pressing **Record** will select the first available empty track for recording, thus setting the RecTrk to the newly record-enabled track number and placing an (R) in that track's status display. Song Status will change to REC READY or RECORDING, depending on the previous Song Status as described above.

**Play** plays back any recorded data when pressed while the song status is STOPPED. Playback will begin from the bar and beat specified in the Locate parameter.

When the Song Status is REC READY, pressing the **Play** soft button will begin recording.

**Pause** and **Play** share the same soft button. **Pause** appears only when the Song Status is PLAYING or RECORDING. Pressing **Pause** while the song is playing will stop the playback (soft button switches to **Play**), and the location remains at the current bar and beat, allowing you to continue from that location by pressing **Play** again.

Pressing **Pause** while recording will stop the recording process as if you had pressed **Stop**.

---

**Stop** halts the playback or recording, and resets the Song's location to either the default "bar 1, beat 1" value, or to whatever location you defined with the Locate parameter. If the location is defined as something other than "bar 1, beat 1", press **Stop** twice to return to 1:1.

Pressing **Stop** when the Song Status is RECORDING will always prompt the "Save this song?" dialog (shown on page 12-16), and provides you with the opportunity to listen to the "New" Song and compare it with the "Old", previously saved, Song before answering Yes or No.

Here are some useful alternatives to using these button presses:

- Keyboard models have dedicated front panel buttons for **Record**, **Play/Pause**, and **Stop**. You'll find them just below the eight mode buttons.
- Footswitches can be assigned to **Play/Stop** and **Record/Stop**. On the MIDIMode : TRANSMIT page, assign either footswitch to controller number 102 or 103. Using controller 103, you can even record from within the Song editor.
- Double button presses allow Song **Play**, **Pause** and **Stop** soft button functions from most places where these buttons are not available. Use the cursor **Left+Right** double-press for **Play/Stop**, and use the cursor **Up+Down** double-press for **Play/Pause**.

**Erase** removes all channelized data from the track on which the cursor is currently positioned (on either the Track or Channel fields) or from the record enabled track if the cursor is positioned elsewhere. As shown on page 12-17, a dialog appears that allows you to verify your intentions before permanently erasing any data. Once you've confirmed the erasure, it takes place immediately, so be sure you really want to erase the track before pressing the **OK** soft button. You will not be able to revert to a version of the song that includes the erased tracks - even if you exit the song without saving edits. So before you go removing data that you may not want to permanently lose, you should also check out Edit Song: Track - Delete (page 12-36) and Edit Song: Track - Erase (page 12-34).

The Erase function will not prompt the dialog or erase any data if an empty track is selected.

**MISC** accesses more sequencer control parameters found on the SongMode : MISC page. These miscellaneous controls include record and play mode settings, auto punch-in points, quantization on input settings, tempo, and click attributes, as documented later in this chapter, starting on page 12-18.

**MIXER** lets you view a graphic representation of panpots and level faders for each track in the EditSong : MIX mode. You can modify program changes, volume and pan position in this Mode. Record status can also be selected here. The section on the Mixer begins on page 12-23.

## Save this song? Dialog

The following dialog page appears after you have recorded a track and pressed **Stop**, or if you have entered the Song editor and made changes, then pressed Exit, or if you press **Save** in the Song editor.

Save this song?

PlayOld PlayNew Stop

Yes No

**PlayOld** appears along with the **PlayNew** soft button in the "Save this song?" dialog after the recording process has been stopped. Pressing **PlayOld** will play the current song, minus the last, but not yet saved, recorded data. The exception is when there is no previously recorded data (i.e., recording the first track of a NewSong), then you will only have a **Play** soft button to audition the recorded data just entered.

**PlayNew / Play** soft buttons allow you to play all of the recorded data, including data on the track(s) you have just recorded.

**Stop** halts the playback of either the Old or the New version of the Song you are currently auditioning. Press the **Stop** soft button to stop the playback or recording, and reset the Song's location to either the default "bar 1, beat 1" value, or to whatever location you defined in the Locat parameter

**Yes** saves the data on the track(s) you just recorded. Whatever was played back when you pressed **PlayNew** will be the version of the Song to be saved when you press Yes. The "Save New Song?" Dialog (shown below) will be displayed.

**No** returns you to the Song mode page in which you were last recording, without saving any unsaved changes to the current Song.

## Save New song? Dialog

Save NewSong as: ID#200

**Object**

**Rename**

**Save**

**Cancel**

Saving a Song requires that you assign it an ID# and this is where you do so. If you select an ID# of an existing Song Object in RAM, the **Save** soft button will become **Replace** and "Replace existing Song" will appear on the display as a warning and a chance to change your mind.

There are only 20 ID#s available for Song Objects per bank in banks 100 - 900. In the zeros bank, there are 75, ID#s 1 - 75. In the 200's bank there are ID#s 200 - 219, in the 300's bank, ID#s 300 - 319, and so on.

**Object** accesses some useful database functions while still in Song mode, before you actually save the current Song. Pressing **Object** jumps you directly to the Object Utility, described on page 11-6. When you press **Done** on the Utility page, you will be returned to the "Save New Song" dialog.

The **Rename**, **Save**, and **Cancel** soft buttons function the same as they do in all other editors.

## Erase Track Dialog

This dialog asks you if you are sure you want to erase a specific track. The track number will correspond to the track currently selected by the cursor position. When RecTrk is set to Multi or None, the Erase track function looks to the cursor position to determine the track to erase. If the cursor is positioned somewhere other than the Track or Channel fields, Erase will have no effect. Erasing a track will not alter the Song's End point, nor will it remove any tempo events, since these elements are common to all tracks in the Song.

**Yes** erases all data on the selected track and returns to the MAIN Song Mode. The Song will keep the erased track record enabled, but it will be an empty track.

**No** aborts the erasing of the selected track and returns to the MAIN Song Mode with all previously recorded tracks intact.

## MISC Page

This page contains parameters that determine how the sequencer behaves during playback and recording, and when the K2500 is connected to another MIDI sequencer. Most of these settings are stored in the Master Object, (all except the Auto punch-in points) and none of these are saved with the Song.

```

SongMode:MISC  Events:188K  STOPPED
RecMode :Linear  Quant:Off   CountOff:1
PlayMode:Loop   Grid :1/8   Click :Rec
KeyWait :Off    Swing:0%    ClickCh :16
Locate  : 1:1   Sync :Both  ClickPrg:198
AutoIn  : 1:1   Clock:Int   ClickKey:C 4
AutoOut : 1:1   Tempo:Auto  ClickVel:90
Record  Play  Stop  New  In/Out  MAIN

```

PARAMETER	RANGE OF VALUES	DEFAULT
RECORD MODE	LINEAR, FIXLEN, LOOP, UNLOOP, AUTO	LINEAR
PLAY MODE	LINEAR, LOOP, LIST, CHAIN	LOOP
KEY WAIT	ON, OFF	OFF
LOCATE	-25:2 – 999:4	1:1
AUTO IN	1:1 – 999:4	1:1
AUTO OUT	1:1 – 2500:	1:1
INPUT QUANTIZE	OFF, 1% – 100%	OFF
GRID	1/1 – 1/384	1/8
SWING	-99% – 125%	0%
SYNC	BOTH, XMIT, RECV, OFF	OFF
CLOCK	INT, EXT	INT
TEMPO	AUTO, FIXED	AUTO
COUNTOFF	OFF, 1, 2, 3	1
CLICK	OFF, REC, ON, CNT	REC
CLICK CHANNEL	1 – 16	16
CLICK PROGRAM	1 – 999	198
CLICK KEY	C-1 – G9	C4
CLICK VELOCITY	1 – 127	90

The top line displays the amount of free RAM space and the current sequencer state, which is one of the following: STOPPED, PLAYING, REC READY, RECORDING, KEY WAIT, or EXT. CLOCK.

### RecMode

When recording the first track of a new Song, you will be able to record as if you had an endless length of “tape” no matter what the Record Mode is set to.

When you record the first track, all Record Modes operate the same way. This is because until you define the length of a new Song, its End point is the default setting of Bar 8001, the maximum amount of Bars in a Song. The End point is referenced and modified in different ways depending on the RecMode.

The End point of a song is defined when:

- 1) The **Stop** button is pressed to end the recording of the first track. The new End point is aligned to the nearest downbeat of the (empty) Bar immediately following the last Bar you were recording when **Stop** was pressed.

- 2) **Stop** is pressed while recording any track past the previously set End point in Linear or UnLoop Record Mode. Again, the new End point is aligned to the downbeat of the (empty) Bar immediately following the last Bar you were recording when **Stop** was pressed.
- 3) The AutoOut Bar and Beat is set past current End point, after recording in Auto RecMode, and when the first track is recorded in Auto RecMode, the AutoOut Bar and Beat becomes the End point.
- 4) A new End point is entered in the EVENTS edit mode.
- 5) Using the Track edit functions Copy, Insert, and Delete to alter the Song's length.

The End point of a song is used as a loop point in Loop and UnLoop modes, and it defines the Fixed Length of a Song when you record in FixLen mode.

**Linear** - Record as if you had a nearly endless length of "tape".

**FixLen** - The song will not continue recording past the End point of the song when the RecMode is set to Fixed Length. Recording will automatically stop at the End point.

**Loop** - While RECORDING, the song will play the data between 1 : 1 and the End point over and over, allowing you to overdub in each pass if the Mode parameter on the MAIN SongMode page is set to Merge. Make sure the Mode parameter is set to Merge if you intend to overdub in Loop Record, or else each consecutive pass in Loop RecMode will erase the data recorded on the previous pass. While you are still recording in Loop mode, you may selectively erase individual note events by pressing and holding the **Enter** button and depressing the desired notes on the keyboard during the times you would like them erased. Once the Song is saved, you can not use this feature to erase individual notes.

**UnLoop** - When recording in UnLoop Mode, any existing tracks will be played back as if they were looping from Bar 1 : Beat 1 to the End point, but they are actually being re-recorded linearly over absolute Bars and Beats until you press **Stop**. UnLoop allows you to record a linear track over a short looping section without first having to copy the section over and over again to achieve a new desired Song length. The End point of the Song is extended to the downbeat of the (empty) Bar immediately following the last Bar you were recording when **Stop** was pressed.

For example, let's say you have a recorded a four bar drum loop and now want to record an eight bar bass line. This would be a situation where UnLoop would come in handy. While the drum track keeps looping, the bass track will record in linear fashion, and the end point will be moved to the point at which you press **Stop**. Actually, the drum track will also change. It will play through its loop twice, but while the information is repeating in the loop, it will be recorded to the track. So now if you look at the drum track, you will see information in bars 5-8 (a duplicate of the information in bars 1-4).

**Auto** - Set RecMode to Auto to punch-in record, (either in merge or erase mode), on a track between the Bars and Beats Defined in AutoIn and AutoOut. To punch in and continue recording until you press **Stop**, set the AutoOut point before the AutoIn point.

## PlayMode

The PlayMode setting, along with the End point, determine how the Song or Songs are played back. The song's PlayMode setting will be ignored when the song is included as a step in an arrangement; See "Edit Song: ARRANGE Page" on page 12-46.

**Linear** - Set the PlayMode to Linear to hear the current Song played only once, from the song position set in the Locate parameter to the End point. The Song will be returned to the Locate Bar and Beat when it reaches the End point.

**Loop** - The Loop PlayMode will loop the current Song from the End point back to Bar 1, Beat 1 continually during playback until **Stop** is pressed.

**List** - Song Objects in memory can be played back to back in numerical ID# order starting from the current Song followed by the Song with the next highest ID#. Once the current Song reaches its End point, immediately the CurSong parameter is updated to the next highest Song Object ID# and it will play from Bar 1, Beat 1 to its End point. When the Song with the highest ID# in memory has played through to its End point, the sequencer will stop and that Song will be the new "CurSong".

**Chain** - For every Song, there is a parameter called ChainTo found on the COMMON page that determines what other Song, if any, will immediately follow this Song's playback when PlayMode is set to Chain. When the current song ends, the ChainTo Song will replace the previous CurSong, and if its ChainTo parameter is set to any value other than "0 None", then the Chain PlayMode will continue playback with the next Song being chained. When the last Song in the chain has played through to its End point, the sequencer will stop and that Song will be the new CurSong.

The most common use for the Chain PlayMode is to construct a Set List of different Songs. Since the Songs chained together are played immediately after one another, it is recommended that you include a few measures of silence either at the end or the beginning of each Song in the Chain.

### **KeyWait**

KeyWait specifies whether the sequencer will wait for a Note event before going into PLAYING or RECORDING status. With the KeyWait On, press the **Play** soft button while the sequencer is STOPPED or REC READY and the new status, KEY WAIT, will flash in the Song Status Field until a key is played. You can override the KEY WAIT status by pressing the **Play** soft button twice.

### **Locate**

The Locate Bar and Beat will change in real time during play back and recording to reflect the Song's current position. It can be set to a Bar and Beat before (negative values) or during a Song. Once a Song's length is defined, the End point is the maximum value for the Locate parameter. If Locate is not set to 1 : 1, the count off, if any, is disabled during play back or recording.

This parameter is identical to the Locate parameter on the MAIN page.

### **AutoIn**

When you are in Auto record mode, AutoIn is the Bar and Beat when the recording will begin. If AutoIn is not set to 1 : 1, the count off, if any, is disabled. The AutoIn setting will not have an effect on recording unless the RecMode is set to Auto.

### **AutoOut**

When you are in Auto record mode, AutoOut is the Bar and Beat when the recording will stop. Set the AutoOut location earlier than the Bar and Beat defined as the AutoIn point in order to record to the very end of a Song. The AutoOut setting will not have an effect on recording unless the RecMode is set to Auto.

### **Input Quantize (Quant)**

This parameter determines how much Note events are moved towards grid locations upon the initial input of the events. If set to Off, no Quantizing will occur while you record, and the exact timing of your performance will be preserved during play back. If set to 100%, every recorded Note event will be aligned to the closest grid location, defined by the Grid setting. Input

Quantize is used to quantize your performance as you record it in. However, you may wish to record without quantization and go back and quantize at a later point. To do this, use the Quantize function in the Track editor, as described on page 12-36.

## Grid

This setting determines the size of the Input Quantize grid expressed as a fraction of a Bar with a 4/4 meter. Set Grid to 1/1 for whole note grid, 1/16 for sixteenth notes. All of the standard note durations and every fractional Bar divisions in-between (including triplets, e.g., 1/12=1/8tr, 1/24=1/16tr, etc.) are available as the size of the Input Quantize grid. You can select commonly used Grid values by double-pressing the Increment/Decrement (+/-) buttons.

## Swing

The Swing percentage is applied to the quantize grid. Zero percent swing is straight time, 100% produces a swing (triplet) feel. A positive Swing value determines how close every other grid location is moved to a point 1/3 of the way towards the next grid point. Negative Swing moves every other grid location closer to a point 1/3 of the way towards the previous grid point.

## Sync

The Sync parameter is used in conjunction with Clock. It controls transmission and reception of MIDI sync messages except actual clock. These are the messages it controls: Song Start, Song Stop, Song Continue, Song Select, and Song Position Pointer.

## Clock

Specifies the source clock as being internal or external. When the Clock is set to external, the K2500 will wait to receive MIDI clock data, via its MIDI In Port, from another device capable of generating MIDI clock data before playback and real time recording can begin.

## Tempo

The Tempo parameter, when set to Fixed, provides a tempo lock feature to override any real time tempo changes recorded into a sequence. If set to Auto, tempo changes will be respected.

## CountOff

Selects the number of bars of countdown, if any, before playback or recording starts. This works in conjunction with the click, so if the click is Off, the CountOff setting will have no effect. If the click is set to record only, then the CountOff will only happen when RECORDING.

## Click

The Click parameter controls the click behavior. Set to Off, there is no click, and consequently no CountOff. When it is set to On, a click is present during playback and recording. To have a click only while RECORDING, set the Click to Rec. The Cnt value means that there will only be a click during RECORDING CountOff, if any.

## ClickCh

Specifies which MIDI channel will be used for the metronome click.

## ClickPrg

If click is in use, ClickPrg specifies which Program will be used as the metronome click's sound. The Click Channel will be locked on to this Program internally, and this program number will be transmitted via MIDI on the Click Channel to external any device(s) when playback or record is started. The default click program is "198 Click," which uses the Clave keymap.

**ClickKey**

The ClickKey is the note to be used for the metronome click.

**ClickVel**

ClickVel determines the attack velocity to be used by the metronome click. The first beat of each measure will be the click played at exactly this velocity level while the other clicks will be scaled to about 90% of this value as a way to provide an accent.

**Soft Buttons on the MISC Page**

**Record**, **Play(Pause)**, and **Stop** work the same as they do on the MAIN Page, described above.

**New** selects “1 NewSong” as the current Song and jumps back to the SongMode : MAIN page. The tracks in the new Song will be empty, but all initial program, volume and pan settings, and all parameters in the MISC page remain set the same way they were in the previous Song.

**In/Out** provides a quick way to enter the Auto punch-in points in real time. This button has no effect when the sequencer is stopped, but pressing it once during playback (or while recording) will set the AutoIn point to the Bar, Beat, and Tick nearest to the time you pressed this soft button. Press InOut a second time to set the AutoOut point.

The AutoIn and AutoOut parameters only display bars and beats. However, the actual In and Out points will be precise to the Tick occurring at the time the In/Out button was pressed.

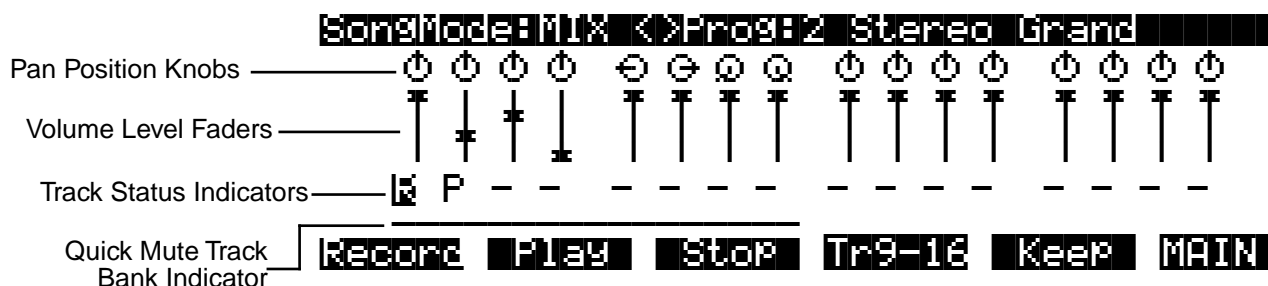
**MAIN** returns you to the MAIN Page.



## MIX Page

The MIX page allows you to set and keep new initial settings for all sixteen tracks' program changes, volumes, and pan positions. You can set these three parameters for each track and then press the **Keep** button, prompting a dialog that asks if you really want to update these settings. Press **Yes** to make the changes. Another way to set these initial settings is to press **Record** to put the sequencer into REC READY status, make any desired changes, then press the **Stop** soft button.

You can also record real-time changes by changing the value of the highlighted fader or knob while recording on that track. If you do make recording changes in real-time, though, make sure that the Record Mode is set to Merge, or else the data previously recorded on the track will be erased.



### Changing Programs on the MIX page

The top line of this page displays the Program number and name for the currently highlighted track. The different tracks are selected by moving the cursor with the Left and Right arrows (< / >). The Program for each track can be changed in the MIX page with the CHAN/BANK buttons. Pressing the CHAN/BANK buttons at the same time will jump to the next Bank of 100 Programs.

### Pan Position

Position the cursor over any one of the sixteen Pan Position "Knobs" on the display and turn the Alpha Wheel to change the panning for the selected track. The graphic display will move smoothly between the left and right settings and these changes can be recorded in real time. Use the numeric keypad to enter in a value between 0 (hard left) and 127 (hard right) if you want to have a track jump immediately to a new pan position. The default pan position is 64 (straight up).

### Volume Level

Position the cursor over any one of the sixteen Volume Level "Faders" on the display and turn the Alpha Wheel to change the MIDI Volume for the selected track. The graphic display will move smoothly, setting volume changes that can be recorded in real time. Use the numeric keypad to enter in a value between 0 (no volume) and 127 (maximum volume) if you want to have a track jump immediately to a new volume level. The default value is 127 (maximum volume).

### Track Status Indicators:

Using the Up, Down, Left, and Right cursor arrows to position the cursor onto a Track Status Indicator, you can toggle an empty track (-) into Record (R) with the Alpha Wheel or Increment/Decrement buttons.

Once a track contains data, it will have a (P) as a Track Status Indicator, and it will be played during playback. You now will be able to toggle between Play (P), Mute (M), and Record (R).

The track selected as the RecTrk will have an (R) in this field, designating it as the record enabled track. If the RecTrk is set to Mult, then all of the empty tracks will have Record (R) as their Track Status Indicator. If RecMode is not set to Mult and there isn't a track with an (R) in this field, the RecTrk parameter's value will be None. If you are in Multi record mode, and have turned all record enabled tracks (R) back to empty (-) so that there isn't a track with an (R) in this field, the RecTrk will remain set to Mult.

### Quick Mute Track Bank Indicator

This parameter appears as an underscore directly below the graphics for either the bank of tracks 1 -8 or tracks 9 - 16. It indicates which bank of eight tracks will respond to the Quick Mute feature, described with the **Tr 1-8/9-16** soft buttons, below.

## Soft Buttons on the MIX Page

**Record**, **Play(Pause)**, and **Stop** work the same as they do on the MAIN page, described earlier.

**Tr 1-8 / (Tr 9-16):** The eight mode select buttons to the left of the screen on the front panel of the K2500 (PROGRAM - DISK) are used as Track Mutes when on the MIX page. For example, press the SETUP button to mute track 2; notice that its track status indicator changes to "M." Each of these buttons has a LED to indicate that the corresponding track is being muted, but since there are sixteen tracks and only eight buttons, this soft button will select, and display, the bank of eight tracks that can be muted in this fashion.

Press the **Tr 1-8** soft button to toggle the Quick Mute Track Bank Indicator under tracks 1 -8 or tracks 9 - 16, selecting which bank of eight tracks will respond to the Quick Mute feature.

**Keep:** If you have made any changes to the initial program, volume or panning of a track on the MIX page, press **Keep** to prompt this dialog:

Update initial Prog/vol/Pan?

Yes

No

Press **Yes** if you are sure you want to update the initial program change, MIDI volume, and pan settings for tracks already containing data in the current Song, to the new values you just made on the MIX page. The settings will be modified and you are returned to the MIX page. Press **No** to abort the updating of the initial program change, MIDI volume and pan settings for the current Song.

**MAIN** returns you to the MAIN page.

## The Edit Song Pages

In general, you access the EditSong pages by pressing the Edit button on the front panel of the K2500 any time you are in Song mode. The exception is if a program is currently highlighted on the display; in this case, you will enter edit mode for that program.

There are a few conventions shared by all of the EditSong pages (except the Arrange page). Displayed at the top of each EditSong page is the name of the page and the currently selected track(s). All of the values for the parameters found in any of the EditSong pages are saved in the Song Object.

### Soloing the Current Track

In all of these Edit pages you can **solo** the current track by pressing either the **Setup** or **Quick Access** mode buttons. This in effect mutes all other tracks in the current Song. Using the CHAN/BANK buttons allows you to select the current track for editing and/or soloing. You can select the track to solo even if you are already in the solo mode.

## Edit Song: COMMON Page

Press the Edit button on the front panel of the K2500 to display the Edit Song: COMMON page and begin editing a Song. This is where you will find parameters common to all tracks, such as tempo and time signature, control parameters for effects and arrangements, and soft buttons for switching to other Song edit modes.

```

EditSong:COMMON <> track:1
Tempo :120 StartStep :1
TimeSig :4/4 TempoControl:Song
EffectChan:1 TriggerChan :1
ChainTo :0 None TriggerCtl :ON
TrackDest :---- ---- ---- ----
DrumTrack :---- ---- ---- ----
<more> EVENT TRACK STEP ARRANG <more>

```

PARAMETER	RANGE OF VALUES	DEFAULT
TEMPO	1 – 255 BPM	120
TIME SIGNATURE	1 – 99 / 1, 2, 4, 8, 16	4/4
EFFECTS CHANNEL	1 – 16, NONE	1
CHAIN TO	0 NONE, SONG ID	0 NONE
TRACK DESTINATION	-, L, M, X	-
DRUM TRACK	-, D	-
START STEP	NONE, 1 – 255	1
TEMPO CONTROL	SONG, ARRANGEMENT	SONG
TRIGGER CHANNEL	1 – 16, NONE	1
TRIGGER CONTROL	CONTROL SOURCE LIST	ON

The top line of this page displays the selected track or tracks. Select the available current track(s) for editing by using the CHAN/BANK buttons. Press both of the CHAN/BANK buttons together to select All tracks.

Although the current track information is displayed on the top line, the COMMON page's parameters are global settings for the song and do not directly affect individual tracks.

**Tempo**

This is another place where the Song's initial tempo can be set or modified.

**TimeSig**

Affects the click, playback looping, and locate function as well as some editing operations. Does not change the recorded data.

**EffectChan**

If the FX Mode and FX Chan parameters in Effects mode are set to Auto and Current respectively, this parameter specifies which channel will control effects while in SongMode and during playback of a Song. In this situation, the effect assigned to the program (in the Program Editor) currently on the channel designated as the EffectChan will be heard.

The EffectsChan is selectable per Song. See "Understanding FX Mode and FX Channel" on page 9-2 for more information.

**ChainTo**

The ChainTo setting is only used when the PlayMode parameter on the MISC page is set to Chain. Another Song object is selected as the ChainTo value and will playback immediately after the current Song has played for its entire length.

**Parameters used with the Arrangement Feature**

The next four parameters, as well as the Drum Track parameter, are used in conjunction with the Arrangement feature, which you access by pressing the ARRANG soft button. This feature allows you to create a song by arranging it in a series of sections, called Steps. You can save the various sections of your sequence as separate Songs, then assemble them using the Arrangement editor. This method allows you to save memory, since you can repeat steps without having to duplicate the actual song data.

Another great feature within the arrangement editor gives you the ability to trigger Steps by striking a specific note or group of notes on your controller. See page 12-46 for more information on arrangement features.

**StartStep**

The value for this parameter determines which Step in this Song's Arrangement is played first. If set to None, the playback of the Arrangement, using the transport buttons, will be disabled, though Steps can still be triggered from Note events.

**TempoControl**

When the current Song is composed of other Songs entered as Steps in the Arrangement editor, the TempoControl parameter determines whether the current Song's tempo setting or the arranged Songs' tempo settings will be the tempo(s) used for the playback and recording of the Arrangement.

Set TempoControl to Song if you want to use the current Song's tempo as the master tempo. All of the Steps' Songs will playback at the same tempo, ignoring the tempos originally set for each Song when you recorded or edited them.

Use the Arrangement setting for TempoControl if you would like the current Song (containing arrangement data) to playback and record at the tempos originally defined in each Arrangement Step's Song, ignoring its own tempo setting. The master tempo will change to the tempo of the current Step's Song

**TriggerChan**

This parameter controls the MIDI channel used to trigger arrangement Steps via key presses as defined in the Arrangement editor.

**TriggerCtl**

This parameter specifies the global control source used to enable the triggering of Arrangement Steps via key presses as defined in the Arrangement editor.

**TrackDest**

```
TrackDest :LLL- M--- -M-- ---x
DrumTrack :D--- ---- -D-- ----
```

The MIDI data on each track has a destination assignment selectable in the TrackDest field. There are four possible indicators:

- (-) This is the default setting for all tracks in a new Song. A hyphen (-) means that MIDI data on the track is transmitted both locally to the K2500, and out the MIDI Out Port (on the track's channel) to any external devices.
- (L) An (L) means that the track's MIDI data will only be transmitted locally to the K2500's internal sound generator. None of the data on that track will be sent to the MIDI Out Port.
- (M) An (M) means that the track's MIDI data will only be transmitted out to external devices via the MIDI Out Port on the back of the K2500.
- (x) Select the (x) to disable the transmission of the track's MIDI data to either of the destination assignments, (L)ocal or (M)IDI.

**DrumTrack**

Any of the Song's tracks can be defined as "Drum Tracks" so that their Note events do not get transposed when a transposition is applied in the Arrangement editor.

This feature is particularly useful when a drum kit Program (or any other non-pitched Program) is used in a Song, assigned to a Step of an Arrangement, which is being triggered over a range of keys, and you want the sounds produced by each note number in that Program to be preserved in each transposition. If in the Song being used as a Step in an Arrangement (NOT the Arrangement Song itself), there is a (D)esignating the track playing the drum Program as a "Drum Track", the originally recorded Note events on that track will remain unchanged.

The Drum Tracks settings do not have any effect on edits made in the EditSong: TRACK mode. Any tracks defined as "Drum Tracks" will be transposed when a transposition is applied to these tracks from the EditSong: TRACK mode.

**Soft Buttons on the Edit Song: COMMON Page**

**EVENT** accesses an Event List style editor. From the EditSong: EVENT page, you can scroll through, modify, add, or delete any or all of the tracks' MIDI events. The EditSong: EVENT page is described on page 12-28.

**TRACK** brings you to the EditSong: TRACK page. This page accesses useful track based edit functions. There is a selectable edit function that can be applied to the selected track or all tracks in your Song. The EditSong: Track page is described on page 12-31.

**STEP** brings up the EditSong: STEP page, from which non-real time note/rest entry is performed. The term STEP is used here to refer to Step recording, and does not refer to the Steps on the Edit Song: ARRANGE page. The EditSong: Step page is described on page 12-43.

ARRANG enters the EditSong : ARRANG page, gaining access to a group of arrangement specific parameters. The EditSong: ARRANG page is described on page 12-46.

Edit Song: EVENT Page

Every type of recorded MIDI event is visible from this page. You can view and change these events if necessary.

Location	Bar:Beat:Tick	Event Type and Value			
editSong:EVENT (Ch 1)		<>Track:1			
1:1	1:1.000	CTRL	BANK	0	
1:1	1:1.000	PCHG	64		
1:1	1:1.000	CTRL	VOL	127	
1:1	1:1.000	CTRL	PAN	64	
1:1	1:1.000	C 3	v 98	^ 93	0.356
2:1	2:1.000	END			
Cut	Copy	Paste	View	AllOn	Done

To scroll through the events, make sure the first column bar and beat is highlighted. Use the wheel, the up and down cursor buttons, or the + and - buttons. As you scroll through the events, each event is executed by the sequencer. In the case of note events, you will hear the note played, although the duration will be short. If you have scrolled through a Sustain (controller 64) message with an On value then you will hear the note sustain as if the sustain pedal was depressed. The note will continue to sustain until you scroll through a Sustain message with a value of Off.

You can also jump directly to a specific bar and beat by typing the bar number and beat number, then pressing ENTER. Keep in mind if you have controller or program data previous to the point that you jump to, those events may not have been executed and you may hear unexpected results. For instance, if you have program changes at bar 1 and bar 8, then if you jump from bar 1 to bar 9 any notes you scroll through will be played with the program change from bar 1.

The channel of the selected Event is displayed on the top line of the page. When all of the Tracks are being viewed in the Event List, the Track of the selected Event is displayed along with its channel. Use the CHAN/BANK buttons to select an active track to view and edit the MIDI events recorded on it. You can view the data on all tracks by pressing both CHAN/BANK buttons at the same time. As you scroll through each Event, the track, in addition to the channel, for the selected Event will also be displayed.

Location

The first column represents the Bar and Beat Locations of the different events in a Song. Scroll through the events on the selected track(s) with the Alpha Wheel or enter in a specific Bar and Beat on the numeric keypad to jump to events occurring on that Beat. A quick way to jump to the End point in a track is to press 9999 and then Enter on the numeric keypad.

Bar, Beat, and Tick

Bar, Beat, and Tick are editable parameters for each Event. They determine when an Event happens relative to the other Events within the Song. To enter values with the numeric keypad, first position the cursor in this column and then simply enter up to three digits if you only want

to adjust the Tick value, four or five digits in order to change the location to a new Beat and Tick within the current Bar, or enter in the complete Bar, Beat, and Tick value to relocate the event to another Bar. No punctuation is necessary when entering any of these values.

### Event Type and Value

The Event Type field displays the MIDI event type at each Event List location in the Song. With the exception of a Note event, which is displayed as the actual note name (e.g., C#4), an event type cannot be changed to another event type. You can also change controller numbers and values on the EVENT page.

The PCHG, BEND, MPRS, SYSX, TMPO, and END events will show their values in the display field directly to the right of the Event Type field. All of their values can be edited. Here are the ranges of values for each event type:

<b>Program Change (PCHG)</b>	0 - 127
<b>Pitch Bend (BEND)</b>	(-8192) - 8191
<b>Mono Pressure (MPRS)</b>	0 - 127
<b>System Exclusive (SYSX)</b>	The message in hexadecimal form
<b>Tempo (TMPO)</b>	1.00 - 255.00 bpm
<b>End Point (END)</b>	Any Bar : Beat : Tick setting

Note events have four editable values: Note Number, Attack Velocity (indicated by a "v"), Release Velocity (indicated by a "^"), and Note Duration.

<b>Note Number</b>	C-1 – G9
<b>Attack Velocity</b>	v1 - v127
<b>Release Velocity</b>	^1 - ^127
<b>Note Duration</b>	Beats : Ticks

Controller events have two editable values: Controller Type and Controller Value. Defined controllers are referred to by their names.

<b>Controller Type</b>	0 - 127
<b>Controller Value</b>	0 - 127

### Soft Buttons on the Edit Song: EVENT Page

**Cut** removes the currently selected event from the Event List and temporarily stores it in a memory buffer so that you can immediately Paste it into a new location.

**Copy** makes a duplicate of the currently selected event and temporarily stores it in a memory buffer so that you can immediately Paste it into a new location.

**Paste** inserts the most recent cut or copied event into the Event List at the currently selected Bar : Beat : Tick location. The pasted event will share the same location with the event that already existed at that location in the Event List, but it will appear before the pre-existing event.

**View** brings you to the EVENT : View page where you can set the view filter parameters for the Event List.

```

EVENT:View
Notes      :On      Volume      :On
PitchBend  :On      Pan          :On
ModWheel   :On      Bank/Prog   :On
Data       :On      Tempo       :On
Sustain    :On      SysEx       :On
Pressure   :On      Other        :On
Toggle AllOff AllOn Done

```

The EVENT edit page can be set for you to view all, some, or just one type of MIDI event on a track by adjusting which events are on or off in the EVENT : View page. If you turn an event Off for viewing, then when you return to the Edit Song: EVENT page, the events will not be seen and they will not be executed as you scroll through the list. For example, if you turn sustain Off, then as you scroll through the list none of the notes normally affected by sustain will be sustained. However, if you exit the editor and play the Song, all events will be executed regardless of the View filter settings.

When all of the parameters in the EVENT : View page are set to On, the Event list in the EditSong : EVENT page will display all of the recorded MIDI data on the selected track(s). If any of these view parameters are set to Off, the **View** soft button in the EditSong : EVENT page will be in brackets (as shown below) indicating that you might not be seeing all of the recorded MIDI data on the selected track(s).

```

Cut Copy Paste [View] AllOn Done

```

Press the **Toggle** soft button to switch all View filter parameters to the opposite of each one's currently set value. **AllOff** sets all View filter parameters to Off. The only item viewable in the EditSong : EVENT page will be an END point. **AllOn** sets all View filter parameters to On. This enables you to view all of the MIDI events that are on any or all tracks. The **Done** soft button in the EVENT : View page will return you to the EditSong : EVENT page. The view filter settings will be applied to the Event List.

**AllOn** on the EditSong : EVENT page sets all View filter parameters to On. This enables you to view all of the MIDI events that are on any or all tracks. The **View** soft button will no longer be in brackets, indicating that all recorded event types are being listed.

**Done** on the EditSong : EVENT page returns to the EditSong : COMMON page.



## Edit Song: TRACK Page

This page allows you access to useful track-based edit functions. These functions are:

- Erase
- Copy
- Bounce
- Insert
- Delete
- Quantize
- RefQuant
- Shift
- Transpose
- Change
- Thin
- Remap
- Grab

For each function, there is a set of parameters to control how the function operates, and on what region of the selected track(s). As usual, the top line of this page displays the selected track or tracks. Select the available current track(s) for editing by using the CHAN/BANK buttons. Press both of the CHAN/BANK buttons together to select All tracks.

Below is an example of the Edit Song: Track page for the Bounce function.



You will notice that the screen is divided into two halves, with the right half being its own "window". This is called the Region/Criteria window. The parameters in this window are used to select the range of bars as well as which types of events will be edited with using the chosen function. The parameters in this window will generally be the same for most functions. However, for certain functions, certain parameters may not apply. For example, Quantize and Transpose apply only to notes, while Remap applies only to Controllers. In addition to the Region/Criteria window parameters, the Locate parameter is also found on each function. Since these parameters are common to the various functions, we will define them first. Then we'll describe the individual functions along with the parameters specific to each, which are normally found on the left side of the screen. Two functions, (Quantize and Reference Quantize) have a unique parameters in their Region/Criteria windows. We'll describe those parameters along with the functions.

Once you've chosen a function and set the parameters to your liking, press **Go**. This executes the editing function. You can then play the sequence to hear the results of your edit. If you don't like your edit, simply exit the editor and press **No** when you are asked if you want to Save. If you do like your edit, you can press **Done** and use the more buttons to get to **Save**, or just exit the editor and save the changes. Or, you can go to another edit function. Keep in mind though, that if you choose to perform more than one edit without saving, and you are not satisfied with one of the changes you make, you will have to exit the editor without saving and then redo each of the changes you made. That's why it's usually best to save after each successful edit.

## Common Parameters for Edit Song: Track Functions

### ***Locate***

This parameter is available in every TRACK edit function.

The Locate Bar and Beat will change in real time during play back and recording to reflect the Song's current position. It can be set to any Bar and Beat, including negative values. Play back begins at, and Stop resets the Song to the Locate Bar and Beat.

## Region / Criteria Window Parameters

### ***From and To***

From and To are available in most TRACK edit functions to define a range of time on the selected track(s).

The From value defines the first Bar and Beat in a range of time selected for editing. Although From is always expressed in Bars and Beats, and will be aligned exactly on the Beat when adjust manually, it is possible for From to represent a value with finer resolution when a range of time is defined in real time using the **FromTo** soft button.

The To value defines the Final Bar and Beat in a range of time selected for editing. Although To is always express in Bars and Beats, and will be aligned exactly on the Beat when adjust manually, it is possible for the To value to represent a value with finer resolution when a range of time is defined in real time using the **FromTo** soft button.

### ***Events***

Any and all types of MIDI events are available for editing, selectable in this parameter. Some events will provide you with settings for a range of values, or other MIDI event specific criteria. Available Values are: All, Notes, Controllers, MonoPress, PitchBend, ProgChange, PolyPress, SysEx, and Tempo

### **When Events is set to: All**

All MIDI events on the track(s) you are editing, that occur in the region of time between the From and To settings, will be affected by the edit function.

### **When Events is set to: Notes**

Note Number and Key Velocity ranges can be set for Note events.

### ***LoKey***

Determines the lowest note in a range of notes to be affected. This can be set to any MIDI note value; the default is C-1.

### ***Hi(Key)***

Determines the highest note in a range of notes to be affected. This can be set to any MIDI note value; the default is G9.

### ***LoVel***

An attack velocity range can be specified as criteria for selecting Note events for editing. The LoVel parameter sets the lowest velocity a Note needs to have in order to be edited. Notes on the selected track(s) with a attack velocities lower than the LoVel will not be affected by the edit. The available values are 1 – 127; the default is 1.

**Hi(Vel)**

The Hi(Vel) parameter sets the highest attack velocity a Note needs to have in order to be edited. Notes on the selected track(s) with attack velocities higher than the Hi(Vel) will not be affected by the edit. The available values are 1 – 127; the default is 127.

**When Events is set to: Controller**

If your Song contains any MIDI Controller data, the Ctl parameter is used to select particular controller data to be edited. Refer to the Control Source list in Chapter 6 of the *Reference Guide*.

**LoVal**

You may further specify a particular range of values to edit by setting a high and low value. LoVal will define the lowest modifiable value in the selected controller's recorded data. Value ranges are not definable when Ctl is set to All. Available Values are 0 - 127.

**Hi(Val)**

Hi(Val) will define the highest modifiable value in the selected controller's recorded data. Value ranges are not definable when Ctl is set to All. Available Values are 0 - 127.

## Soft Buttons on the Edit Song: Track Page

**FromTo** is a quick way to define the region of time you intend to edit. There are a couple of ways to use this feature when the sequence is playing back in real time, and both ways will set the temporal boundaries of the region to a finer resolution than Bars and Beats.

One way is to first position the cursor over the From field in the *Region / Criteria Window* and then press the **Play** soft button. At this point, every time you press **FromTo**, the From Bar and Beat will reflect a location in the Song at each time the button press was made. Position the cursor over the To field and set the To location in a similar fashion.

When the cursor is positioned somewhere other than over the From or To fields, the function of the **FromTo** soft button is slightly different. Now, once you start the sequence playing back, and you press **FromTo** before the current To Bar and Beat location, you will set a new From location. Without stopping the sequence and starting it again, press **FromTo** a second time, and if it was pressed at a point after the From location, that point in time will represent a new To location. If the second **FromTo** press happens before the current From Bar and Beat, that point in time becomes the new From location, and the old From location becomes the new To.

To sum up this functionality, when this button is pressed while the Song is playing, the K2500 will know which field, (the From or the To), to update based on the relative position, (before or after the current settings), when the button was pressed.

**Play** will start the playback of the Song from the Bar and Beat set in the Locate parameter. When the Song is playing, this soft button becomes **Pause**.

**Stop** stops the playback of the Song and return to the Bar and Beat set as the Locate value.

**Go** performs any of the Track-based edit functions described above.

**Done** will return you to the EditSong : COMMON page.

## Edit Song: Track Functions – Erase

This function erases specified events from a region of time, but it doesn't delete the region of time. The result is like erasing a section of recording tape. If you want to completely remove a segment and shorten the length of the track, you can do it with the Delete function.

```

EditSong: TRACK          <> track:1
Function:Erase           From : 1:1 To: 2:1
                          Events: SysEx

Locate : 1:1
FromTo Play Stop        Go Done

```

## Edit Song: Track Functions – Copy

Use the Copy function to duplicate the selected events from the current track and place them in the same track or on another track, either merging with or overwriting existing data.

```

EditSong: TRACK          <> track:1
Function:Copy            From : 1:1 To: 2:1
DstTrack:1               Events: Controllers
Location: 1:1            Ctl : MWheel
Mode : Merge             LoVal : 0 Hi: 127
Times : 1
Locate : 1:1
FromTo Play Stop        Go Done

```

If you do not want to copy all of the MIDI events in the defined range of time on the current track, use the Events parameter in the Region/Criteria window to select a specific MIDI event type you would like the edit function to affect. Some event types provide you more criteria selection parameters. It is often a good idea to set Events to Notes when copying, and then add any necessary controller or other data to the track at a later time.

**DstTrack:** 1 - 16 / All

Select a destination track for the copied events with the DstTrack parameter. All selected events described in the Region / Criteria Window will be placed in the destination track(s) at any Bar and Beat you specify.

If the currently selected track is All tracks then the destination track will be All tracks as well.

No matter what channel the current track (source track) is set to when you use the copy function, the Events will be played on the destination track's channel.

**Location:** 1 : 1

Specify a Bar and Beat location in the destination track where the copied data will be placed with the Location parameter. If the length of the copied region extends from the Location point beyond the Song's existing End point, a new End point is defined.

**Mode:** Merge / Erase

The Mode setting determines whether the copied events merge with, or erase existing events on the destination track from the location point to the end of the copied region.

**Times:** 1 - 127

The value selected for the Times parameter determines how many copies of the selected region are placed, one after another, in the destination track.

## Edit Song: Track Functions – Bounce

Use the Bounce function to move the selected events from the current track to another track, either merging with or overwriting existing data on the destination track. The Bounce function differs from the Copy function in that the original data is not preserved in the original track. As on a multi-track tape recorder, Bounce will always put the data in the same timeline on the new track that it was on the old track.

```

EditSong: TRACK          <> track:1
Function: Bounce
DstTrack: 1
Mode      : Erase
From      : 1:1 To: 2:1
Events: Notes
LoKey : C -1 Hi: G 9
LoVel : 1 Hi: 127

Locate : 1:1
FromTo Play Stop Go Done
  
```

**DstTrack:** 1 - 16

Select a destination track for the events to be moved to with the DstTrack parameter. All selected events described in the Region / Criteria Window will be placed in the destination track at the data's original location.

No matter what channel the current track (source track) is set to when you use the bounce function, the events will be played on the destination track's channel.

**Mode:** Merge / Erase

The Mode setting determines whether the bounced events merge with, or erase existing events on the destination track from the location point to the end of the copied region.

## Edit Song: Track Functions – Insert

The Insert function is used to add blank time to the current Song, modifying the Song's End point appropriately. The Insert function will affect all tracks. This is similar to splicing a piece of blank tape to an existing segment of recording tape.

```

EditSong: TRACK          (Effects all tracks)
Function: Insert
Location: 1:1
Amount  : 1:0

Locate : 1:1
FromTo Play Stop Go Done
  
```

**Location:** 1 : 1

The insertion point for the blank time being added is selected as a Bar and Beat Location value. Events that occurred at or after this Bar and Beat, before you insert time, are not erased when you perform this function, rather they are offset by the length of the blank time being added to a Bar and Beat later in the Song.

**Amount:** 1 : 0

The length of the blank time being added is defined as a number of Bars and Beats in the Amount parameter.

There are no Region / Criteria Parameters available for the Insert function.

## Edit Song: Track Functions – Delete

The Delete function is used to remove a region of time from the current Song. This function is different from the erase function because not only does it remove the events from the selected time, it will delete the entire selected range of time from the Song, modifying the Song's End point appropriately (on all tracks). This is similar to cutting a section out of a tape and splicing the ends.

```

EditSong: [RHCK] (Effects all tracks)
Function: Delete From : 1:1 To: 2:1

Locate : 1:1
FromTo Play Stop Go Done

```

## Edit Song: Track Functions – Quantize

Use the Quantize function to adjust the timing of Note events. Keep in mind that only Note events are quantized; other types of events, such as controllers, are not quantized.

```

EditSong: [RHCK] <> track: 1
Function: Quantize From : 1:1 To: 2:1
Quant : 100%
Grid : 1/8 LoKey : C -1 Hi: G 9
Swing : 0% LoVel : 1 Hi: 127
Shift : 0.000 Release: No
Locate : 1:1
FromTo Play Stop Go Done

```

**Quant:** Off / 1% – 100%

The Quantize parameter determines how much the selected Note events are moved towards grid locations. If set to Off, no aligning of previously recorded Notes to grid locations will occur. If set to 100%, every recorded Note event will be aligned to the closest grid location, defined by the Grid setting. Notes will be moved to a position half way between the grid location and the original Note event location if Quant is set to 50%.

**Grid:** 1/1 – 1/384

This setting determines the size of the Quantize grid, expressed as a fraction of a Bar with a 4/4 meter. Set Grid to 1/1 for whole note grid, 1/16 for sixteenth notes. All of the standard note durations and every fractional Bar divisions in-between are available as the size of the Input Quantize grid. The double button press for quickly selecting Grid values is pressing the Increment/Decrement at the same time.

**Swing:** -99% – 125%(defaults as 0%)

The Swing percentage is applied to the quantize grid. Zero percent swing is straight time, 100% produces a swing feel (triplet feel). A positive Swing value determines how close every other grid location is moved to a point 1/3 of the way towards the next grid point. Negative Swing moves every other grid location closer to a point 1/3 of the way towards the previous grid point.

**Shift:** -26.020 – 26.020 (Beats and Ticks, 480 Ticks = 1 Beat)

In addition to quantizing the selected note events to specified grid locations in varying amounts, the Quantizing function allows you to offset the original note locations forward and backward in time any number of ticks (1/480th of a Beat) up to @ 26 Beats, before aligning them to Grid locations. Shift is used to compensate for any notes played too early or late.

### Region / Criteria Window

**Release:** Yes / No

Set the Release parameter to Yes if you would like each quantized note event's Note Off message to be aligned to the Grid location nearest to the time the key was originally released.

## Edit Song: Track Functions – Reference Quantize

The Reference Quantize function is similar to the Quantize function in that it aligns Note events to a grid. The difference is that the grid locations are not mathematically perfect divisions of a Bar. Instead, Reference Quantize defines the grid based on the timing of note events from a previously recorded reference track.

```

EditSong: [RHCK] <> track:1
Function:RefQuant From : 1:1 To: 2:1
RefSong :1 NewSong
RefTrack:1 LoKey :C -1 Hi:G 9
Timing :100% LoVel :0 Hi:127
Velocity:Off Width :1/8
Locate : 1:1
FromTo Play Stop Go Done
  
```

**RefSong:** 1 NewSong

The previously recorded reference track can be selected from any Song object currently residing in the RAM memory. Select the Song containing the desired reference track as the RefSong value.

**RefTrack:** 1 – 16

The RefTrack value is set to the track number of the desired Reference Track in the Reference Song. The timing of this track's note events is used as the Grid that other tracks will reference when being Referenced Quantized.

**Timing:** Off / 1% – 100%

The Timing parameter determines how much the selected note events are moved towards grid locations. If set to Off, no aligning of previously recorded Notes to grid locations will occur. If set to 100%, every recorded note event will be aligned to the closest grid location, defined by the timing of note events from the reference track. Notes will be moved to a position half way between the grid location and the original Note event location if Quant is set to 50%.

**Velocity:** Off / 1% – 100%

In addition to referencing the timing of note events on the reference track, you may also scale the attack velocities of the note events being quantized to velocity values closer or identical to the velocities played on the reference track.

Leave the Velocity parameter set to Off if you just want to reference the note event timing when reference quantizing. Set Velocity to 100% in order to have the note events being quantized to have velocity values matching those of the note events on the reference track. A setting of 50% will change the velocities to values half way between the values originally played and the values of the velocities on the reference track.

### Region / Criteria Window

**Width:** 1/1 – 1/384

The Width setting determines the duration of a window of time centered around each of the referenced grid locations. If a note event on the track being quantized happens during this window of time, it will be moved closer to the referenced grid location according to the Timing percentage. Note events occurring outside this window of time remain unquantized.

## Edit Song: Track Functions – Shift

The Shift function allows you to offset the existing MIDI events forward or backward in time any number of ticks (1/480th of a Beat) up to approximately 26 Beats. This function does not affect the End point.

Events can not be shifted beyond the End point or before Bar 1 : Beat 1. The events can be shifted only as far as these temporal boundaries. All events that can't be shifted the full Ticks amount will be placed at the boundary location.



**Ticks:** -26.020 – 26.020 (Beats and Ticks, 480 Ticks = 1 Beat)

The Ticks parameter specifies the number of Beats and Ticks that the MIDI events, from within the selected region, are moved forward or backward in time relative to their original locations.



## Edit Song: Track Functions – Transpose

Use the Transpose function to change the MIDI Note numbers of the selected Note events.

EditSong: TRACK		<> track:1	
Function: Transpose		From : 1:1	To: 2:1
Semitone: 12ST		LoKey : C -1	Hi: G 9
		LoVel : 1	Hi: 127
Locate : 1:1			
FromTo	Play	Stop	Go Done

Semitone: -128ST – 127ST

An increment of one semitone represents a change of one MIDI Note number. Note events are only transposable within the range of MIDI Note numbers from zero to 127.

## Edit Song: Track Functions – Change

The Change function is used to modify attack velocities, release velocities, or the values of any existing controller data on the current track. A static change of values can be made as well as having the change take place over a region of time.

Change can not modify or add data that doesn't exist on the current track. If you hear note events played back on a track, then you know there is an attack and release velocity value for each one, and the effect of the Change function can usually be easily detected. Controller values are sometimes more difficult to change since there can be inconsistent gaps of time in-between each controller event.

EditSong: TRACK		<> track:1	
Function: Change		From : 1:1	To: 2:1
Scale : 100%	Events: Velocity	LoKey : C -1	Hi: G 9
Offset : 0		LoVel : 1	Hi: 127
Mode : Constant			
Locate : 1:1			
FromTo	Play	Stop	Go Done

Scale: 0% – 20000%

The selected velocity or controller events' values can be changed to a percentage of the original values determined by the Scale parameter. A setting of 100% has no affect. Values are scaled lower with a Scale percentage set from 0% to 99%. Low values can be set higher using a Scale percentage above 100% on up to 20,000%, although the maximum value of 127 can not be exceeded for any velocity or controller type.

Offset: -128 – 127

Offset can be used alone or in conjunction with Scale to add or subtract a set amount to or from the original (or scaled) values. Values for velocities can not be less than 1 or greater than 127. Values for controllers can not be less than 0 or greater than 127.

As an example, to set all Velocities to a value of 55, you would set Scale to 0% (multiplies all original values by zero) and set Offset to 55 (adds 55 to the product of the Scale parameter).

**Mode:** Constant / PosRamp / NegRamp

Set Mode to Constant to have values modified in a uniform fashion, as determined by the Scale and Offset settings, for the entire selected region of time and range of values.

When the Change function is applied with Mode set to Positive Ramp, the selected velocity or controller values will gradually change over the region of time, defined by the locations set for the From and To parameters, from the original value to the new value determined by the Scale and Offset settings. The first events being modified within the region will have little or no change from their original values. The amount of Scale and Offset applied will increase as the Song approaches the Bar and Beat defined in the To parameter, where the full amount of described change will occur.

You can set Mode to Negative Ramp to achieve the opposite dynamic effect of Positive Ramp. Negative Ramp works in the same way, but the amount of Scale and Offset applied will decrease from the full amount of change described by Scale and Offset to little or no change as the Song approaches the Bar and Beat defined in the To parameter.

## Edit Song: Track Functions – Thin

Use the Thin function to reduce the number of actual controller events used for any controller type on the current track. Sometimes a controller can produce the same noticeable effect by using fewer events to describe its change, which in turn will save memory and possibly prevent timing delays caused by an over abundance of MIDI data.

Edit Song: Track Functions		<> track: 1	
Function: Thin	From : 1:1	To: 2:1	
Percent : 50%	Ctl : MPress		
	LoVal : 0 Hi: 127		
Locate : 1:1			
From To	Play	Stop	Go Done

**Percent:** 0% – 100% (defaults as 50%)

The Percent value determines how much effect the Thin function will have on a stream of controller data. With this value set to 0%, there will not be any reduction of data when Thin is applied. Set Percent to 100% to eliminate most of the specified controller's events from the current track. Even at 100%, certain controller messages will not be eliminated; these include all pitch bend messages with a value of 0 and the initial settings of some controllers, such as volume, pan, etc.

## Edit Song: Track Functions – Remap

Use the Remap function to apply the values of any one type of controller data, already recorded on a track, to another controller type. The effect the real time changes of the “Old” controller had will be replaced by the effect the “New” controller has by using the exact same controller values.

```

EditSong: TRACK          <>Track:1
Function: Remap          From : 1:1 To: 2:1
Old      : MWheel
New      : Volume

Locate   : 1:1
FromTo   Play Stop      Go Done

```

**Old:** Controller Codes (0 – 120)

Defined controllers are referred to by their names.

The “Old” controller is set to the controller type that you wish to remap. This controller data must already exist on the current track in order to apply it to the “New” controller type.

**New:** Controller Codes (0 – 120)

Defined controllers are referred to by their names.

The “New” parameter is set to the controller code you wish to have use the existing values, once used by the “Old” controller, to produce a different effect.

## Edit Song: Track Functions – Grab

Grab is similar to the Copy function, except that the Grab function allows you to copy selected data from tracks that exist in other Songs in memory.

```

EditSong: TRACK          <>Track:1
Function: Grab           From : 1:1 To: 2:1
SrcSong : 1 NewSong      Events: Controllers
DstTrack: 1             Ctl : MWheel
Location: 1:1           LoVal : 0      Hi: 127
Times   : 1
Locate  : 1:1
FromTo   Play Stop      Go Done

```

**SrcSong:** 1 NewSong

The Source Song parameter is set to the ID# and name of the Song in RAM memory that contains the desired track data you wish to grab in order to use it in the current Song. The source track is determined by the <>Track parameter displayed on upper right hand side of the page, selectable with the CHAN/BANK buttons.

**DstTrack:** 1 – 16 / All

Select a destination track for the grabbed events with the DstTrack parameter. All selected events from the source Song and track described in the Region / Criteria Window will be placed in the destination track(s) at any Bar and Beat you specify.

If the currently selected track is All tracks then the destination track will be All tracks as well.

No matter what channel the current track (source track in the source Song) is set to when you use the grab function, the Events will be played on the destination track's channel.

**Location:** 1 : 1

Specify a Bar and Beat location in the destination track where the grabbed data will be placed with the Location parameter. If the length of the grabbed region extends from the Location point beyond the Song's existing End point, a new End point is defined.

**Times:** 1 – 127

The value selected for the Times parameter determines how many copies of the selected region are placed, one after another, in the destination track.

## Edit Song: STEP Page

```

EditSong:STEP                                     <>Track:1
DubMode :Erase
Duration:1/16
Modifier:None
GateTime:75%
Velocity:Played
Locate : 1:1.000
  Play  Stop  <Back  Step>  Done

```

PARAMETER	RANGE OF VALUES	DEFAULT
DUBMODE	MERGE, ERASE	MERGE
DURATION	1/1 – 1/64	1/8
MODIFIER	NONE, TRIPLET, DOTTED, DOTDOT	NONE
GATETIME	0 – 250%	75%
VELOCITY	PLAYED, 1 – 127	PLAYED
LOCATE	BAR:BEAT.TICK	1:1.000

The Step Editor allows for non-real time entry of note events and rests of varying durations and attack velocities at any location within a Song. The Step Editor can be used to enter the first note events into a new Song, but it will not set the Song's End point. Initial program changes, volume and pan settings are remembered as if you recorded the first track using one of the real time recording modes. The End point will be modified if tracks already containing data are Step edited beyond the current End point.

The top line displays the currently selected RecTrk and can then be switched to any other track. When All tracks are selected, the Step edits are performed on the currently selected RecTrk.

## Recording with the STEP editor

Recording using the step editor is easy. Set the parameters to your desired settings and make sure the Locate parameter is set to the bar and beat at which you wish to start recording. Then just strike a key. The note you play will be displayed in the window on the right along with its velocity or duration. If the Velocity parameter is set to Played, then it will reflect the actual velocity you played. Otherwise, it will be specific amount determined by that parameter. The duration of the note is determined by a combination of the Duration and Gate Time parameters.

Once you release the key, the Locate parameter will advance in time, by an amount determined by the Duration parameter. If you want to enter a rest, press the **Step>** soft button without striking a key, and once again the Locate parameter will advance the selected amount.

You can enter note events longer than the selected duration by holding a key and pressing the **Step>** button. You will see the duration of the current note change in the window on the right. This can be a quicker method of entry than changing the duration parameter if you need a note that is twice or three times as long as the duration. For instance, if the duration is set to 1/8, you can quickly enter 1/4 note by holding the note and pressing **Step>** once, or enter a whole note by pressing the **Step>** button three times. Note that if you are using this method and the Gate Time parameter is set to less than 100%, it will only be applied to one of the steps. For example, if you have a Duration of 1/8 and a Gate Time of 75%, then striking a note will produce a duration of 180 (75% of 240). But if you hold the note and press **Step>**, then the duration will be 420 (180 + 240), instead of 360 (75% of 480).

You can strike more than one note at a time to enter chords. You can also strike several notes one at a time (holding on to the previous notes) to create a chord. As long as you do not release all of the notes, the Locate parameter will stay at the same point and the notes will be entered as a chord. If you want to create a chord where certain notes have longer durations than others, strike the notes of the chord, then release the notes you wish to be shorter and press the **Step>** button. The window on the right will change to show only the notes you are still holding with their longer durations.

If you make a mistake while you are entering notes, you have two ways to fix it. With Dub Mode set to Merge, if you are still holding the note, you can press the **<Back** soft button. If you set the Dub Mode to Erase, you can just press the **<Back** button to erase the previous event and then strike the correct note. Or you can set the Locate parameter to the proper spot and strike the correct note.

### DubMode

DubMode determines if the note events entered in the EditSong : STEP page will merge with, or erase all other existing MIDI events located on the selected track, and in the Bars and Beats being Step edited.

You would set DubMode to Erase to replace all existing track data just in the Bars and Beats you edit. Data before and after the edited locations will be preserved. Set to Merge to add note events to existing track data.

The DubMode value will reflect the last value given to the Mode parameter on the SongMode : MAIN page, and if the DubMode in the Step editor is changed, the MAIN page Mode will be changed to the same value.

### Duration

The actual duration of a note event entered in Step time is determined by the values given for the Duration, Modifier, and GateTime parameter.

The Duration parameter sets the base note duration where 1/1 is a whole note, 1/2 is a half note, 1/32 is a thirtysecond note, and so on. The setting for this parameter, along with its Modifier, determine the size of the jump from the current Song location to the next Step entry location. The double button press of the Increment/Decrement buttons is one method of selecting a Step's duration.

Use the numeric keypad to quickly select a note duration and modifier whenever the cursor is positioned over the Duration field. These are the keypad duration settings:

- 1 whole note
- 2 half note
- 3 thirtysecond note
- 4 quarter note
- 5 sixtyfourth note
- 6 sixteenth note
- 7 applies a dotted modifier to current duration
- 8 eighth note
- 9 applies a triplet modifier to current duration
- 0 removes any modifier

### Modifier

The Modifier parameter allows you to change the Duration value with three standard modifiers. Set to None if you do not chose to use a modifier.

Use the triplet modifier for a resulting duration 2/3 of the value set in the Duration parameter. (e.g., an eighth note = 240 ticks, an eighth note triplet = 160 ticks).

A Dotted modifier adds half of the current Duration value to that duration, and a DotDot modifier will add 3/4 of the current Duration value to itself. (e.g., an eighth note = 240 ticks, a dotted eighth note = 360 ticks, and a double dotted eighth note = 420 ticks).

### GateTime

GateTime determines the percentage of the note duration, (set with the Duration and Modifier), that is actually played. The value given for the GateTime parameter does not affect the size of the jump from one Step to another, but only the length of time that the note(s) entered for each Step are sustained. You can use a value greater than 100 for a legato feel, but keep in mind that when two identical notes overlap you may get unexpected results. This is because the Note Off command from the first (overlapping) note will also cut off the second note (when it is identical).

Double-press the - and + buttons to increment this value in 20% intervals.

### Velocity

Each note event entered in the Step edit mode is assigned an attack velocity either by setting the Velocity parameter to a value between 1 and 127, or by setting the value to Played, where the velocity at which you actually enter notes is used for each Step.

### Locate

The Bar, Beat and Tick displayed as the Locate value is the location where the next Step entry will be placed in the Song, and once the Step is entered, Locate will advance an amount of time determined by the Duration and Modifier settings. The Bar and Beat, but not the Tick, will update during playback. It can be set to any Bar and Beat, including negative values, so that playback begins at, and Stop resets the Song to the Locate Bar and Beat. If the Bar and Beat value is set beyond the current End point, and no notes are entered before you press Play, then that location will be ignored when you begin playback and the Song will start from Bar 1 : Beat 1. If set to a negative Bar and Beat, no Step entry is possible.

## Soft Buttons on the Edit Song: STEP Page

**Play** starts the playback of the Song from the Bar and Beat set in the Locate parameter. When the Song is playing, this soft button becomes **Pause**.

**Stop** halts the playback of the Song and return to the Bar and Beat set as the Locate value.

**<Back / Step>** move the current location of the Song backward and forward in time, as reflected in the Locate Bar, Beat, and Tick. How far the location is changed from its current setting is determined by the Duration and Modifier settings.

**Done** returns to the EditSong : COMMON page when you are finished Step editing.

## Edit Song: ARRANGE Page

This page allows you to create a song by arranging other songs together in the order you specify. The other songs become sections (called Steps) of the current song, which can be repeated, transposed, etc. You can even trigger Steps by striking specific keys on the controller.

```

EditSong:ARRANGE          <>Step:5/14
Step :5
Song :206*bow3
Mutes:---- ---- ---- ----
XPose:0st Times:2x  Mode :Next

LoKey:Off HiKey:Off Latch:Off VelTrk:On
Add  Delete SetRng Play  Stop  Done

```

You can also create an arrangement of other songs to play back simultaneously with the current song. This gives you up to 32 tracks: 16 tracks in the current song plus 16 in the song or songs called by the arrangement (which are specified by the ARRANGE page's Song parameter). Alternatively, the current song might not contain any song data *per se*, but could just include the arrangement data, by which the pieces of a composition are assembled together. One thing you can't do, however, is "nest" arrangement songs. In other words, if you include a song with arrangement data in another song with arrangement data, it won't work.

More often than not, you would want to start using the Arrange feature from a "New Song" that doesn't have any track data recorded yet. Press the **ARRANG** soft button on the **COMMON** page to enter the EditSong : ARRANG page, gaining access to a group of arrangement specific parameters. All Songs have these parameters for creating complex arrangements of existing Song Objects, using ranges of the keyboard and realtime performance controls for triggering steps, but not every Song will utilize them.

When a Song containing an Arrangement is selected as the "CurSong" from the **MAIN** page, three dashes (---) appear above the Track parameter name to indicate that the current Song has been saved with arrangement information.

```

SongMode:MAIN  Events:317K  STOPPED
CurSong:207*cowboy  Tempo:85
RecTrk :1  Vol:127 Pan:64  Mode :Erase
Program:48 Studio Kit 1  Locat: 1:1
---
Track  :R - - - - - - - - - - - - - - -
Channel:1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Record Play Stop Erase MISC MIXER

```

These three dashes show that the song includes arrangement information.

When you play a song that includes arrangement information, dashes are replaced by the play indicator (>), Song ID number, and Step number.

```

SongMode:MAIN  Events:317K  STOPPED
CurSong:207*cowboy  Tempo:85
RecTrk :1  Vol:127 Pan:64  Mode :Erase
Program:48 Studio Kit 1  Locat: 1:1
>206-1
Track  :R - - - - - - - - - - - - - - -
Channel:1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Record Play Stop Erase MISC MIXER

```



When an Arrangement is PLAYING, the three dashes on the MAIN page are replaced with the ID# of the Song currently being played (or triggered) in the Arrangement, and the number of the current Step.

The square Track Mode Indicators will flash when MIDI data is present on the tracks during play back of an arrangement. The actual track data playing back is supplied by the Songs entered into the steps of an arrangement.

```

EditSong:ARRANGE      <>Step:4/5
Step :4
Song :0 None
Mutes:----  ----  ----  ----
XPose:0st Times:1x  Mode :Next

LoKey:Off HiKey:Off Latch:Off VelTrk:On
Add  Delete SetRng Play  Stop  Done

```

PARAMETER	RANGE OF VALUES	DEFAULT
STEP	1 – 99	1
SONG	ANY SONG IN MEMORY	0 NONE
MUTES	-, M	- (NOT MUTED)
XPOSE	-128ST – 127ST	0ST
TIMES	1X – 120X, INFINITE	1X
MODE	NEXT, STOP	NEXT
LOKEY	C-1 – G9, OFF	OFF
HIKEY	C-1 – G9, OFF	OFF
LATCH	ON, OFF	OFF
VELTRK	ON, OFF	ON

The first number displayed in the **Step** field on the top line is the current step. The second number is the total amount of steps in the arrangement. The current step in the screen above, for example, is the fourth step in an arrangement consisting of five steps. Use the CHAN/BANK buttons to select the current step for editing.

### Step

The number of steps in an arrangement defaults to 1. All other steps are then added when needed. The step of the Arrangement selected for editing is displayed as the Step. Use the CHAN/BANK arrows to select the different steps in your Arrangement. StartStep on the EditSong: COMMON page (see page 12-25), as well as the key ranges of the steps, control what steps are played. The behavior and control of each step is defined with the parameters described below.

### Song

The Song is the ID# and name of a Song Object in memory that will play back in proper numerical order with the other steps in an arrangement or will respond to any triggering from a defined keyboard range in that step. This song's (up to) 16 tracks will play in addition to any tracks in the current song.

### Mutes

The possible sixteen tracks of the current step's Song can be individually muted. When the selected step's display has tracks with an (M) in this field, those tracks will be muted during playback of the Song in the Arrangement.

### Xpose

Each Step in an Arrangement can impose a transposition on the Song selected in its Song parameter during playback. Xpose determines how many semitones a Step's Song is

transposed above or below the Song's original key during playback. All tracks of the Step's Song will be transposed, except for those tracks defined as drum tracks in the EditSong : COMMON page of the Step's Song. See page 12-27 for a discussion of drum tracks.

### Times

Times sets the number of times that a Song is looped before an Arrangement moves on to its next Step. You can set this value to repeat the Song up to 120 times before the next Step is played, or you can specify an infinite loop ("Inf"), in which case the arrangement will play that Step until **Stop** is pressed. A song's PlayMode setting has no effect when the song is included as a step in an arrangement.

### Mode

The Mode setting found in the Arrangement editor determines whether an arrangement plays the next Step or if the sequencer will stop after playing the current Step. Unless you want the arrangement to loop (i.e., return to step one after playing the song's final step) set the final step's mode to stop. A song's PlayMode setting has no effect when the song is included as a step in an arrangement.

## Triggering Steps from a Key

You can play and transpose the steps in your arrangement with the keys on your keyboard. This is a good way to try out arrangement or composition ideas, and also has some exciting live performance applications. If you were suddenly struck by inspiration at a gig, for example, you could spontaneously repeat or transpose sections of an arrangement.

The four parameters along the bottom of the Arrange page let you set up real time control of the Steps in an Arrangement. Make sure that the TriggerCtl parameter, found on the COMMON page (see page 12-27), is set to ON, which is the default, or to the global control source that you are using to enable this feature. TriggerChan, also on the COMMON page will need to be set to the current MIDI channel; the default is channel 1.

### LowKey

The first parameter, LowKey, is the setting for the lowest note in a keyboard range, that when played by the TriggerChan, will trigger the start of the current Step. Triggering a Step from the LowKey will play back its Song in its original key transposed the number of semitones set in the Xpose setting. As you play up the keyboard chromatically, the Step will transpose its Song in semitone increments. All tracks of a Song being transposed by keyboard triggering will be transposed accordingly, except for tracks defined as Drum Tracks on the COMMON page.

### HiKey

The highest note of the keyboard range designed to trigger the current Step is set in the HiKey parameter. HiKey defines the largest transposition of a Step from the key of the Song triggered by the LowKey. The HiKey must be a higher note value than the LowKey in order to trigger any Step from the keyboard.

### Latch

If Latch is set to Off, the playback of the Step, triggering it from the defined keyboard range, will be gated for as long as the triggering Key is depressed. Set Latch to On if you want the use the keyboard to start the Step, and have it continue playing after you stop holding the key down. The Step will play for its entire length unless you either press **Stop** or retrigger the step.

### VelTrk

Turn the VelTrk parameter On to make your real time Arrangements more dynamically controllable. The original attack velocity of every Note event in the triggered Step's Song will be scaled to values determined by the attack velocity of the note you play when triggering.

## Soft Buttons on the Edit Song: ARRANGE Page

**Add** adds another Step to the Arrangement. The new Step will be inserted as the Step number directly after the current Step. You would add a Step to place a new section into a Song Arrangement or to define another keyboard range for triggering.

**Delete** removes the current Step.

**SetRng** lets you quickly define the keyboard range for triggering the current Step from the MIDI keyboard. This dialog will appear:

Strike low key...

**Cancel**

“Strike low key...”

Play the lowest note of the range you wish to define, then:

“Strike high key...”

Play the highest note of the range you wish to define. At any time you may abort the range setting procedure by pressing the **Cancel** soft button, which returns to the ARRANGE page.

**Play** lets you audition your Arrangement from the ARRANGE page, starting playback from the currently selected Step.

**Stop** halts the playback of an Arrangement from the ARRANGE page.

**Done** exits the ARRANGE page and returns to the COMMON page.

## Selecting a Song for Playback

Select the Song parameter with the cursor buttons, then use any data entry method to scroll through the list of songs. Press the **Play** soft button, and the song will begin playing. Press the **Stop** soft button, and the song will stop and “rewind” to the beginning. If you press **Play** while the song is playing, the song will stop and the play pointer will revert to the Locate parameter, and will show your current location in the song. At this point you have two options. If you press the **Play** button again, the song will continue from its current location. If you press the **Stop** soft button, the song will return to 0:0.

The K2500 automatically selects programs for playback based on the MIDI channel(s) and the programs assigned to them at the time the song was recorded. When you start playback, the K2500 sends program changes, on all relevant channels, to its sound engine and to the MIDI Out port if the PChng parameter on the MIDI mode XMIT page is turned on.

If you want to use a different program for playback than the one originally recorded, you’ll need to defeat this automatic program selection. To do so, set the ProgLock parameter to a value of On. The ProgLock parameter is found on the MIDI mode CHANLS page; you can set it independently for each of the 16 MIDI channels. When a channel’s ProgLock parameter is set to a value of On, the K2500’s sound engine will ignore all program changes it receives on that channel, either via MIDI or from the K2500’s front panel.

When you’ve set the ProgLock for each of the channels used in the song, you can go to Program mode, select the MIDI channels used by the song, and assign the programs you want to use. When you return to Song mode and play the song, the automatic program changes are defeated, and the song plays the programs you assigned.

## Effect Selection During Recording and Playback

When you're recording or playing back a song, the preset effect applied to the song is determined by the program assigned to the FX Channel. Only one effect can be applied at a time, even for multi-part songs using more than one program. Depending on the combination of values for the FX Mode and FX Channel parameters in Effects mode, the FX channel may automatically track the current MIDI channel. In this case, if you change the current MIDI channel during playback, or during a MIDI recording, the current FX Channel, and consequently the current effect, will change also, which might not suit your needs. Consequently, there's a way to force the FX channel to remain constant during playback or MIDI recording, even if you move to another mode during the playback or MIDI recording and change the current MIDI channel.

This is done by setting the FX Mode parameter to a value of Auto, and the FX Channel parameter to a value of Current—which is the most generally useful combination of settings for these parameters. In this case, when you're in Song mode—or when you move to another mode during playback or MIDI recording—the FX Channel parameter is temporarily forced to a value of 1. Therefore the effect is determined by the program assigned to MIDI channel 1, and will not change, even if you change the current MIDI channel during playback or MIDI recording.

Of course, changing MIDI channels during playback or MIDI recording will not change the FX channel if the FX Channel parameter is set to a value from 1 to 16, or if the FX Mode parameter is set to a value of Master.

## Synchronizing Songs

The K2500 has an internal MIDI clock, which is always running at a speed set by the Tempo parameter. When you're in Song mode and the Clock parameter is set to a value of Internal, songs will sync to the K2500's internal clock. At this setting, the clock signal is sent to the K2500's MIDI Out port. This is standard MIDI Sync, and any device that accepts MIDI Sync will play in sync with the K2500.

If you set the Clock parameter to External, the K2500 expects to receive MIDI Sync at its MIDI In port. When you play back a song, pressing the **Play** soft button will cue the K2500 to wait for the sync signal. It will begin playing when it receives the signal, or when you press the **Record** button again. If no sync signal is received, the K2500 uses its internal clock at the tempo of the current song.

If you're using Song mode to capture a sequence that you've recorded on an external sequencer, you'll want to consider the Clock parameter's setting before you record via MIDI. If you have the Clock parameter set to External, the K2500 will follow the clock of your external sequencer. As a result, the notes in the song you create will fall regularly on the beats (unless your externally recorded sequence uses a time signature other than 4/4). This will make it easier for you to find the notes in the Song Editor. If you set the Clock parameter to Internal, the notes in the song will not necessarily align with the beats of the measures in the song, but the song will play back exactly as you recorded it on the external sequencer—including tempo changes you may have incorporated into the externally recorded sequence. If the Clock parameter is set to External, tempo changes will not carry over to the K2500, and will not be heard when you play back the song.

Finally, keep in mind that when the Clock parameter is set to External, programs that use one or more of the Clock control sources (see Chapter 6 of the *Reference Guide*) will sync to the external MIDI signals. If no external sync signal is received, the Clock control sources are disabled.

## Memory Limits

While there's no actual time limit to the length of the songs you record, their size is limited to 64K (or to the maximum amount of available free RAM space you have, if it's less than 64K). However, you can create longer songs by recording each section as a separate song, then putting it together with the Arrange feature. If you run out of RAM space while recording a song, the recorder stops and prompts you to save the song. It's a good idea to check your free RAM space before you begin recording a song, and to check the "Used" readout as you record. If you've used all the available RAM for recording, you may find that when you go to the Song Editor to delete a song or edit its tempo, the K2500 tells you that there's not enough memory to edit. In this case you won't be able to edit any object greater than 4K in size. Objects smaller than 4K can still be edited, because the K2500 always reserves a minimum of 4K of RAM.

If you want to delete a song and the K2500 won't let you enter the Song Editor, select one of the Metronome songs. Since the metronome songs are smaller than 4K, you'll be able to enter the Song Editor. Press the **Delete** soft button, then use the Alpha Wheel to select the program you want to delete. Press the **Delete** button again, and the song will be erased, freeing up enough RAM to edit other songs.

## Loading Songs From Disk

If you have a Type 0 or Type 1 MIDI sequence file stored on an MS-DOS disk (720K or 1.4 M), you can load it into one of the RAM banks, and the K2500 will be able to play it from Song mode.

## Recording Multi-timbral Sequences via MIDI

You can record sequences from an external MIDI device using Song mode. Program numbers and MIDI channel assignments of multi-timbral sequences are recorded with the notes. To record via MIDI, connect the MIDI Out port of your sequencer to the K2500's MIDI In port. Select Song mode, and set the Clock parameter to External. This will sync the K2500 with the MIDI clock of the external sequencer.

You will probably want to set the Local Keyboard Channel parameter to None when recording from an external sequencer, since the rechannelizing effect of that parameter could have unintended results.

To record all your tracks in one pass, set the RecTrk parameter to Mult and make sure that for each channel of information on your source sequence, you have a track enabled to record and a unique channel assigned to that track. (The default setting of all tracks enabled to record on channels 1-16 will always work.)

You can also record individual tracks from your source sequence by setting the RecTrk parameter to a specific track. The K2500 will record only information coming in on the channel that the RecTrk parameter is set to.

Press the **Record** button, and the K2500 will wait for the first clock start from the sequencer. Make sure that your sequencer is set to send MIDI clock signals, and start the sequence. The K2500 will begin recording when it receives the first MIDI clock start from the sequencer. When the sequencer has finished its playback, press the **Stop** soft button, and the K2500 will stop recording and ask if you want to save the song.

## Chapter 13

# Disk Mode

Disk mode lets you load and save files of objects between the K2500 and the outside world, through the K2500's floppy drive or SCSI ports. The floppy drive accepts MS-DOS format double-sided double-density (DSDD—720K) and high-density (HD—1.4 M) floppy disks. It will also accept single-sided double-density disks (SSDD—360K) if they're formatted as double-sided. To save data from the K2500 to a floppy disk, the disk must be formatted (initialized) by the K2500 or by an IBM®-compatible computer running MS-DOS® software.

Most SCSI (Small Computer System Interface) devices will operate with the K2500. The K2500's SCSI ports require 25-pin SCSI connectors, and will work with hard drives up to 2 gigabytes in size. The most common use for the SCSI ports is to connect a hard disk or chain of hard disks for storing samples and other objects. You can also connect a CD-ROM drive for reading files to the K2500. The K2500 will treat a CD-ROM drive like any other SCSI device (except that you can't save files to it).

Disk Mode in the K2500 allows flexibility to organize disk files and their contents. Many powerful operations are included that can save a lot of time by allowing you to easily specify exactly what you want to load or save. Examples of this range from organizing related files into directories, to loading macro lists of files or selected objects from multiple SCSI drives, to setting up programs to automatically link with samples off of a CD-ROM.

Lastly, the K2500 offers MS-DOS file system compatibility, sample transfer using the standard audio file formats "AIFF" and "WAVE", and saving of song files as MIDI Type 0 format. Furthermore, the K2500 can read song files in MIDI Type 1 format.

## Disk Mode Page

To enter Disk mode, press the Disk button, and the Disk mode page will appear:

```

DiskMode      Samples:65535K  Memory:747K
Path = \DRUMS\
(Macro on)
CurrentDisk:SCSI 4          Startup:Off
                           Library:Off

Direct Access, 121MB
TAXMOR XL3-1001 1.07
<more> Load Save Macro Delete more>

```

As usual, the current mode is displayed on the top line. At the middle of this line, the amount of available sample RAM is shown (if you have added optional sample RAM; if you haven't, this part of the top line will be blank). To the right of the top line you see the amount of memory available for storing all other RAM objects. Straight from the box, this number should be around 240K. If you have added the optional P/RAM extension kit, the number will be about 1260K. The P/RAM kit enables you to store hundreds of additional programs, songs, and any other objects. If you do a lot of programming or sequencing, the P/RAM option is a good idea.

In the center of the page is a line indicating the currently selected device. Select different devices using any data entry method. You can select the floppy drive, or SCSI IDs 0–7. (If you connect an Apple Macintosh® personal computer, don't select SCSI ID 7, since that's the SCSI ID of the Mac, which can't be changed. All SCSI devices connected in a network must have different SCSI IDs in order for the network to function). When you want to communicate with any of the SCSI storage devices in your network, set the Current disk parameter to the value

that matches the SCSI ID of the device you want to address. That is, the K2500 will interact with the SCSI storage device whose SCSI ID matches the value of the Current Disk parameter. If you choose a value of Floppy, the K2500 will communicate with its floppy disk drive.

The manual for your SCSI disk should tell you its SCSI ID. Most newer SCSI disks show their SCSI IDs on their rear panels, and many have adjustable SCSI IDs.

The currently selected device will be read from or written to when you load, save, rename, or delete files. Use the soft buttons to start any of these operations. Refer to "Soft Buttons on the Disk Mode Page" on page 13-5 for complete information.

### **Formatting a Floppy Disk**

Floppy disks must be formatted (initialized) to the Kurzweil format before they can be used with the K2500. The K2500 uses the MS-DOS format, so floppies formatted on DOS computers should work with the K2500. To format a disk on the K2500, insert a floppy disk in the K2500's floppy drive. Make sure the disk is unlocked (the sliding switch on the back of the disk is set so you can't see through the hole on that side of the disk).

Press the DISK button to enter Disk mode. Make sure the Current disk parameter says "Floppy," so you don't accidentally format any SCSI devices you might have connected! Press the soft button labeled Format. The K2500 will ask you if you want to format, and a pair of Yes/No soft buttons will appear. Press the **Yes** soft button, and three soft buttons will appear, letting you choose between formatting the floppy as a 720K (double-density) or 1.4M (high-density) disk. Press the appropriate soft button, or press the Cancel soft button.

If the floppy you inserted is double-density and you press the 1.4M soft button, the format procedure will fail. This is also true if the disk is high-density and you press the 720K soft button. If the floppy is single-sided (SSDD), you can press the 720K soft button, and the floppy will be formatted as a double-sided disk. (The only difference between single- and double-sided floppies is that both sides of a double-sided floppy have been tested by the manufacturer.)

Once you select between the 720K or 1.4M format, the K2500 will remind you that formatting will erase the floppy, and will give you two more chances to cancel the formatting procedure—we want to make sure you don't accidentally erase any disks. Press the **Yes** soft button to continue formatting. When formatting begins, the display will tell you that the disk is being formatted. You'll hear the disk drive turning, and the disk drive LED will light.

Formatting a 1.4M floppy disk takes just under three minutes, including the automatic verification. When formatting is finished, the display returns to the Disk mode page.

### **Connecting a SCSI Device**

It's easy to connect SCSI devices to the K2500's SCSI ports. Using a SCSI device will give you off-line storage, and can speed up your loading and saving operations considerably.

You'll need a SCSI cable with a 25-pin SCSI connector on the end to be connected to the K2500. If your SCSI device does not have a 25-pin connector at one end, you can find SCSI cables like these at any personal computer store. Connect the 25-pin end of the cable to either of the K2500's SCSI ports, and the other end to your SCSI device. Two SCSI ports are provided to enable you to chain SCSI devices together. Before you start connecting cables, however, please read the next section carefully. We've also included important information about SCSI in Chapter 10 of the *Reference Guide*.



## SCSI Termination

Simply put, SCSI termination prevents the electrical signals used by SCSI devices from being reflected from unconnected SCSI ports, and possibly disrupting the data stream. The rule for SCSI termination is that the two SCSI devices on the ends of a chain of SCSI devices must be terminated, and all devices in between, however many, must be unterminated. Newer SCSI devices usually make it easy to enable or disable their termination settings. Older SCSI devices may require an external terminator to be installed. These are available at all personal computer stores. Make sure you get the right size for your device (25-pin or 50-pin).

It's impossible to describe all the possible configurations of SCSI devices, so we'll provide you with a few general guidelines that will cover the requirements for most SCSI systems. If you're chaining large numbers of SCSI devices together, you may have to do a little juggling, but chances are you'll already have some experience with SCSI termination.

First of all, it's *very* important that you terminate your SCSI system properly. Improper termination can result in lost data, can interfere with the operation of your SCSI devices, and over the long term, can damage them.

If your SCSI system includes an Apple Macintosh® personal computer, you'll need to be sure that it is internally terminated. This is because the Mac has only one SCSI port, and therefore is necessarily going to be at one end of your SCSI chain. If your Mac has an internal hard disk, you can go ahead and hook it up, since it's sure to be internally terminated. Some Macs without internal hard disks are *not* terminated, so this may cause a problem. If your Mac has no internal hard disk, and you're not sure whether it's internally terminated, you should call your local Apple dealer for confirmation.

If your SCSI system includes only the K2500 and an internally terminated Mac, you're all set. Just connect the Mac's SCSI port to either of the K2500's SCSI ports, and you're ready to roll. The K2500 is internally terminated, so in this case, you've satisfied the requirement of terminating both ends of the chain.

If you have an internally terminated Mac, a K2500 and an external hard disk with *two* SCSI ports, setting up is also painless. Connect the Mac's SCSI port to one of the hard disk's SCSI ports, and the K2500's SCSI port to the hard disk's other SCSI port. Make sure the hard disk is not terminated, since it's in the middle of the chain. In this configuration (with a terminated Mac at one end and the K2500 at the other), you can chain up to six hard disks between them. Make sure they're all unterminated, and don't forget to set each disk's SCSI ID to a different value.

### Disabling the K2500's SCSI Termination

If you have a Mac, a K2500 and an external hard disk with only *one* SCSI port, you'll need to have your K2500 serviced. The only way to connect a system of this sort is to have the K2500 in the middle (with one SCSI port connected to the Mac, and the other to the hard disk). In this case, you'll need to have your Kurzweil/Young Chang dealer or service center remove the K2500's internal termination. This is very easy for a qualified technician to do, and shouldn't take more than a half hour or so. Late model K2500s have a "SCSI Termination Enable/Disable" switch on the back panel that lets you change the termination status of your K2500 without opening the unit.

If you install an internal hard disk in your K2500, the technician who does the installation should remove the K2500's termination as part of the installation. This will not necessarily enable you to connect the K2500 in the middle of a SCSI chain, however, since the internal hard disk has its own termination. With an internal hard disk, the K2500 should be installed at the end of a SCSI chain. The internal hard disk's terminator will provide the termination for itself and the K2500. You can remove the termination from the internal hard disk as well, if you want to connect your K2500 in the middle of a SCSI chain, though we do not recommend this.

If you're planning to buy an external SCSI hard disk to use with your K2500, it's a good idea to buy one with two SCSI ports. Most new hard disks have two ports, and can be terminated or unterminated relatively easily. This gives you added flexibility, since you can install it at the end of a chain, leaving its termination in place, or in the middle of a chain, using both its SCSI ports, and removing its termination.

When your SCSI device is connected, you can select it with the Current disk parameter on the Disk mode page. Use any data entry method to select the SCSI ID that matches the SCSI ID of your SCSI device. If you're using the numeric keypad to select the device, enter "9" to select the floppy. Newer SCSI devices usually have an external switch for setting their IDs. Older units may not have these; check your device's owner's manual for its SCSI ID.

## Using your K2500 in a SCSI System

### SCSI IDs

All devices in a chain of SCSI devices must have different SCSI IDs, including the K2500. The K2500's SCSI ID is set at 6 by default, and can be changed on the RECV page in MIDI mode. If your SCSI system includes a Mac, be sure not to use SCSI ID 7 for any of your other devices, since the Mac's SCSI ID is 7, and can't be changed.

Once you've made sure that all connected devices are set to different SCSI IDs, you should be able to select the devices, format them, and start loading and saving files.

### Formatting a SCSI Device

The procedure for formatting hard disks is essentially the same as with floppy disks, once the SCSI device is selected with the CurrentDisk parameter. The K2500 will recognize the disk as a SCSI disk, and will warn you that formatting will erase the contents of the disk. Compared with personal computers, the K2500's formatting time for SCSI disks is surprisingly short.

## Directories

A directory is a file on the disk that lets you group other files together as you might separate documents using folders in a file cabinet. You can create directories on K2500 Format SCSI and floppy disks. You can even create directories within directories, which are also called *subdirectories*.

Directories are very useful for organizing your sample, song, and program files. The K2500 provides many operations for setting up and managing the directories on your disks and the files within them.

### Path

The Path field shows the current directory on the current disk if it is a K2500 format disk. This field is displayed upon returning to the Disk Mode page after you have pressed one of the disk function soft buttons and viewed the file contents of a specific disk.

The current directory is always set to the top-level or root directory when the K2500 is powered on, or when the CurrentDisk parameter is changed. After this, the current directory is remembered by the K2500 as the most recent directory accessed by the disk functions.

The root directory is displayed as

**Path = \**

If you press the Load button and load in a file from a subdirectory called SOUNDS, the Path field will appear as

```
Path = \SOUNDS\
```

The slash character is also used as a directory separator, as in the following Path:

```
Path = \NEWTUNE\SAMPLES\DOGS\
```

This represents the subdirectory DOGS, that is located in the SAMPLES subdirectory of the NEWTUNE directory in the root directory. If the path is too long to fit on the top line of the screen, its display may be abbreviated. The maximum length of a path in the K2500 is 64 characters (including '\' characters).

## Startup

The Startup parameter determines what disk drive will be used for loading the power-up macro file BOOT.MAC (See "Creating a Startup File," page 13-65). If this is set to None, then the K2500 will power-up in a normal fashion. If this is set to a SCSI device or Floppy, when the K2500 is next powered on it will look for the BOOT.MAC file in the root directory of the specified disk, and load in each of the entries in the macro specified within.

This feature provides a very flexible way to automatically configure your K2500's memory contents whenever you turn the power on.

## Library

This feature works in conjunction with the Macro feature to provide a way to distribute Macro files that load data from removable media without having to know in advance the SCSI ID of the removable drive. A Macro file stores its references to drives by drive ID (SCSI ID or Floppy), or by either a "Library" or "Unspecified" designation (see the section on Macro files, page 13-38.) Typically, you would set the Library parameter to be the same as the SCSI ID of your CD-ROM drive, if you were loading Macro files from a Floppy disk or another SCSI drive that referenced CD-ROM files containing samples or keymaps.

Use of the Library feature is described in more detail later on in this guide.

## Disk Drive Information

Below the media type and disk capacity is displayed specific information about the current disk's manufacturer, model number and internal mechanism. The K2500 requests this information from a SCSI drive when the drive is selected by setting the Current Disk parameter. This information may be needed when determining if a given drive is compatible for SCSI operation with the K2500.

## Macro On Indicator

When the "(Macro on)" indication is displayed, it means that any load operation performed will be recorded in the K2500's macro table. The Macro feature is described starting on page 13-38 in this guide.

## Disk Mode Soft Buttons

Here is a brief description of each of disk mode's soft button:

<b>Load</b>	Loading files or loading files from the current disk into K2500 memory.
<b>Save</b>	Saving banks of objects, selected objects, or a macro as a K2500 file on the current disk.

<b>Macro</b>	Controls macro recording. Allows modifying entries from the current macro list.
<b>Delete</b>	Delete files from the current disk if it is a K2500 disk. See "Deleting files", below.
<b>Rename</b>	Change the file name of a file on a K2500 disk. See "Renaming Files", below.
<b>Move</b>	Change the location of a file from within one directory to another (on the same disk).
<b>Util</b>	Check the free space, find files, and view directory organization and sizes on the current disk.
<b>NewDir</b>	Create a new directory on K2500 disks.
<b>Backup</b>	Hierarchical file backup between drives.
<b>Copy</b>	Single file copy between drives.
<b>Sleep</b>	Send SCSI sleep command to the current disk. See "The Sleep Soft Button", below.
<b>Format</b>	Format the current disk as a K2500 disk.

### The Sleep Soft Button

Many SCSI devices will "sleep" when they've been idle for a few minutes. In other words, the disk will stop spinning, in order to save power and reduce wear. The K2500 lets you tell your SCSI devices to sleep. Just press the Sleep soft button, and if your devices have this feature, they will sleep. This is particularly useful in a quiet studio situation.

Any Disk mode operation will "wake" the device again. The K2500 will ask you to wait while the device's disk starts spinning. As soon as the disk is spinning at full speed, the K2500 will execute the operation you selected. Some SCSI devices automatically sleep when they power up. (A device of this type usually provides a way to override this feature; check its manual). Any Disk mode operation will wake a disk in this case, as well.

## File List Dialog

The file list dialog is displayed anytime you select a disk function to operate on one or more files on a drive (such as Load or Rename.) Here is a typical file list dialog, for the Load function:

```
Dir:\          Sel:0/3      Index: 1

File to load: BASSOON .KRZ  3456K
              MAY25      (dir)
              PERC      .KRZ  101K
Total: 3557K
Select  Root  Parent  Open  OK  Cancel
```

When you enter this dialog, the K2500 selects the current directory to display, and displays its files in an alphabetized scrolling list. If the current directory cannot be located (for instance, if a removable media disk has been changed), the default is to display the disk's root directory. The root directory will also be selected if the disk drive was just chosen by the CurrentDisk parameter on the Disk Mode page (remember that the current directory is always set to the top level when the CurrentDisk parameter is changed, or if the K2500 has just been powered on.)

The display for all drives (including floppy) shows the 3-character extension of all files in the directory (except directories themselves). Extensions are created when the file is saved by the K2500. You cannot modify the extensions on the K2500. This is because the K2500 uses the extensions to tell it what kind of data is stored within the file.

Directories created by the K2500 have up to 8-character names, with no extension. A directory can have an extension if it is created on an external computer (more on this later on).

Here is a list of extensions used by or accepted by the K2500:

<b>.AIF</b>	Audio Interchange File Format (AIFF)
<b>.KOS</b>	Kurzweil K2500 operating system file
<b>.KRZ</b>	Kurzweil K2000 format file
<b>.K25</b>	Kurzweil K2500 format file containing objects and/or sample data
<b>.MAC</b>	Kurzweil K2500 disk macro file
<b>.MID</b>	MIDI type 0 sequence file
<b>.WAV</b>	Microsoft RIFF WAVE format

When loading files, the K2500 will try to find out the type of file if the extension is not the same as is suggested above (with one exception, .MAC files.) The K2500 can create files with any of the above extensions.

The top line of the file list contains several items of information pertaining to the currently displayed directory contents. A typical information line looks like this:

```
Dir:..\HATS\      Sel:0/54      Index: 24
```

In the center of this line is an indicator of the number of files in the currently displayed directory. This number is grouped together with the number of selected files:

```
Sel:0/54
```

The first number means "No files selected" and the number after the slash means "Fifty-four files are in the directory". File selection is possible in several of the disk functions, more on this below. The total number of files also includes any subdirectories of the current directory, but not the files within the subdirectories.

On the left end of the top line of the file list page is the current directory, sometimes in an abbreviated form. If you are in the root directory, the display will read:

```
Dir: \           Sel:0/54      Index: 24
```

If you are in the directory \MONDAY, the display will read:

```
Dir:\MONDAY\     Sel:0/54      Index: 24
```

If you are in a directory that is more than one level down from the root directory, such as \FX2\GLASS\BREAKING, the display will read:

```
Dir:..\BREAKING\ Sel:0/54      Index: 24
```

The "..\" indicator tells you that you are more than one level down from the root directory.

## The File Index

On the right end of the top line is the Index field. This tells the position of the highlighted file relative from the beginning of the file list. The first entry in a file list is index 1.

```

Dir:\                               Sel:2/23       Index:    3
      HXM      .KRZ*    122K
      CHIME     .KRZ     42K
File to load:DOORS .KRZ* 3456K
      JUNE27      (dir)
      LONGSMPS    (dir)
Total:21034K
      FLUTE      .KRZ     .5K
Select Root Parent Open OK Cancel
  
```

Typing a number on the K2500 numeric keypad will automatically scroll the display to the corresponding entry in the file list. Typing an out of range value such as 999 is a quick shortcut to get to the end of the file list.

In addition to remembering the current directory on the most recently used drive, the K2500 now also remembers the index within the file list for the current directory. For instance, if you were to hit cancel on the above screen, and then either choose a different disk function (such as Rename), or even if you exited Disk Mode, went to Program Mode, and then later returned to Disk Mode to load a file off of the same drive, the file index would still show up on 3 (DOORS.KRZ). This index is remembered until a new drive is selected by scrolling the Current Disk parameter on the Disk Mode page.

There are exceptions to this however. For example, when a file is written to the drive using the Save function, the index will subsequently be set to the file that was just saved (see Save function). The index can also be explicitly set using the List and Find utilities (see below).

If there are no files in the current directory, then the index is 0:

```

Dir:\                               Sel:0/0       Index:    0

File to load:

Total: 0K
Select Root Parent Open OK Cancel
  
```

You will also see that there is a blank cursor bar, but no file names are displayed.

The maximum number of files that can be accessed within a single directory is 360. If you have more files than this amount in a single directory, then you will not be able to view the entries past index 360.

While in this dialog, pressing the CHAN/BANK buttons (to the left of the LCD display) will scroll the file list in either the forward or backward direction by "pages" of 5 entries. It is often easier to scroll the list this way when looking to see if a particular file is present in a directory.

## Soft Buttons in the File List Dialog

Use the **Select** soft button for multiple file selection in the Load, Delete, and Move functions. In the display below, there are two files selected, as indicated by the asterisk ("\*") after the .KRZ file extension. If OK were pressed for the following screen,

```

Dir:\                               Sel:2/23       Index:      3
AXM                                .KRZ          122K
CHIME                             .KRZ           42K
File to load:DOORS                 .KRZ*        3456K
JUNE27                            (dir)
LONGSMPS                          (dir)
Total:21034K                       FLUTE         KRZ*         .5K
Select  Root  Parent  Open  OK  Cancel

```

the files DOORS.KRZ and FLUTE.KRZ would be loaded.

The **Select** button will toggle the selection, meaning that if you press **Select** on a given file, the asterisk will go on if it is currently off, and vice-versa. Selecting can only be done for files, not for directories. You can select as many files as you wish using the **Select** button. There is also a way to select all files at once, or clear all file selections at once, using a double-press of the cursor buttons:

**Left/Right cursor double-press -> Select All Files**

**Up/Down cursor double-press -> Clear All Selections**

Pressing either the Left or Right cursor individually performs a separate function for finding directories, described below.

You can only select multiple files from within a single directory. Changing directories clears any selections.

Once you have selected one or more files, press **OK** to perform the disk function (in this example, Load) on all files marked with an asterisk, whether or not they are visible on the screen. If there are no files marked with an asterisk, the function will operate only on the highlighted entry, as in earlier versions of the K2500 system software.

The **Root** soft button returns the listing to the top-level or root directory. If the display is already at the root directory (as indicated by the Dir: \field on the top line of the display) the only effect of pressing **Root** will be reset the file index to 1 if there are files in the directory.

The **Parent** soft button is used to go up one level in the directory hierarchy. If the display is already at the root directory, this button will have no effect.

The **Open** soft button performs a different operation depending on the disk function and the type of file that is currently indexed in the display. In all disk functions, pressing **Open** on a subdirectory (indicated by a "(dir)" after the file name) will open that directory, read in and display the file list.

When a directory is first opened for viewing, the index will be set to one (the first file in the list). You might notice that the K2500 remembers the index the previous directory you were in before you pressed **Open**, so that if you return to that directory by pressing **Parent**, you are put back where you were in the list. This index is only remembered for one level down, and therefore is useful when stepping through a list of subdirectories from a single directory level.

In the Load function, pressing **Open** for a standard .KRZ file will start the Load Object feature. This allows selected individual objects from the file to be loaded in to the K2500. If **Open** is pressed on a Macro file (.MAC extension), then individual file entries within a macro file can be selected for loading. Both of these features are explained in great detail below.

For all other functions, if **Open** is pressed when censored to a .KRZ or a .MAC file, the object file or the macro file will be opened for viewing. For instance, pressing **Open** on a .KRZ file

while in the Delete function will display the objects within the file in a scrollable list, however no delete action will be possible on the individual objects.

Pressing the **OK** soft button will cause the K2500 to proceed with the selected function. After pressing OK, there may be further dialogs such as bank specification (for the Load function), confirmation (for Delete), or name entry (for Rename). One exception to this is in the Load function, if the file list is scrolled to a directory (dir) entry, then the OK button will enter the directory, the same as if the Open button is pressed on a directory entry.

The **Cancel** soft button exits the file list dialog, completing the disk function with or without any operation taking place. The K2500 returns to the Disk Mode page. Pressing the Exit button will do the same thing as Cancel.

## Total

The total size of all the files in the directory is indicated at the bottom left of the file display above the soft buttons. This total represents only the disk space used by the files in the directory being viewed. The K2500 includes a free space utility that can be used find out how much space is being used on the current drive all of the files on the disk. Also, there is a List utility that can be used to calculate the size of all files within a selected directory sub-tree. These functions are described under "Disk Utilities" on page 13-56.

## Quick Scrolling to Subdirectories

It is sometimes difficult to locate a subdirectory entry in the file list for the current directory, if there are many files in the current directory. To make this easier, individually pressing either the Left or Right cursor buttons will set the file index to the next or previous directory (respectively) in the current directory list. The index will wrap around the beginning or end of the list, so that repeated presses of either cursor button will cycle through all of the subdirectories. If you have many subdirectories, you can scroll through them all very quickly using this method.

For example, given the following file list display,

```
CYMBALS      (dir)
DOGS         .KRZ      122K
DOORS        .KRZ*    3456K
E4PROG       .KRZ      10K
LONGSMPS     (dir)
LUTE         .KRZ*     .5K
```

Pressing the right cursor takes you two entries further to the next directory,

```
DOORS        .KRZ*    3456K
E4PROG       .KRZ      10K
LONGSMPS     (dir)
LUTE         .KRZ*     .5K
MOON         .KRZ*    3456K
TRIANGLE     .KRZ      10K
```

or, pressing the left cursor takes you two entries back to the previous directory



```

ALTO          (dir)
ROOBAMS      .KRZ*   .5K
CYMBALS      (dir)
DOGS         .KRZ    122K
DOORS        .KRZ*   3456K
E4PROG       .KRZ    10K

```

## Creating Directories

As stated above, you can create directories for organizing your K2500 files, whether you are using SCSI or Floppy disks. No reformatting is necessary to add directories to disks that were formatted with a K2000.

Directories appear in the normal file list with the indicator "**(dir)**" to the right of the directory name.

There are two ways to create new directories.

First, the **NewDir** button can be pressed directly from the Disk Mode page, or second, a new directory can be created and entered in one step during the Save operation, by pressing **Save** then **NewDir**.

### Creating a Directory with NewDir

When you press **NewDir**, you will first name the directory that you are about to make, for instance:

```
Directory name: THINGS_
```

```
Delete Insert >>End Choose OK Cancel
```

Pressing **>>End** will take the cursor to the last character in the name. The **Choose** button allows you to grab a file name from the current disk (this is discussed later, in the section titled "Saving Files".) Otherwise, the name will default to either "NEWFILE" after a power-up, or the name will be that of the most recent file saved or loaded. Once you choose a name to start with (or the default), you can edit the name using the left and right cursor buttons, the **Delete** and **Insert** soft buttons, and the **>>End** soft button.

After you have chosen the directory name and pressed OK, you have the choice of where (in what directory) to put the new directory you are creating.

```
Use current directory for THINGS?
(Path = \)
```

```
Change OK Cancel
```

Pressing **OK** will select the default path, which is the current directory. Pressing **Change** will allow you to view the disk, traversing its directories, until you find the one in which you want to create the new subdirectory, in this case "THINGS". (see Change Directory Dialog, below)

## Creating a Directory with the Save and NewDir Soft Buttons

As a convenience when saving files to a directory, you can press **Save** from the Disk Mode page and then press **NewDir** at the "Save what:" dialog. After you have created the directory in this way, the current directory is automatically switched to while you are still in the save dialog. See the section on saving files for more information.

When you have confirmed your choice of what directory to create a new sub-directory in, the K2500 will create the named directory on your drive, and print a brief message such as:

```
Created directory /THINGS
```

If the new directory function was created by pressing the NewDir button on the DiskMode page, the K2500 returns to the Disk Mode page. Otherwise, if NewDir was pressed from the Save dialog, the K2500 returns to the "Save what:" dialog, having set the current directory to be the one just created.

## The Directory Selection Dialog

When making a new directory, as well as in many of the disk functions, you will be presented with the opportunity to change the current directory, or the default directory for a disk operation. A good example is the "Use current directory for THINGS? (Path = \)" screen above. If you press "Change" you will see a familiar file list dialog, with a few changes, through which you can select any directory on the disk. The display looks like this:

```
Dir:\                               Sel:0/23           Index: 7
                                MELLOTRN .KRZ      122K
                                N123AB   .KRZ       42K
Select directory:003HNS .KRZ    3456K
                                STRINGS   (dir)
                                T1         (dir)
Total:21034K  UNDULATE .KRZ    .5K
Root  Parent  Open                               Current  Exit
```

When you enter this dialog, you will be in whatever directory was displayed as the default. From here you can go into other directories by using the soft buttons Root, Parent, and Open. Notice that there is no "Select" button. This is because the purpose of this dialog is to choose a single directory as opposed to selecting multiple files. However, the Root, Parent, and Open buttons function exactly as described above (for the file list dialog). The Sel: parameter (on the top line) is useful as an indication of the number of entries in the directory you are looking at.

If you are scrolled to a directory entry, there is one additional soft button displayed, "SetDir". Notice the "Current" button moves over one button to the left:

```

Dir:\                               Sel:0/23       Index:    8
                                N123AB      .KRZ      42K
                                OCEANS      .KRZ     3456K
Select directory:STRINGS          (dir)
                                T1          (dir)
                                UNDULATE    .KRZ      .5K
                                ZORK        .KRZ      .5K
Total:21034K
Root  Parent  Open  Current  SetDir  Exit

```

There are two ways that you can select a directory in this dialog.

**Pressing "Current"** - This selects the directory you are currently in (whose file list you are viewing), as specified in the Dir: parameter on the top line of the display. For instance if you wished to select the directory STRINGS using the "Current" button, you would first press "Open" to display the contents of that directory, and then press "Current". If you instead wanted to choose the root directory, you would simply press "Current", since that is the directory you are viewing (notice the Dir: \ " at the top).

**Pressing "SetDir"** - This selects the directory you are scrolled to, such as STRINGS in the display above. This method is often quicker and more convenient than pressing "Open" followed by "Current," which does the same thing. The "SetDir" soft button is only present in the display when the scrollbar highlights a directory entry.

## Disk Mode Functions

Now that you are familiar with the basics of creating directories and moving around in the K2500 file system, it is time to discuss some of the features provided in the disk functions themselves.

### Loading Files

The **Load** button instructs the K2500 to copy a file, from the currently selected device, to the K2500's RAM. Press the **Load** button, and a list of files stored in the currently selected device will appear. Scroll through the list of files with the Alpha Wheel or Plus/Minus buttons, then press **OK**—or press **Cancel** to return to the Disk mode page.

When you press **OK**, the Bank dialog will appear, as described in "Load Function Dialog" on page 13-18, and you'll be asked to select the memory bank to load the file into. Scroll through the list of banks with the Alpha Wheel or Plus/Minus buttons until the desired memory bank is highlighted, then press **OK**. Or press **Cancel** to back up a page and select another file to load. Once you have selected a bank to which to load, you will be asked to choose a method for loading. The method you choose determines how the objects in the file will be ordered when loaded into the bank.

### Loading Individual Objects

Since files can contain over 3000 objects, it is often useful to load only a subset of the information contained in a K2500 file. Sometimes, this capability is necessary even to be able to load certain files, if the size of the file's samples or data is greater than the K2500's internal RAM size (in case you haven't gotten those extra SIMMs or SIPP's yet).

You can select individual objects or groups of objects (i.e. Samples, Programs, Keymaps, Effects, Songs) for loading from within a single K2500 file.

The Load Object feature is accessible from within the Load File dialog. To activate it, scroll the file list until you have highlighted the file that you wish to load objects from:

```

Dir:\                               Sel:0/6       Index:    3
                                BASSDRMS .KRZ      426K
                                HIHATS  .KRZ      788K
File to load:SAXES                 .KRZ      2510K
                                TOMS     .KRZ       301K
                                TOMS1    .KRZ     1400K
Total: 5037K                      XCYMB   .KRZ     1012K
Select  Root  Parent  Open  OK  Cancel

```

Press **Open** to begin the Load Object dialog. (Note: The file must have a .KRZ extension in order to load individual objects from it.) The K2500 then scans the file contents in order to present a list of all of the objects in the file. Sometimes this procedure can take a few moments, depending on how many objects are in the file. During this time, you will see the following display:

Reading file SAXES.KRZ [...]

```

Select  Next  Type  Multi  OK  Cancel

```

The soft buttons in the above display do not become active until the process of scanning is finished. When this happens, the K2500 will display a list of the file's objects, in the exact order that they are stored in the file:

```

Func:LOAD                               Sel:0/5       Index:    1

Sample  200  Tenor  Sax1                G 2S      250K
Sample  201  Tenor  Sax2                C 3S      179K
Sample  203  Tenor  Sax3                F#3S     198K
Keymap   200  Tenor  Sax                  176
Select  Next  Type  Multi  OK  Cancel

```

The objects in the list are usually grouped by type (Sample, Program, Keymap, etc.) The list can be scrolled using the Alpha wheel or the up/down cursors. The CHAN/BANK buttons on the front panel can be used for fast scrolling. The list will jump by five entries at a time, moving the entry on the bottom line to the top line.

NOTE: When scrolling through large numbers of objects (more than 100), the K2500 may sometimes pause for a few seconds if it needs to get more information from the disk file. When this happens, some gyrating dots will briefly appear in place of the Index value on the top line of the screen.

Each line in the scrollable list represents one object, and displays the object's type, ID, name, and size. Samples have additional information, the sample's root key and a stereo sample indicator:

Object Type	ID	Name	Root Key and Stereo Indicator (samples only)	Size
Sample	203	Tenor Sax3	F#3S	198K

The ID numbers are the same numbers that were used to reference the objects when the file was last saved by the K2500. These numbers will usually be different after the objects are loaded in, depending upon the bank (i.e. 200...299) and mode that is specified for loading. There is more information on these modes below, in the section "Load Function Dialog."

The Size field is interpreted differently for samples and non-samples. For non-samples (i.e. Songs, Programs, etc.), it shows the number of bytes used by the object in the file, and hence the amount of program memory that the object will occupy in the K2500. For samples, the size field shows the size of the all sample data associated with the object, and is displayed in kilobytes (K).

For samples, the letter 'S' after the root key indicates a stereo sample.

Due to display space constraints, if the sample's root key happens to be in the lowest MIDI octave range (i.e. C -1 through B -1), it will be displayed in a truncated form. For instance, if a sample's root key was set to G#-1, the display would read:

Sample	293	Tenor Sax	G#-	198K
--------	-----	-----------	-----	------

The status line at the top of the screen specifies the object function being performed,

Func:LUHU

the number of selected objects in the list followed by the total number of objects in the file,

Sel:0/5,

and the current list index:

Index:1

As with the file list, entering in a number from the numeric keypad will jump to the indexed entry, and typing in a large number like 9999 will go to the end of the list.

The soft buttons on this page are used for multiple selection of the objects in the list as well as for moving around the list when there are many items selected or listed. This same dialog is also used for many other functions in the K2500, namely for saving selected objects to disk and for several object utility functions that are described later.

Here is a brief description of each button's function, followed by a detailed explanation of its operation.

<b>Select</b>	Select or deselect an object.
<b>Next</b>	Jump to the next selected object.
<b>Type</b>	Jump to the next object of a different type.
<b>Multi</b>	Go to the Multiple Object Selector page.
<b>OK</b>	Tell K2500 to proceed to load the selected objects.
<b>Cancel</b>	Exit back to the File List Dialog.

Press the **Select** button to choose the highlighted object for loading. An asterisk (\*) is placed in between the object name and the object ID for any items that are selected. Deselect a selected

object by pressing **Select** again. The asterisk will disappear. The easiest way to choose objects for loading is to scroll the list and individually press **Select** on each object you want to load.

If you only want to select one object for loading, you need not select it with the **Select** button. Instead, pressing **OK** implicitly selects the highlighted object if there are no other objects selected. If there are objects selected, however, then the highlighted object will not be loaded unless it is selected.

This screen shows 3 samples selected for loading:

```

Func: LUHU      Sel: 3/20      Index:      4
Sample  201*Machine Snare      C 4S      100K
Sample  203*Big Bass Loop      C 4S      218K
Sample  304*Distorted Riff     C 4S      401K
Sample  305 EPiano1            D 2       24K
Sample  306 EPiano2            D 3       25K
Sample  313 EPiano3            D 4       15K
Select  Next  Type  Multi  OK  Cancel

```

The **Next** button will cause the index into the list to jump to the next selected object, forward in the list. When the end of the list is reached, the search will wrap around from the beginning. If you have more than one object selected, then if you repeatedly press **Next** you can easily cycle through all selected items. If there are no items selected, then this button doesn't do anything.

The **Type** button jumps to the next object of a different type from the one that is currently highlighted. This is a convenient way to find a particular type of object in the list. If you want to skip over the samples and the keymaps in an object list and jump right to the Programs, press **Type** about two or three times, stopping when you notice that a Program is highlighted.

Press the **Multi** button to enter the Multiple Object Selector (described on page 13-32.) This powerful utility can be used to control the selection or deselection of many objects, cross-referenced by object types and ranges of ID numbers.

When you are all done selecting objects to load, press **OK**. As stated above, if only one object is to be loaded, it is implicitly selected if it is the currently highlighted object and there are no other selected objects in the list.

**Cancel** returns to the file list dialog, highlighting the file you just opened. You can load the entire file after pressing **Cancel** by pressing **OK** when you return to the file list.

## Shortcuts when Loading Objects

### Select All/Deselect All

Selecting or deselecting all of the objects at once can be done with the same double-presses as described for the file list dialog, namely:

**Left/Right cursor double-press -> Select All Objects**

**Up/Down cursor double-press -> Clear All Selections**

If you want to load most but not all of the items from a file (for instance, if there happens to be a Master Table in the file that you don't want to load), it may be fastest to first select all objects using the Left/Right double-press, and then manually deselect any unwanted items.

## Viewing the Name Table

The Name Table is an object that appears in files that were created using the "Save dependents by name" feature (described below, in the section on the "Save" function). This object contains a list of dependent objects needed by the other objects in the file at the time the file was saved. There is more information about this later on, however it is worth mentioning here that a highlighted Name Table object's contents can be viewed by pressing either one of the Left or Right cursor buttons.

**Left cursor OR Right cursor -> View Name Table Contents**

## Loading Dependents of Selected Objects

When you press OK after selecting one or more objects, the K2500 may ask the question:

**Load dependent objects?**

**Yes**

**No**

This question will be asked if any of the selected objects could possibly have dependents associated with them in the file. (Remember, dependents are those objects needed by other objects; Samples are dependents of Keymaps, Effects and Keymaps are dependents of Programs, and so on. ) If this question is asked it does not necessarily mean that there really are dependents of the selected objects. The K2500 will not know whether there are dependent objects in the file until it begins to read in the selected objects, and determines what their dependents are.

Answering "Yes" to the question tells the K2500 to also load the dependents. You may wish to answer "No" if, for instance, you are simply loading in a Program or a Keymap as a template for use with other objects. You can also manually select only some of an object's dependents, and then answer "No" to "Load dependent objects?" to prevent other unwanted dependents from being loaded.

To summarize, it is not necessary to select any of the dependents of an object if you plan on loading all of the dependents in. As an example, for a file containing dozens of Programs, Keymaps, and Samples, you may choose to highlight a certain Program and press OK, and answer "Yes" to the "Load dependent objects?" question. The K2500 will do the rest, by only loading in the samples and keymaps that are needed by the selected program.

Similarly, if you were only to select certain keymaps from a file, and then answered "Yes" to "Load dependent objects", the K2500 would figure out exactly what samples need to be loaded in as dependents of the selected keymaps.

## Auditioning Samples from a Disk File

Often when working with files that contain samples it is helpful to be able to hear what the samples sound like before loading in all or part of the file. It is possible to audition samples in the file, from within the Load Object dialog.

To audition a sample, first scroll to the sample that you wish to hear. Then, press either the Left or Right cursor. The K2500 will load in up to 1 second of the sample, or less if the total sample time is shorter than one second. The audition starts from the very beginning of the sample data

(note that if the first second of data is silence then you won't hear very much when the sample is auditioned.) If the loop points fall within the first second of the stored sample data, they will be loaded in as well. The K2500 display will "blink" after the completion of loading the sample audition data. When the sample segment has been loaded, it can then be played back at its root key as well as transposed up and down the keyboard.

Once a sample has been auditioned, it remains active across the keyboard until another sample is auditioned. The audition function ends when either OK or Cancel are pressed.

There must be sufficient sample RAM in the K2500 to load in one second of the sound for auditioning. This amount varies according to the sample rate of the sample, but for most samples this will be less than 100K bytes. If the auditioned sample does not play, check that there is enough free sample memory in the K2500. It is also possible to see the following error if the K2500 object RAM is full or very near full:

**Not enough memory to audition**

## Loading Objects from Floppy Disk Files

Individual objects can be loaded from K2500 floppy disk files, with the caveat that some samples in a multiple-floppy disk file cannot be auditioned because they reside on more than one disk. These samples will still appear in the object list, but their size will be in parentheses meaning they cannot be loaded (either explicitly or as dependent objects) and cannot be auditioned.

## Load Function Dialog

### Bank Status Indicator

After you have chosen what you wish to load, you are presented with a dialog allowing you to determine what bank will be used to load in the file's data.

This indicator takes the form of an asterisk after the bank. If an asterisk is present after the bank (i.e. 400...499\*), it means that there are objects in the bank, whether they are RAM or ROM objects. ROM objects are in the 0's and 100's bank, unless you have an optional ROM block installed. In that case ROM objects are also stored in the 800's and/or 900's bank. If there is no asterisk on the line for a bank, it means the bank is empty.

In the following screen, there are user objects in the 200's and 400's bank, and possibly also in other banks that become visible when the selection is scrolled.

```
Load this file as: 200...299*
                  300...399
                  400...499*
                  500...599
                  OK Cancel
```

This indicator makes it easier to find an empty bank to use for loading, if needed.

### Loading Methods

Once you have pressed **OK** to decide on what bank to use, you will see this screen if the bank is empty:



```

Load this file as: 200...299*
                   300...399
                   400...499*
                   500...599
                   600...699
                   Append Fill Cancel

```

You will see the following screen if the bank contains any objects (in RAM or ROM):

```

Load this file as: 200...299*
                   300...399
                   400...499*
                   500...599
                   600...699
                   700...799
OvFill Overwrt Merge Append Fill Cancel

```

The soft buttons control the mode for loading and renumbering of objects from the file. The meaning of these soft buttons is:

- OvFill** First deletes all RAM objects in the selected bank, and then loads in objects using consecutive numbering.
- Overwrt** First deletes all RAM objects in the selected bank, and then loads in objects using the object ID numbers stored in the file.
- Merge** Preserve the object ID numbers stored in the file for the objects to be loaded, overwrite objects already in memory if necessary.
- Append** Try to use the object ID numbers stored in the file for the objects to be loaded. If an ID number is already in use, increment the ID number until a free slot is found.
- Fill** Ignore the object ID numbers stored in the file. Try to use consecutive numbering from the beginning of the selected bank. If an ID number is already in use, increment the ID number until a free slot is found.
- Cancel** Cancel the mode selection, and go back to choosing a bank. Scrolling to a different bank value will have the same effect as Cancel.

Typically, you will just want to use the **Fill** method. **Append**, **Merge**, and **Overwrt** try to preserve the numbers stored with the objects in the file, but this should only really be necessary if you depend on program numbers or effect numbers to be at a certain MIDI program change number. **OvFill** is like **Fill** except the selected bank (or Everything) is cleared out before loading.

**Overwrt** and **OvFill** operate in different ways after a selected bank has been filled up for a given object type (for example, after you have loaded in more than 100 programs into a bank). **Overwrt** will begin continue to preserve the objectIDs stored in the file, and will individually overwrite objects in the bank following the just filled bank. **OvFill** will not overwrite any further objects past the end of the selected bank; it instead skips over object IDs that are in use. Because of this difference, it can sometimes be faster to load in a file using **OvFill** rather than **Overwrt**. However, this only applies if the objects to be loaded would extend past the end of a selected bank.

Note that when loading into a specific bank (as opposed to loading as "Everything"), the object ID's in the file are used as follows: The "bank" digit is ignored, and the remainder of the number is used when the K2500 re-banks the object ID into the bank that you specify. For instance, if you save Program 453 into a file, and load it back into the 300's bank, the K2500 will use the number "53" when deciding upon a new object ID. If the 300's bank was previously empty, and the load mode is Append, then the Program will end up with ID 353.

For loading as "Everything", the ID number for an object stored in a file is taken literally, and not re-banked (except if Fill or OvFill mode is chosen, in which case the K2500 will use ID numbers starting from 200.)

Here is an example that compares all of the above load modes.

EXAMPLE: Starting with the following objects already stored in the K2500 internal RAM:

```
Program 200 Existing Pro91
Program 204 Existing Pro92
Program 205 Existing Pro93
Program 210 Existing Pro94
Program 211 Existing Pro95
```

Suppose you were to load file containing the following objects into the 200's bank:

```
Program 405 Electric Piano
Program 406 Stereo EPiano
Program 409 Weird Piano
Program 410 Wild Funk Piano
```

Here is what these programs would end up with as ID numbers when loaded in, depending on the load mode:

<u>Name</u>	<u>Append</u>	<u>Fill</u>	<u>Merge</u>	<u>Overwrt</u>	<u>OvFill</u>
Electric Piano	206	201	205	205	200
Stereo EPiano	207	202	206	206	201
Weird Piano	209	203	209	209	202
Wild Funk Piano	212	206	210	210	203

In the Merge case, the existing programs 205 and 210 would have been deleted. In the Overwrite case, all of the existing programs in the 200's would have been deleted.

## Multiple Selection of Files to Load

As stated previously, you can select multiple files for loading into the K2500 from within a single directory, in one operation. This is done from the file list dialog with the **Select** button.

After you have selected one or more files in this way, you will still choose a bank and mode to be used for the load process, just as with loading a single file. However, the dialog prompt will say "Load selected as":

```
Load selected as: 200...299*  
                  300...399  
                  400...499*  
                  500...599  
                  OK Cancel
```

If you selected any macro files (.MAC extension) from the directory, then once you have select the mode for loading, you will see the question:

```
Load macros as specified?
```

```
Yes No
```

The answer to this question instructs the K2500 that any macro files will have their macro entries loaded according to the bank and mode:

**Yes** specified in the macro entry.

**No** currently specified for this multiple file load. In other words, whatever you select for Bank and Mode will override the instructions for each entry in the macro.

At this point the files will begin to load. When all the files have been successfully loaded or the load process has been aborted, the K2500 returns to the Disk Mode page.

If there are any errors encountered during a multiple file load, such as running out of object RAM, you will be asked once if you wish to abort the load. In some cases, you may wish to continue loading. If you continue (and don't abort), the only way to abort will be to use a special procedure described in the next paragraph.

## Aborting a Multiple File Load

There is a way to abort the process of loading multiple files. Aborting can only be done "in between" files that are being loaded, and not during the load of any one file (short of powering off or soft-resetting the machine by pressing +/-, 0, and CLR simultaneously, but this is not recommended!).

Aborting a multiple file load is done by pressing and holding down either of the Plus (+) or the Minus (-) buttons that are located just below the Alpha wheel. This should be done at least one-half second before you anticipate the current file to finish loading, or else the K2500 will not sense that you wish to abort the load.

You will see the following question after the current file being loaded is completed:

Abort the load?

Yes

No

It may be a good idea to practice using this method of aborting a multiple file load, so that when the time comes that you accidentally select 100 files, you will remember how to abort the process. This same method (of holding the Plus or Minus buttons down) is also used to abort the Backup feature and the Macro file load feature.

If you run out of object or sample RAM, you will have one opportunity to abort the load as explained above. However, if you continue from that point you may end up seeing the same error message "Memory is full" for each file that you had selected. This can be a rather tedious process, however it is still possible to abort out of this by holding down the Plus or Minus button simultaneously while pressing "Yes" when you see the following question:

Abort this Partial load?

Yes

No

## More Load Function Enhancements

There are more features having to do with the Load function that are described later on in this guide (i.e. loading Macro files, loading AIFF files, etc.)

## Saving Files

The Save button starts the process of saving from the K2500 to the currently selected device. When you press the **Save** soft button on the Disk Mode page you will see the bank dialog:

```
Save selection: 200...299
                300...399
                400...499
                500...599
Export Macro Object NewDir Ok Cancel
```

The **Macro** soft button will only be present if Macro recording is turned on. (See page 13-38 for more information on Macros.)

You can save an entire bank of objects, or by pressing the **Object** soft button, select individual objects to be saved. If you choose to save using the bank method, all RAM objects within that

bank will be saved. (You cannot save ROM objects. If you wish to save a ROM object, such as a program, you must first save it internally as a RAM program.) If any objects within the selected bank have dependent RAM objects that exist in a different bank, you will be asked if you want to save dependent objects. See page 13-26 for more on saving dependent objects.

Use one of the data entry methods to select a bank to be saved. If you press the **Cancel** soft button, you'll return to the Disk mode page. After you've selected the bank, press **OK**. The following page will appear:

Save as: NEWFILE

Delete Insert >>End Choose OK Cancel

You can now name the file according to the naming procedures outlined in Chapter 5. You can enter up to eight characters. When you've entered a name, press "OK" to save the file as shown in the display, or press Cancel to return to the file dialog. When the file is saved, the K2500 adds an extension (.KRZ) to the filename. This enables the K2500 to recognize it as a Kurzweil file when it examines the disk's directory.

### Saving Master and Everything Files

Among your choices in the Bank dialog are Master files and Everything files. Master files consist primarily of the items on the Master mode page and the three MIDI mode pages. They also include information like marked pages, view settings, and MIDI channel and program assignment. In fact, saving Master files (or dumping them via SysEx) is a good way to configure your K2500 (or another K2500) to your performance or sequencing needs. For example, you might save different Master files with every sequence you create using an external sequencer. Then, when you load the Master file, you would have all the correct programs assigned to the appropriate MIDI channels.

Everything files consist of the Master file parameters and every other RAM object. Saving an Everything file will literally save everything in RAM, including samples, into a single file. These can be quite large, so if you don't have a hard disk, be sure to have a few preformatted floppy disks handy.

### Split Files

When you're saving memory banks, it's not uncommon to create files larger than 720K or even 1.4M—especially when you're saving RAM samples. If you're saving to floppy disks, the K2500 will create split files stored on multiple disks. A little advance preparation is necessary for this.

If the K2500 fills a floppy disk before it finishes saving a file, it will prompt you to insert a second disk in the floppy drive. This disk must be preformatted; the K2500 will not format disks in the middle of a save operation. When the second disk is inserted, the K2500 will continue to save. This process will be repeated until the entire file is saved. The K2500 marks these disks internally with a number that indicates the sequence in which they were saved.

When you're loading these split files, they must be inserted in the same sequence as they were saved. When the K2500 has loaded the contents of the first disk, it will prompt you to insert the second disk, and so on. Make sure to label your disks as soon as you save to them, so you'll keep them in the correct order.

## Soft Buttons in the "Save Selection" Dialog

The meaning of the soft buttons in the "Save selection:" dialog is as follows:

- Export** Save a sample or a song in an exported file format (i.e. AIFF, WAVE, MIDI Type 0). This feature is described in the section "Using Standard File Formats."
- Macro** Save entries from the current macro table as a Macro file (.MAC). This soft button is only displayed if macro file recording is on. This is described later in the section "Saving Macro Files."
- Object** Save selected objects from the K2500's RAM.
- NewDir** Create a new directory on the current disk, and return to this dialog afterwards. This is described previously in the section "Creating Directories."
- OK** Save all the objects from the highlighted bank (i.e. 200...299), and optionally also save dependent objects.
- Cancel** Exit from the Save function.

**Export**, **Macro**, and **NewDir** are all explained elsewhere in this guide. This section will describe the process of saving K2500 objects into K2500 format disk files.

## Saving Individual Objects

You can select any group of objects in the K2500's RAM for saving into a single file. To save individual objects, from the above dialog, press **Object**. The K2500 will display a scrollable list of all the objects in RAM, very similar to the display for the Load Object feature (described previously):

```

Func:SHVE      Sel:0/8      Index: 1

Sample 500 Lo Vocal      A 3S 250K
Sample 501 Hi Vocal      G 4S 179K
Keymap 500 VocalsMap      176
Program 500 Dry Vocals    270
Select Next Type Multi OK Cancel

```

The procedures for saving objects are essentially the same as the procedures described on page 13-13 for loading objects.

## Shortcuts when Saving Objects

### Select All/Deselect All

Selecting or deselecting all of the objects at once can be done with the following double-presses (two front-panel buttons simultaneously pressed):

- Left/Right cursor double-press -> Select All Objects**
- Up/Down cursor double-press -> Clear All Selections**

If you want to save most but not all of the items from a file (for instance, if there are some songs in RAM that you don't want to be saved in the file), it may be fastest to first select all objects using the Left/Right double-press, and then manually deselect any unwanted items.

## Viewing Selected Objects

When there are lots of objects selected, but they are scattered in the objects list, it can be helpful to be able to view a list of only the currently selected objects. Such a list is displayed by double-pressing the CHAN/BANK buttons to the left of the K2500's LCD display.

### Double-press of CHAN/BANK buttons -> View Selected Objects

For instance, if there were 10 objects selected, and you pressed both CHAN/BANK buttons simultaneously, the K2500 would show a list similar to this:

```

View Selected Objects 10/134
Program 300 Biggest Kit 7124
Program 301 Repercussions 7124
Song 300 Drum Groove 7 12092
Song 301 Drum Groove 8 24700
Song 421 Nasty Funk 3122
Song 500 Beethoven 1024
OK

```

The top line shows 10 objects selected out of 134 that are in RAM. If the number of selected objects is larger than the 6 objects that fit on one screen (as in this example), the list can be scrolled to view all of the information.

Note that this feature is not available in the Load Object dialog.

## Auditioning Objects in RAM

When deciding which individual objects to save, it can be difficult to know if you are selecting the correct ones. This is especially true if many objects have similar or identical names, or if the names of the objects are not descriptive enough to know what they are. The K2500 has a feature that allows auditioning of Samples, Keymaps, Programs, and Songs right from the Save Object dialog (as well as all of the other object utility dialogs that are discussed later.) To activate this feature, scroll to an object of an appropriate type to be auditioned, and press either the Left or Right cursor button. The screen will "blink", and the objects can now be heard as follows:

**Samples** play at their root key, as well as transposed across the keyboard. Stereo samples will play in stereo. Auditioning samples in this way is similar to listening to samples from the **MASTER->Sample** page. The samples are auditioned using a "hidden" program set up according to the parameters in **Program 199 Default Program**. This default program can be customized if needed by editing and saving a new program 199.

If you audition any sample objects, the last one that you audition will become the "preview" sample the next time you go to the **MASTER->Sample** page. This can be a quick way to edit the sample without having to edit a program and a keymap.

**Keymaps** are reproduced accurately, and are played according to the parameters in Program 199 Default Program. This default program in ROM is set up to have a 0% effects level (dry). Therefore, auditioning keymaps can be a very convenient way to hear them isolated from the effects.

**Programs** play exactly as they would if they were selected from the Program Mode page.

**Songs** start playing when either the Left or Right cursor button is pressed, and stop playing when either cursor is subsequently pressed. The most recent song that is

auditioned from this page become the Current Song (as seen on the **Song Mode** page).

Once auditioned, the above object types remain active on the keyboard until another object is auditioned, or **Cancel** is pressed. If a song is being auditioned, no other objects are auditioned until the song audition is stopped (by pressing one of the Left or Right cursor buttons).

## **Saving Dependent Objects**

When you save a file, you may see a prompt as part of the Save dialog that asks you whether you want to save dependent objects. A dependent object is simply an object that's associated with another object. The dependent object can be stored in a different memory bank, for example, a RAM sample with ID 301 that's used in a program with ID 402, or in the same bank as the file being saved. Rather than forcing you to save dependent objects separately and to keep track of them yourself, the K2500 gives you the option of automatically saving the dependent objects as part of the file you save. When you load the file again, the dependent objects will be loaded along with the objects to which they're attached.

There are a few things to keep in mind regarding dependent objects. First, it's not uncommon for RAM samples to be dependent objects, and they can take up quite a bit of memory. If you save a RAM sample as a dependent object when you save a bank of programs to disk, you may create files which won't fit onto a 720K or 1.4M floppy disk. Since the K2500 can't format floppy disks in the middle of a save operation, you should have spare formatted disks ready to go before you start saving. See the section called "Split Files" on page 13-23.

While the K2500 makes it easy for you to keep track of your dependent objects, you need to keep aware of what happens with dependent objects when saving to disk and reloading. Consider this example. Suppose you create 30 new programs, each of which uses a keymap containing four different RAM samples. If you save these programs to a disk file, and save dependent objects with them, you've created a file containing 30 programs and 120 dependent RAM samples. So far, so good. Suppose you then load that file into the 300s bank. The K2500 will load the 30 programs into the 300s bank just fine, but it will be able to load (at most) only the first 100 dependent objects to the 300s bank (each memory bank can hold a maximum of 100 objects of a given type). The remaining 20 dependent objects will be loaded into the 400s bank. If there are no objects of the same type in the 400s bank, there's no problem. But if there are objects of the same type in the 400s bank, some or all of them will be replaced by the newly loaded dependent objects.

The easiest way to prevent this is to make sure that you don't create more than 100 dependent objects attached to the other objects in a given memory bank. The easiest way to do *this* is to avoid creating dependent objects when possible, by saving objects with IDs in the same memory bank as the objects to which they're related. For example, if you create a program that uses RAM samples, and you save the program with ID 201, resaving the RAM samples used by that program with IDs in the 200s will prevent dependent objects from being created for that program. If you do this, you'll minimize the number of dependent objects you create, and you'll be unlikely to force dependent objects to be loaded into a higher-numbered memory bank when you load files.

Once you have selected objects for saving (either individually as just described or by bank selection), the K2500 will determine if any of the items chosen to save have any dependent objects in RAM that were not chosen. For instance, if you select a program to be saved and nothing else (using the Save Object feature), the program may have dependent effects, keymaps, and samples that are in RAM. Dependent objects that are in ROM (i.e., ROM samples or keymaps) do not get saved to disk.

You will see the following dialog displayed if there are any dependent objects in RAM of any objects that were selected for saving:



## Save dependent objects?

**Names** **Yes** **No**

Choosing **Yes** will cause any dependent objects to be saved in the file together with the selected objects. Choosing **No** means that unselected dependents will not be saved. The **Names** button creates a new kind of object to be stored in the file, called the Name Table.

### The Name Table

The Name Table is a list of any dependent objects that were not explicitly selected for saving in the file. Each entry in the Name Table contains the object type, object ID, and the name of a dependent object.

A file's Name Table is used by the K2500 at only one time, when the file is loaded in. At that time, the K2500 will search for dependent objects that were not saved in the file originally. The search matches dependent objects by name with objects that are already in RAM, and links them to the "parent" object. The Name Table data is then discarded when the file load is finished. This search feature is referred to as **Relink-by-Name**.

Relink-by-name offers new and very efficient ways of working with K2500 objects and disk files. Careful use of this feature can save you many megabytes of disk storage, as well as a lot of time spent working on music and production instead of waiting for sample data to be resaved.

Relink-by-Name allows you to save objects and their dependent objects separately (in multiple files) and be able to link them up later on by loading the files in the correct order. This can be a very efficient way of working with the K2500's many levels of dependent objects. The most common way in which Relink-by-Name speeds up development of sounds is when making small adjustments to a program that has as its dependents a large amount of sample data. you can separate the program and sample data, so that after changing a program parameter, only a file containing the program and a Name Table need be re-saved.

When loading in a file that contains a Name Table, the following rules should be observed in order for correct relinking to occur.

1. **Use unique names for dependent objects at every level.** As an example, this would mean that if you were going to be re-linking several samples from one file with a program and a keymap from another file, that each sample should have a different name. Otherwise, the dependent objects (the samples) will not get re-linked properly. This will create problems such as keymap ranges that don't play as they are supposed to.
2. **The dependents to be re-linked must already be loaded.** Otherwise they will not be found and re-linked when the file containing the parent objects is loaded in. This constraint on the order of file loading can be made easier to work with by using the Macro file feature (described later). You can construct a Macro file to automatically load in the dependents files and the parent files in the correct order, making sure that any files containing dependents are loaded first. An alternative to loading the files with a Macro would be to save the dependent and parent files in the same disk directory with similar file names such that they will appear consecutively in the alphabetized file list. Once you have done this, it is easy to select both files for loading in the correct order.

These rules may appear complicated at first, but they will seem natural once you have worked out a few examples with your own files.

The **Search Algorithm** used for relinking dependent objects to their parent objects during loading is as follows:

**The search for a dependent object (whose name matches that of an entry in the name table) begins at the beginning of the bank that is specified for loading the parent file.** All possible IDs are then consecutively searched. When the last ID of the 900's bank has been searched (typically 999), the search will wrap around to ID 1 up until the end of previous bank to the specified bank. The search stops once a dependent with a matching name has been found and relinked.

For example, if a file containing a one-layer program is loaded into the "400's" bank, and the file includes a Name Table that lists the layer's keymap by name, then the K2500 will begin to look through all possible keymap IDs starting at 400, until ID 999. The search then continues from ID 1, stopping at ID 399. If the search does not successfully find a match, the dependent will be unresolved, and in this example the program would have an "*object id* Not found" indication in its Keymap parameter, where the *object id* is the value that was stored in the file.

The search is done in this "circular" manner so as to allow you to direct which dependent objects get re-linked. This may be necessary if you end up with multiple copies of dependent objects with the same name; you can differentiate between them by loading the parent file into a specific bank that is the same bank or "before" the bank containing the objects you wish to re-link to. Note that this can only be taken so far, since it would be impossible for the K2500 to differentiate between objects with the same name within the **same** bank.

The re-linking process happens in the background, without any notification or error messages if items cannot be relinked.

## Working with Relink-by-Name

Here are a couple of more in-depth examples that can show how Relink-by-Name works in a practical situation.

Consider that your K2500's RAM contains the following one-layer program and also its dependent keymap and samples (the techniques used in this example could well apply no matter how many programs with any number of layers you want to save):

Program:           **Program 317 Steinwave Piano**

Keymap:           **Keymap 300 Steinwave Piano**

Samples:           **Sample 300 StwaveG1 ..... Sample 310 StwaveC7**

In this case you might wish to save the samples and the keymap in one file, and the Program in another file. So, from the Save Object dialog you could first select all the Samples 300-310, and Keymap 300, for saving into a file, let's say **STWAVE1.KRZ**.

You would then return to the Save Object dialog and save just Program 317 in a separate file in the same directory, let's say **STWAVE2.KRZ**.....only this time, you will be asked the "Save dependent objects" question pictured above. Answer this by pressing **Names**.

After saving, the file STWAVE2.KRZ will contain two objects in it, Program 317 and a Name Table. You can easily verify this by going to the Load function (or any other disk function) and

pressing "Open" on the file just saved (which should come up already highlighted). The display of objects for the file will look like this:

```
Func:LOHU      Sel:0/2      Index: 1
```

```
Table   36 Names      334
Program 317 Steinwave Piano      274
```

```
Select  Next  Type  Multi  OK  Cancel
```

The Name Table will always be the first object in the list. You can verify the exact contents of the name table by using the "View Name Table" shortcut (previously described in the Load Object section); make sure the Name Table object is highlighted, and **press either the Left or Right cursor button** on the front-panel (as if you were "auditioning" the Name Table object.) You would then see the following informational display:

```
Name Table Contents
Keymap   300 Steinwave Piano
Sample   300 StwaveG1
Sample   301 StwaveD2
Sample   302 StwaveB2
Sample   303 StwaveE3
Sample   304 StwaveB3
Sample   305 StwaveG4
```

```
OK
```

The **Name Table Contents** display shows what would have been saved in the file had you answered Yes to "Save dependent objects?" instead of answering by pressing Names. More importantly, it allows you to see what objects will need to be already loaded in to the K2500 before loading this file.

The object IDs shown in the table are the same numbers that those dependent objects used at the time this file was saved. (The ID numbers are necessary in order for Relink-by-Name to function, since they are the "link" between the higher level objects and the names of the dependents.)

An important thing to notice about this particular Name Table is that the Sample names are not needed by the K2500 for relinking purposes. In fact, the only information necessary for relinking the dependent objects of this file is the Keymap object. The reason for this is that when this file containing the program is loaded, all of these dependent objects should already have been loaded, and the Keymap should already be correctly linked to the Samples. Although the Samples' names are redundant from the K2500's point of view, they are included for free, so to speak, and you may find them very helpful if you ever need to know exactly what the dependents of this file were intended to be.

The Name Table Contents List is scrollable if there is object data past the 7 lines that fit on the screen.

Now that the two files STWAVE1.KRZ and STWAVE2.KRZ have been created using the Name Table, they can be loaded back in and correctly relinked. The files can be loaded into any bank - they do not need to go back into the bank they were originally in - since the STWAVE2.KRZ file will search through all the banks to find the objects by name in order to relink them. In fact, if you were to immediately reload just the file containing the Program (STWAVE2.KRZ), into any bank, you would find that it was automatically relinked to the correct Keymap, since the Keymaps and samples are currently in memory.

Furthermore, you could edit the program and create more variations of it that reference the "Steinwave Piano" keymap, add ROM layers, and / or effects programs if desired, and resave all of the programs (and any effects) to the same or a new file (remember to press **Names** when you are asked "Save dependent objects?". ) You never have to resave the file STWAVE1.KRZ that contains the keymap and samples, if all you have done is edited the programs or added more of them. This can be a tremendous time-saver.

If the Keymap and Sample files are found on a CD-ROM disk, then using Relink-by-Name is not only a time-saver, but a disk space saver as well. If you like the samples and keymaps from a CD-ROM file, there is no need to duplicate the sample data on your own writable hard drive. Instead, all you have to do is save a program file in the above manner, and then make sure the CD-ROM file is loaded first before you load the program file.

If you needed to add some sample data to the file (for instance, you want to add a root to the keymap or process and reloop a sample from the CD-ROM), you can do this by explicitly selecting the new sample data and the keymap for saving along with the program and the Name Table. Then, the new sample would not be listed in the Name Table (it would be in the same file as the Name Table), and the Keymap would be relinked to all of the samples by name instead of the program being relinked to the keymap (as before). What you put in the different files is up to you, and there is no limit to where you can break up the objects in one file or another. The main thing to be aware of are the two rules for Relink-by-name mentioned above:

1. **Always use unique names for like objects types.** (NOTE: In cases where duplicate names do exist in different banks, load the file that contains the name table into a bank equal to or preceding the bank in which the file with dependent objects was loaded.)
2. **Files containing dependent objects must be loaded first.**

As you will see later, you can create a **Macro file** that will automatically load both of the files in the correct order, no matter what drives they are on or what disk directories they are in. By using Macro files in this way, you can avoid having to explicitly load 2 files and remember the correct order each time.

You can also use the **Multiple Object Selector** (described in the next section) to help in the process of identifying dependent objects and parent objects that you want to place into separate files. For instance you could easily select all dependent Keymaps and Samples of any group of programs, to create a "dependents" file. Then, you could quickly select the programs and any other objects that you wanted to be relinked later on, and save them in another file.

Here is another practical example using Songs (sequences). Suppose you have loaded several files into your machine, such that you now have all your favorite instruments on-board in the K2500's RAM. Then, you make a bunch of songs using a combination of ROM programs and the RAM programs you loaded in.

The dependent object structure of the songs would look something like this:

Songs:	<b>Song 400 Wild Jam ,Song 401 Memphis Groove</b>
Programs:	<b>Program 600 Drawbarz   Program 245 FendJazzBass Program 231 Funky GTR   Program 400 ObieWarble Pad Program 103 3 Layer Kit (from ROM)</b>
Effects:	ROM effects
Keymaps, Samples:	Lots of them...

In this case you might want to save all of the songs in one file, and be able to automatically relink the dependent programs used by the song tracks. All of the programs are presumably already saved in separate files. The only file that needs to be created is one that contains all of

the Song objects, plus a Name Table. Once again, this is done by selecting the Songs from the Save Object dialog, and answering Names to "Save dependent objects?". The contents of this file can then be displayed by pressing Open (as was done for the previous example).

```

Func:LOAD    Sel:0/2    Index:    1

Table    36 Names    700
Song     400 Wild Jam    12114
Song     400 Memphis Groove    34002

Select  Next  Type  Multi  OK  Cancel

```

Also as shown in the previous example, you can display the contents of the Name Table:

```

Name Table Contents
Program 231 Funky Gtr
Program 245 FendJazzBass
Program 400 Obie Warble Pad
Program 600 Drawbarz
Keymap  220 Funk Guitar
Keymap  229 Jazz Bass

                                OK

```

Notice that the ROM program, "Program 103 3 Layer Kit", will not be listed in the Name Table. Any dependent objects that are in ROM do not need to be relinked by name. ROM objects are always directly referenced by their object ID number, since they don't get saved in any files.

Once the Song file has been saved, it can be loaded in at any time and correctly relinked, as long as the other files containing the necessary programs have already been loaded.

For this type of situation, where you may be working on songs always using a consistent set of programs, it is beneficial to make a Macro file that can be loaded in one step to direct all of the various program files to be loaded in. After that, any time you load in a song file containing a Name Table referencing these programs, the songs should get relinked to the correct programs.

If you happen to have multiple copies of the necessary programs already loaded into different banks, you can control which bank of programs will be linked to the songs by choosing a certain bank to load the song file into. The relinked programs will be the first set encountered according to the **Relink-by-Name search algorithm** defined above.

## Not Loading the Name Table

There may be a time that you wish to load in objects from a file containing a Name Table, but you don't want the K2500 to relink any dependent objects according to the Name Table. This can be accomplished by "Opening" the file from the Load function, and selecting any desired objects from within the file, **except** the Name Table object. The selected objects will be loaded in to the bank you specify, however the Relink-by-Name mechanism will not function.

## Relink-by-Name Processing Time

Normally, the time taken to relink several dependent objects using the name search will be insignificant, relative to the time it takes to load in the data from the file. However, if you are attempting to relink a very large amount of dependents by loading one file (i.e. say, 200 samples or so), there may be a noticeable wait while the K2500 searches its object database for the dependents. If this happens, it's best to be patient.

## The Multiple Object Selector Page

The Multiple Object Selector is used for making multiple selections of objects from an object list according to combinations of selection modes such as object type, object ID, and dependencies. This page is available for use in the **Load Object** dialog, the **Save Object** dialog, and all of the object utilities that are described in the Master mode chapter (**Move**, **Copy**, **Name**, **Delete**, and **Dump Objects**.)

The Multiple Object Selector page is entered by pressing the **Multi** soft button when in the dialog pages for the above functions. This page contains parameters to define a group of objects cross-referenced by object type and object ID. Objects can be grouped according to the dependencies of one or more objects, for all of the object functions *except Load Object*. (As described previously, the Load Object function can not determine exact object dependency relationships until the objects are actually being loaded into the K2500. The other object functions operate on lists of the objects currently stored in the K2500's RAM, and because of this, dependent relationships are available to be used as selection criteria.)

This is what is displayed by the K2500 when the Multi button is first pressed:

```
Multiple Object Selector
Select :Type/Range
Type   :Sample
Bank   :200's
StartId: 200      EndId: 299

All    Type Toggle Clear Set Cancel
```

This is the screen from which you can specify a range of object ID numbers cross-referenced with object type. If you increment the **Select** parameter to the **Dependents** setting, you will see the following additional parameter layout (not available for Load Object). This is used for selecting dependent objects of either the object currently scrolled to (from the object list that you were just viewing before pressing Multi) or dependent objects of the selected objects (indicated with an asterisk in the object list):

```
Multiple Object Selector
Select :Dependents
Of      :Current Item
Specify:All

Current = Program 205 Viola Section

All    Type Toggle Clear Set Cancel
```

The "**Current =**" parameter is for information only and represents the currently highlighted item in the object list that you are working with.

Two other selection modes do not have any additional parameter associated with them. These are **Everything** and **Search String**.

```
Multiple Object Selector
Select :Everything
```

```
Multiple Object Selector
Select :SearchStrg
```

"Select: Everything" is one way to choose all objects in the list. The All button performs a similar function, as described below.

"Select:SearchStrg" allows you to enter a string to be searched for in all of the object names in the list. Any objects with names containing the string can be selected. (Like "Dependents", this mode is available for all functions except Load Object.)

## Multiple Object Selector Soft Buttons

Here is a functional description of the soft buttons displayed on the Multiple Object Selector page. This is followed by an explanation of the parameter operation for all of the above object selection modes.

- All** Returns the Select parameter to **Type/Range**, if it was not already set that way. Sets the Type to **All Types** and the Bank to **All Banks**, and also the StartId to **0** and the EndId to **999**. This soft button is provided as a quick way to define the selection range as "Everything". Once the Type and Bank parameters have been set up in this manner, it is easy to change either parameter to further define the selection range (i.e. All Types of objects in the 400's bank, or Programs in All Banks)
- Type** Returns the Select parameter to **Type/Range**, if it was not already set that way. Sets the Bank to **All Banks**, and also the StartId to **0** and the EndId to **999**. The Type parameter is set so that it **matches the type of the currently indexed object from the object list**. For instance, if you were scrolled to a Setup object when you pressed Multi, pressing the Type soft button would set up the Type parameter to Setup. This is usually used to quickly select or deselect all objects of a particular type by scrolling to the first object of that type, and then pressing **Multi->Type->Set** or **Multi>Type->Clear**. If you don't want to include all banks in the selection range, it is easy to adjust the Bank or ID parameters to narrow the range.

The following three soft buttons will trigger the selection and deselection of items from the current object listing, according to the function of the particular button. Only objects that are included in the specified range will be selected or deselected. When this process is completed, the display returns to the object function dialog.

- Toggle** **For each of the objects in the specified range, "toggle" the selection status of the object.** In other words, if an object is already selected, then the object will be deselected (the asterisk will go away). If an object is not already selected, it will be selected (as indicated by an asterisk in the display between the object ID and object name.)

Toggle is very useful for situations where you want to select all objects in the list except those that meet certain qualifications. For instance, in the Delete Object utility (from the Master page, press **Object->Delete**), you may want to delete all objects that are not being used by a song that you are working on, in order to free up sample or object RAM in the K2500. You could do this by first selecting the song and its dependents (with no other objects selected, highlight the song, press **Multi**, then select "Dependents" of "Current Item" and press **Set**) and then returning to the

Multiple Object Selector page, set the selection range to Everything (or press the **All** soft button), and press Toggle. This will return you to the object list display, with everything that is **not** in use by the song now selected for deletion. All you would have to do then is press OK and confirm the delete operation.

- Clear**     **Clears the selection status of all objects within the specified range.** A basic example of this would be if you wanted to save a file containing everything in RAM except Programs. You could first select all the objects with the Left/Right cursor button double-press, and then go to the Multiple Object Selector. From there you could set the selection range to be Programs in All Banks, and press Clear. The K2500 would return to the object list display, with the only the desired objects selected for saving.
- Set**       **For each of the objects in the specified range, select the object.** For Example, to select all Keymaps and Samples in the 300's bank for saving, you would first set the selection criteria to Type/Range, set the Type to Samples, set the Bank to 300's, and press Set. Then, you would press Multi again, and just change the Type parameter to Keymap, and press Set again. The desired range of objects would then be selected and ready to proceed with saving.
- Cancel**    Exits from the Multiple Object Selector page, with no action, and returns to the object function, with the previously displayed object list.

## **Entering Selection Criteria in the Multiple Object Selector**

This section describes the operation of the selection modes provided on the Multiple Object Selector page. These are accessed by scrolling the **Select:** parameter to different values, as pictured above.



### Selection by Type/Range

This mode sets a specific range of object types cross referenced with a specific ID range.

Parameters	Possible Values	Function
<b>Type</b>	Sample, Keymap, Effect, Program, Setup, QABank, Vel-Map, PrsMap, IntTbl, Song, Table, All Types	Sets the desired object type. The value <b>All Types</b> will select all of the other possible types.
<b>Bank</b>	000's, 100's, 200's, 300's, 400's, 500's, 600's, 700's, 800's, 900's, All Banks	<p>Sets the desired bank. Changing this parameter causes the StartId and the EndId to be set to the limits of the chosen bank (i.e. setting this to 300's will set the StartId to 300 and the EndId to 399). Setting this parameter to the value All Banks will set the StartId to 0 and the EndId to 999.</p> <p><b>The actual range used for selections when Toggle, Set, or Clear is pressed is taken from the setting of the StartId and EndId parameters.</b> For example, if you set the Bank to 200s and then change the StartID to 300 and the EndID to 399, the 300s bank will be selected, not the 200s.</p> <p>The Bank parameter is used as a quick way to set up the Id range for an entire bank, or all banks.</p>
<b>StartID</b>	0 - 999	Sets the specific starting ID of the selection range.
<b>EndId</b>	0 - 999	Sets the specific ending ID of the selection range.

It is possible to set the EndId before the StartId. If this is the case, no selection range will be enabled.

### Selection by Dependents

This mode is used to select a group of objects that are dependents of other objects. This is not available in Load Object

Parameters	Possible Values	Function
<b>Of</b>	Current Item, Selected Items	<p>If set to <b>Current Item</b>, selection range is confined to those objects in the object list that are dependents of the currently indexed item (<b>Current =</b>), including the currently indexed item itself.</p> <p>If set to <b>Selected Items</b>, then the selection range includes any objects in the object list that are dependents of any currently selected objects. The currently indexed item that is displayed is ignored unless it is already explicitly selected (has an asterisk).</p>
<b>Specify</b>	All, All->Keymap, All->Program, Keymap->Sample, Samples Only	<p>This parameter is used to limit which dependent objects are included in the selection range for the appropriate objects included via the <b>Of</b> parameter. The normal setting is <b>All</b>, which means all dependents are included. The other settings are primarily useful when separating objects into different files for reloading later using Macros and Relink-by-Name.</p> <p>If set to <b>All-&gt;Keymap</b>, then the selection range will include any dependent objects down to the level of keymaps. That is, samples will be excluded from the selection range.</p> <p>If set to <b>All-&gt;Program</b>, then the selection range will include any dependent objects down to the level of programs and effects. Another way of saying this is that any Keymaps and Samples will be excluded from the selection range.</p> <p><b>Keymap-&gt;Sample</b> cause the selection range to only include dependent objects that are keymaps or samples.</p> <p><b>Samples Only</b> will just include sample dependents.</p>
<b>Current =</b>	Type, ID, and name of the currently indexed object	Displays the object that will be used if Current Items is the value of the <b>Of</b> parameter.

### Everything

"Everything" will include all objects in the list. You may prefer to use the All button for this purpose.

### Search String

This selection mode will ask for a search string to be entered, as soon as either the Toggle, Clear, or Set button is pressed. The range for the selection/deselection will be any objects whose names contain the search string, ignoring upper/lower case. Search String mode is not available in Load Object.

## Working with the Multiple Object Selector

The Multiple Object Selector can be helpful to avoid a lot of button presses and quickly allow you select whatever group of items in the K2500's RAM that you want. It is available in all of the related object management functions.

You may notice that the cursor positions and parameter settings are remembered whenever you exit the Multi Selector dialog, even if you exit the dialog and choose a different different function. For instance, if you end up doing a lot of selecting of samples, or of dependents at various levels, the parameters will stay set up the way you left them as you move from function to function (i.e. from Copy to Delete to Save, etc.)

The "Select Dependents" mode is very useful not just for saving dependents, but also for splitting up groups of objects for placing in different files. By using the optional settings for the **Specify** parameter (**All->Keymap**, **All->Program**, **Samples Only** etc.), you can separate the group of objects that you want to save at any level of the object tree that is necessary.

### Examples of possible operations using Multiple Object Selector:

- Select all the keymaps that are dependents of a block of programs.
- Select all the samples starting from ID number 398.
- Select all the objects that have "piano" in their object name.
- Select the programs, setups, and effects that are dependents of song 200.
- Select all of the keymaps and samples that are dependents of songs 400-410.

## More Features of the Save Dialog

### The Choose File Name Function

When entering in a file name for saving, there is a **Choose** soft button. This was briefly mentioned earlier in the section "Creating Directories." When Choose is pressed from the file naming dialog, the K2500 will access the current disk directory and display the following:

```

Dir:\          Sel:0/10      Index: 1

Choose file name: BOTTLE   .KRZ   48K
                  CLAU     .KRZ   1207K
                  REGGAE   (dir)
Total:664K      STICK     .KRZ   550K
                Root  Parent  OK  Cancel

```

The function of this dialog is to grab the text of any file name on the current disk, and either use it as a starting point in the file naming dialog, or else use the chosen filename exactly. This helps when replacing files on the disk (where the name must exactly match the file being replaced), or adding files to the disk that have similar names or appended revision numbers. You can save time by not having to type the entire file name in from the K2500's keypad.

The **Open** soft button is only visible in "Choose file name:" dialog when a subdirectory is highlighted.

Traversing directories from the Choose function does not change the current default directory.

### Selecting the Directory to use for Saving a File

After you specify the file name when saving any file, you can then choose either the default current directory for saving the file in, or any other directory on the current disk:

```
Use current directory for BOTTLE.KRZ?  
(Path = \)
```

**Change** **OK** **Cancel**

Pressing OK will select the default path (the current directory), which in this display is the root directory, "\". Pressing Change will allow you to view the disk, traversing its directories, until you find the one in which you want to save the file. If you choose a different directory from the default, it will become the new default directory. For more information on selecting a directory, see "The Directory Selection Dialog", near the beginning of this chapter.

### Saving Any File sets the File Index

After saving a file, you can go to any disk function (such as Load), and the just saved file will be automatically highlighted. This makes it easy to find a file that you have just saved, in case you want to delete it, add it to a Macro, move it to a different directory, open it (if it is a.KRZ or a .MAC file), etc.

## Macros

The K2500 lets you create lists of disk files called Macros. The files can be located on any drive, whether SCSI based or the internal Floppy drive. Files from SCSI drives in Roland and Akai format can also appear in Macros. Ensoniq files are not currently supported by K2500 macros.

Macros are stored in a data object called a Macro Table, and these can exist in two forms:

1. A Macro Table object in the K2500's non-volatile RAM.
2. A disk file, containing one Macro Table object. This disk file is called a Macro file, and it has a **.MAC** extension (visible in the directory listing).

Macros are primarily used to configure a K2500 with sound and sequence data from several files, or selected objects within files. When a Macro file is loaded, every file that is listed in its Macro Table will be loaded in, according to the order of the entries in the Macro Table.

### The Macro Page

There can only be one Macro Table in the K2500's memory at any time. This object is created for the first time by turning on **Macro Record** mode, from the Macro page, which you reach by pressing the **Macro** soft button on the Disk mode page, as shown below.

```

DiskMode    Samples:12313K  Memory:102K
Path = \CYBER\

CurrentDisk:SCSI 2          Startup:Off
                          Library:Off
Direct Access, 84MB
Psyquest PS-427          XMC1.7
<more> Load Save Macro Delete more>

```

The following screen is what you will see if Macro recording is Off:

```

Func:MACRO    [ Off ]    Index: 0

```

```


```

```

Select Modify Load Record On Exit

```

The top line displays the disk function ,

```

Func:MACRO

```

the current Macro mode,

```

[ Off ]

```

and an index value into the Macro Table.

```

Index: 0

```

## Macro Modes

The K2500 has 3 Macro modes -- Record, Pause, and Off. The meaning of each of these modes is:

**Off** There is no Macro object in the K2500 memory.

**Record** A Macro object exists, and the K2500 records any file load operations that occur.

**Pause** A Macro object exists, but the K2500 *does not* record any file load operations that occur.

Note that whenever the Macro mode is Off, there are two soft buttons labeled **Record** and **On**. Pressing On will enable Macro Record mode, and then will return to the Disk Mode page. As an alternative, pressing Record will also enable Macro Record mode, but the display will remain on the Macro page. Once you press Record, the soft buttons and the top line of the display will change. The screen will look like this:

```
Func:MACRO [Record]      Index:  0
```

```
Select Modify Load Pause Off Exit
```

The new Macro mode is displayed ([**Record**]), and the soft button that used to say Record now says **Pause**. The soft button that used to say On now says **Off**. Pressing Pause will cause the Macro mode to read [**Pause**] and the Record soft button will reappear. You can switch between Record and Pause by pressing this button repeatedly.

Whenever Macro Record mode is enabled, you will see the indicator "**(Macro on)**" near the top left of the display on the Disk Mode page:

```
DiskMode  Samples:10022K  Memory: 42K
Path = \
(Macro on)
CurrentDisk:Floppy      Startup:Off
                        Library:Off
```

```
<more Load Save Macro Delete more>
```

## The Macro Table

When Macro Record mode is enabled after being in the Macro Off state, a new object called a Macro Table gets created in the K2500's memory. (The Macro Table is sometimes also referred to as a *Macro list*.) In the Save Object function display list, the Macro Table would appear as:

```
Table      35 Macro      14
```

A Macro Table can only be deleted from memory by pressing the **Off** soft button, or by performing a hard-reset of the K2500. Pressing the Off button will display the following question:

```
Reset macro?
```

```
Yes No
```

Pressing Yes will delete the Macro Table from memory, and then will return to the Disk Mode page. The Macro mode is set to Off, and the **(Macro on)** indicator is no longer displayed on the Disk Mode page.

Pressing No will return to the Macro page with no action taken. The "Reset macro?" question is displayed to allow you to change your mind about deleting the Macro Table, in case you have accidentally pressed the Off button.

When the Macro Table is first created it takes up a minimal size (14 bytes) in your non-volatile RAM. With each new entry that is added, the Macro Table will increase in size by approximately 40 to 100 bytes (or possibly more if the entry specifies an individual object list). When the Pause mode is entered, the Macro Table continues to remain in memory but does not change size since file operations are not being recorded. This is useful if you need to load some files into the K2500 but you don't want them to be entered into the Macro Table.

A Macro Table gets progressively larger as more file operations are recorded into it. Because of this, you will want to control when a Macro Table is stored in your K2500's memory. If you are concerned about keeping enough free space for all of your other objects, you may want to practice the following strategy - when you want to record a Macro, enable Macro Record mode, and when you are finished (and possibly have saved some Macro files), turn Macro mode Off. This will reclaim the storage space taken up by the Macro Table.

If you are confident that you have enough K2500 object memory installed and that you can allow the Macro Table to increase in size as you work, then you may wish to leave Macro Record mode enabled all the time. This can be useful if you want to keep a running history of files you have recently loaded in. Both the Macro mode and the Macro Table are remembered between power-cycles of the K2500 via the battery-backed memory.

A Macro can hold as many entries as there is space for in your K2500's non-volatile RAM.

## How to Make a Macro File

This section will take you through creating, saving, and loading in a Macro file. A simple example will be used. Afterwards, you will be able to apply the example and create your own Macro files.

The first step in making a Macro file is to turn on Macro Record mode (from the Macro Page, press On), if you have not already done so.

### Creating the Macro

Suppose you have the following four files on your disk (on SCSI ID 5, in the directory \ANALOG\) that contain analog style synthesizer patches, and you would like to have one Macro file that will load them all in:

```
Dir:\ANALOG\    Sel:4/4    Index:    1

File to load:
MULTIUDX .KRZ*    98K
NOISE .KRZ*    36K
RESONANT .KRZ*    109K
SYNAPSE .KRZ*    421K
Total:664K
Select Root Parent Open OK Cancel
```

Using multiple selection, you can select all four files, as shown. When you press OK you will see the "Load selected as:" dialog, but with the extra soft buttons **Macro** and **Insert**:

```

Load selected as: 200...299
                  300...399
                  400...499
                  500...599
Insert           Macro  OK  Cancel

```

The extra soft buttons are available only when a macro is being recorded. First, select the bank that you want, as usual. Pressing **OK** means to load all of the selected files into the K2500, and add all of the files to the Macro list. If all you are doing is creating a Macro file, you may not want to load the files at this time. However, you still would like to enter the files into the Macro list. Pressing **Macro** does exactly that. It means *don't load the files, but do enter them into the Macro list*.

When you add entries to a Macro, they will be added to the end of the Macro list by default. **Insert** is for inserting file entries at any point in a Macro list. It is discussed a bit later, in the section on Macro editing.

Once you have pressed either Macro or OK, you will see the load modes soft button menu that contains all of the various load modes (OvFill, Overwrt, Merge, Append, and Fill), whether the bank is empty or not. The reason for this is that you might need to be able to specify a mode such as Overwrite even if at the current time, the selected bank is empty. Whatever you choose here for the load mode (as well as your choice of bank) will be saved in the Macro list. For the sake of this example, let's choose the 200's bank and Fill mode. As soon as you choose the mode, the K2500 will either load in the selected files or not, depending upon whether you pressed OK or Macro. In either case, they will be added to the Macro.

### Saving the Macro File

You have now created a Macro. You can immediately verify that the selected files have been added to the Macro Table by looking at the Macro page (from the Disk Mode page, press Macro).

But right now, let's proceed with this example by saving the Macro file. From the Disk Mode page, press **Save->Macro**. You'll see the following screen:

```

Func:SAVE MACRO  Sel:0/4  Index:  1
5:\ANALOG\MULTIVOX.KRZ      200:F:
5:\ANALOG\NOISE.KRZ        200:F:
5:\ANALOG\RESONANT.KRZ     200:F:
5:\ANALOG\SYNAPSE.KRZ      200:F:
Select                     All  OK  Exit

```

This is called the **Save Macro page**. The soft buttons on this page control which Macro list entries (*Macro entries*) will appear in the Macro Table that will be stored in the Macro file. You can do multiple selection of entries with the **Select** soft button. Selected entries will have an asterisk on the first character of the display line, such as this:

```

*5:\ANALOG\RESONANT.KRZ      200:F:

```



You can use the following double-presses to select and deselect all entries in the list:

**Left/Right cursor double-press -> Select All Macro Entries**

**Up/Down cursor double-press -> Clear All Selections**

The top line indicates how many total Macro entries are in the current Macro Table, and how many are selected.

Pressing **OK** will cause only the selected Macro entries to be saved in the file. If you press **OK** but have not selected any Macro entries, then the Macro file will only have one entry in it, that of the highlighted entry.

There isn't much use for a Macro file with only one entry in it, however it can be useful as a way to create a link to an often used file on a drive. For instance, you could create a Macro file called \PERC.MAC in the root directory on the disk that just loads in a file \MYSOUNDS\PERC\GOOD\PERC.KRZ. Then, you would have fast access to loading this file without having to go find it, yet the file would still be organized in a subdirectory.

If you know that you want to save all of the entries into the Macro file (as we do for this example,) just press **All**. The K2500 will go through the standard file saving dialog in which you choose a file name and select a directory to save the file in.

Let's save the file as \ANALOG\SYNTH.MAC. Macro files are automatically saved with the .MAC extension. You will see the following on the screen as the file is being saved:

Writing file SYNTH.MAC...

Table      35 Macro                      - 162 b

### Loading the Macro File

So far, so good. We have created a Macro in memory and saved it to the disk, in the same directory as the files that are listed in the Macro.

While this example only loads files from within a single directory on a single drive, this is for the simplicity of the example. You can have files from any drive or directory in a Macro.

Now, let's go to the Load page and try to load the Macro file. This should load all of the files of the Macro one by one. When you return to the Load page, the file list will automatically highlight the Macro file that was just saved (as it would after any type of file that you save):

```
Dir:\ANALOG\      Sel:0/5      Index:      5
                RESONANT .KRZ      109K
                SYNAPSE  .KRZ      421K
File to load:SYNTH  .MAC      .5K
```

```
Total:664K
Select Root Parent Open OK Cancel
```

Press **OK** to load in SYNTH.MAC. Now the display reads:

```
Load this macro as:Specified
                   300...399
                   400...499
                   500...599
Insert             Macro  OK  Cancel
```

There are a couple of things to notice here. The first is a new choice in the bank list, **specified**. "Load this macro as specified" means load all the files in the Macro following the exact instructions for the bank and load mode for each file. All of the files were specified to be loaded into the 200's bank using Fill mode. If this is acceptable at the time you want to load the macro, you can just press OK. Otherwise, you can override the bank and mode settings for the entire Macro by choosing a different bank and mode before pressing OK (this is called *rebanking* the Macro.)

The other thing noticeable about the above display is that the Macro and Insert buttons are still available. This is because the Macro mode is still "Record". It is also perfectly acceptable to load a Macro file "into a Macro". If you do this, then **the K2500 will add all of the Macro file's entries into the in-memory Macro list**. Later on, you will see how loading a Macro file into the in-memory Macro list is a convenient way to edit a Macro file or combine elements of several Macro files into one Macro.

In this example, the in-memory Macro is still there - it didn't go away after saving the Macro file (this only happens when you turn Macro Off, as described above.) Since Macro Record is still enabled, loading in the SYNTH.MAC file will have the effect of adding a duplicate set of entries into the in-memory Macro.

Whatever method of loading you choose (i.e., specified in the Macro or overriding the Macro), the K2500 will proceed to locate each file in the Macro in the exact order in which the entries were listed when the Macro was saved. If the files were on different drives in your disk system, you would be able to observe your various drives as they were selected in turn and files were loaded from them.

If the K2500 cannot locate one of the files, you'll see a "Not Found" error message. If a drive cannot be accessed (for instance, if the SCSI ID stored with the Macro entries in this example was no longer the current SCSI ID of the drive), you will see the message "Problem mounting disk.", to which you must press **OK**. If "Confirm" is set to "On" on the Master Mode Page, loading will stop on the first error message, giving you a chance to cancel the operation or keep going. If you answer "yes" at this point, the operation will continue, even if the K2500 encounters subsequent errors. If you run into a lot of errors due to loading an out of date Macro file, the Macro process can be discontinued using a special procedure described later in the section "Aborting a Macro Load."

When the Macro is done loading, you will see the screen:

```
Macro SYNTH.MAC completed...
```

The K2500 will then return to the Disk Mode page. You should now be able to go to Program Mode, or the Save Objects display, and verify that all of the objects from the files of the Macro are now in the K2500's memory.

## Macro Entries

Each file load operation that is recorded into the Macro Table is called a Macro entry. While Macro Record is enabled, you have the option of either loading files into the K2500 or just adding files to the Macro without actually loading them.

Each Macro entry stores information about how a disk file should be loaded. Each entry is displayed as a single item in a scrollable list on the Macro page, with various fields indicating the parameters of the entry. This picture shows how the Macro page might look once 4 files have been entered into the table:

```

Func:MACRO [Record]      Index: 1

3:\DRUMS\REALKITS.KRZ    200:F:Obj
3:\BASSES\WALKING.KRZ    200:F:
3:\KEYS\CHROMA12.KRZ     200:F:
F:\SONG42.KRZ            200:F:
Select  Modify  Load  Pause  Off  Exit

```

The following chart describes the information stored in a Macro entry, and also tells how to the understand the settings of a displayed Macro entry:

<b>Disk drive ID</b>	<p>This represents what drive the file is to be loaded from. There are 10 possible values:</p> <p>The numbers <b>0-7</b> represent SCSI 0 through SCSI 7.</p> <p>The letter <b>'F'</b> represents the Floppy drive.</p> <p>The letter <b>'U'</b> means Unspecified drive. (see below)</p> <p>The letter <b>'L'</b> means the Library drive. (see below)</p>
<b>Directory path/file name</b>	<p>This is the directory path and the file name of the file on the disk to be loaded by this Macro entry. This is displayed to the right of the Disk drive ID, following a colon, using as many characters as needed that will fit in the display (up to 28). The internal Macro Table can store any path length, however.</p>
<b>Bank</b>	<p>The bank that the file should be loaded into. This will have a value from 0 through 900 (by 100s), or the letter <b>'E'</b> for Everything. This field is the first field to the right of the path/file name in the display.</p>

<b>Mode</b>	<p>The mode specified for loading in the file. The following one-letter codes are shown in the display:</p> <p>'O' means Overwrite mode (Overwrt)</p> <p>'V' means Overwrite/Fill mode (OvFill)</p> <p>'M' means Merge mode</p> <p>'A' means Append mode</p> <p>'F' means Fill mode</p> <p>This field is to the right of the bank field, after a colon.</p>
<b>Object indicator</b>	<p>If a list of objects will be loaded from the file, the Macro entry display line will have an "Obj" indicator as the last field on the right. (see above, for the file \DRUMS\REALKITS.KRZ)</p>
<b>Object list</b>	<p>If the Obj indicator is present for a Macro entry, it means a list of individual object types and IDs are stored in the Macro entry for the particular file. When the Macro is loaded in, only these individual objects will be loaded from this file.</p>
<b>Disk Format type</b>	<p>If the disk is a third-party SCSI format disk such as Akai or Roland, a format indicator will appear on the Macro entry display line in the last field on the right. The following codes are used:</p> <p>'Aka' means Akai format</p> <p>'Rol' means Roland format</p> <p>Ensoniq format disks are not currently supported in Macros.</p>

## Using the Bank and Mode Fields

The bank and mode fields will only be used if a Macro file is loaded "as specified". (This is one of the options when loading in a .MAC file) This means that each file listed in the Macro will be loaded in exactly as the fields of the Macro entry are specified. These fields can optionally be overridden in the Load dialog, and a new bank and mode can be chosen for the entire Macro.

It is not always important what the bank and mode are set to in a Macro entry, since the Macro can really be loaded anywhere by overriding the bank and mode in the Load dialog. These fields are more important when you want to automatically configure the banks of your K2500 the same way each time you load a Macro, with different files being explicitly loaded into different banks.

## Viewing the Object List for a Macro Entry

If a Macro entry contains an object list, it can be examined by scrolling the Macro list display until the item with the **Obj** indicator is highlighted, and then by pressing either the Left or Right cursor button on the front panel. You will see a display that looks like this:

```
Macro Object List (load dependents)
Program 210
Program 211
Program 212
Program 213
```

OK

Left or Right cursor button -> Displays Macro entry object list

The "Macro Object List" display, a scrollable list, shows what objects are to be loaded from the file specified in the currently indexed Macro entry. You will not see the names of the objects in this display, because they are not stored in the Macro Table. The objects are referenced only by object type and object ID. The "(load dependents)" indicator means that the Macro process should also load all dependents of the objects in this list.

If you need to know the names of objects in a Macro entry object list, it is possible to begin a disk function (such as Load), find the file specified in the Macro entry, press Open to display the file's objects, and look up what the objects are, using the information in this display.

## Unspecified Disk Drive ID

Before the time a Macro file is saved to disk, you can specify that a file's location will not be tied to a specific SCSI ID number. Instead, the file should be loaded from the same drive that contains the Macro file itself. This is called the "Unspecified" drive ID, since the actual ID is not known until the time of loading the Macro.

The Unspecified drive ID will typically be used by people who distribute K2500 sound files and Macro files on removable disk media such as **CD-ROMs** or **Magneto-Optical Cartridges**. This feature makes it possible to avoid specifying the exact SCSI ID used in a Macro file that is to be loaded on someone else's disk drive, since there is no way to predict the SCSI ID of the drive in advance.

Another use would be, if you are consistently working with Macro files that load mostly files from the same drive, then if you set the Macro's drive IDs to Unspecified you won't have to edit your Macro files if you happen to change the drive's SCSI ID.

Setting Macro entries to have an Unspecified drive ID is done from the Modify Macro page (press **DISK->Macro->Modify**), described later in this section.

## The Library Disk

If a Macro entry is set to the "Library" drive ID, it means that the file to be loaded should be found on the drive at the SCSI ID set by the **Library parameter** on the Disk Mode page. This designation is similar in purpose to the Unspecified drive ID, because it is a way to avoid needing to hard-wire the SCSI ID in advance. The Library drive ID is intended to be used in Macro files that reside on removable media such as a Floppy disk, whose Macro entries are supposed to load in particular files on a specific sample CD.

The main purpose of this feature is so that Macro files can be distributed on a floppy disks containing programs that link up with sample files from CD-ROMs. Formerly, a user would have to copy the CD-ROM file to their own hard drive if they wanted to make any new programs that use a CD-ROM file's sample and keymap data.

There are a lot of possibilities in the K2500 to come up with new programs for using with many available sample CDs. Now it is possible to distribute programs or sequences based on a floppy to other K2500 users who already have a certain sample CD in their collection. They

should be able to load the correct sequences of files just by loading a Macro, without those nasty SCSI ID conflicts.

A macro of this type would first load the CD-ROM files containing samples and keymaps (off the "Library" disk), and then load in files from the Floppy disk. The program files from the floppy should correctly link up to the pre-loaded keymaps and samples via the Relink-by-Name mechanism.

In order for the files in the Macro with a Library drive ID to be found, the user of the Macro must have already set up their Library parameter to be the SCSI ID of their CD-ROM drive.

In the following picture, notice that the Library parameter has been set to SCSI 3.

```

DiskMode    Samples:10022K    Memory: 42K
Path = \
(Macro on)
CurrentDisk:Floppy           Startup:Off
                             Library:SCSI 3

<more>  Load  Save  Macro  Delete  more>

```

Assuming SCSI 3 is the user's CD-ROM drive, the user is now prepared to load a Macro file from on a floppy disk that contains the following entries:

```

L:\PIANO4MB.KRZ           200:F:Obj
F:\PNOPROGS.KRZ          200:F:

```

When the above Macro entries are loaded, the file \PIANO4MB.KRZ will be looked for on the drive with address SCSI 3, since this is what the Library parameter on the Disk Mode page has been set to. If the correct CD-ROM has been placed in the drive, the K2500 will load the file, and then continue and load \PNOPROGS.KRZ from the floppy disk.

If the K2500 executes the above Macro, but the Library parameter has not been set (is set to Off) an error message

## Library has not been set

will be displayed when the Macro process attempts to load \PIANO4MB.KRZ.

The above example is a typical case where the Relink-by-Name feature is used to have one file containing programs depend on another file containing samples and keymaps, yet the files are on different drives. The CD-ROM in the example might just as well have been an Akai or Roland format, and the file from the floppy would contain the latest custom K2500 programs making good use of the imported data.

The Library parameter is remembered by the K2500 across power-cycles, via the battery backed memory. This means that if you use this feature to load Macros you only have to set it up once.

Setting Macro entries to have a Library drive ID is done from the Modify Macro page (press **DISK->Macro->Modify**), described later in this section.

Although the Unspecified and Library drive IDs are meant to be used with distributable media such as CD-ROMs and floppy disks, these features will work with any supported disk drives.

## Loading Selected Entries from a Macro File

It is possible to examine the contents of a Macro file from any disk function page - the same way you would open a .KRZ file to check out what objects are stored in it - by highlighting the .MAC file and pressing **Open**:

```
Dir:\ANALOG\      Sel:0/5      Index:    5
                RESONANT .KRZ    109K
                SYNAPSE  .KRZ    421K
File to load:SYNTH  .MAC      .5K
```

```
Total:664K
Select Root Parent Open OK Cancel
```

The K2500 will need to read the Macro file into a temporary area of internal memory, which means there needs to be enough free RAM to accommodate it. When Open is pressed from the Load function, a dialog similar to the Macro page and the Save Macro page will be entered, called the **Load Macro page**:

```
Func:LOAD MACRO  Sel:2/4      Index:    1

5:\ANALOG\MULTIQUX.KRZ      200:F:
*5:\ANALOG\NOISE.KRZ       200:F:
5:\ANALOG\RESONANT.KRZ     200:F:
*5:\ANALOG\SYNAPSE.KRZ     200:F:
Select Check              All OK Exit
```

From the Load Macro page, you can select one or more individual Macro entries for loading, instead of having to load the entire Macro. This is done using the **Select** soft button, identical to the method of saving Macro entries. In fact, this dialog operates identically to the Save Macro dialog, with one exception, the **Check** soft button.

The Check button will cross-check all of the Macro entries in this opened Macro file against the currently in-memory Macro Table, if there is one. Any entries in the opened Macro file that are not in the in-memory Macro Table will be selected when you press the Check button. The selected Macro entries can then be loaded in by pressing OK. This can be helpful to avoid loading in duplicate files if you...

- use Macro Record to keep a running history of files that you have already loaded into the K2500, and
- have a lot of Macro files that load in a similar lists of files.

If the Open button is pressed from a disk function other than Load, you will see the View Macro page:

```

Func:VIEW MACRO  Sel:0/4  Index:  1

5:\ANALOG\MULTIVOX.KRZ  200:F:
5:\ANALOG\NOISE.KRZ    200:F:
5:\ANALOG\RESONANT.KRZ 200:F:
5:\ANALOG\SYNAPSE.KRZ  200:F:
                                Exit

```

The only function of this dialog is to view the Macro list entries stored in a Macro file. This feature is useful when, for instance, you are about to delete a Macro file and want to know what information is contained in the file before you remove it from the drive.

## Editing Macros

The in-memory Macro Table can be edited from the Macro page. You can select one or more Macro entries and do any of the following operations to them:

1. Change the Drive ID.
2. Change the Bank and Mode settings.
3. Delete the selected Macro entries.

To edit a Macro file already saved on your disk, it is necessary to first load the Macro file into the in-memory Macro. If you recall, this is done by the following procedure:

Making sure that Macro Record mode is enabled, go to the Load function, highlight the Macro file you wish to edit, and then either select certain entries from the Macro file (by pressing Open to get to the Load Macro page), or just press OK to load the entire Macro file. When you see the following display, press the Macro soft button, so that the K2500 will not try to load the files listed in the Macro.

```

Load this macro as:Specified
                    300...399
                    400...499
                    500...599
Insert             Macro  OK  Cancel

```

If you want, you can rebank the Macro on the way in by scrolling the bank list to something other than "specified". Similarly, if you override the load mode, it will be reflected in the in-memory Macro Table as well.

To edit entries from the Macro Table, return to the Macro page. For this example we will edit all the entries at once (like other similar dialogs, if you only are concerned with one list entry, it does not need to be explicitly selected with the Select soft button.) You can use the following double-presses to select and deselect all entries in the list:

**Left/Right cursor double-press -> Select All Macro Entries**  
**Up/Down cursor double-press -> Clear All Selections**



With all the entries selected, our display looks like this:

```

Func:MACRO [Record]      Index:      1

*4:\NEWMIX\TRASHX12.KRZ      200:F:
*4:\POTS\TEHPOT.KRZ         300:F:
*4:\PANS\FRYING.KRZ          400:F:
*4:\KITCHEN\SINK.KRZ         500:F:Obj
Select Modify Load Pause Off Exit

```

Two soft buttons have yet to be explained. They are **Modify** and **Load**.

Press the Modify soft button to change any items mentioned at the top of this section. You will see the following display:

```

Modify Macro Entries

Modify:Drive
Drive :SCSI 4

4 entries selected.

Delete OK Cancel

```

This shows the screen that allows you to change the SCSI drive for the selected Macro entries. If you increment the **Modify:** parameter, the display switches to let you modify Bank and Mode information:

```

Modify Macro Entries

Modify:Bank/Mode
Bank :200's      Mode:Fill

4 entries selected.

Delete OK Cancel

```

The initial settings of the parameters on these screens are always taken from the lowest indexed Macro entry that is selected on the Macro page. In addition, every time you return to the "Modify Macro Entries" page, both the Modify parameter and the cursored parameter field will be the same.

Pressing **OK** will set all of the selected Macro entries to have drive ID or Bank and Mode settings according to the parameters set up on this page. The display will return to the Macro page with the same entries still selected. Any modifications to the parameters will be visibly apparent. Selecting multiple entries for editing allows you to change those entries in a uniform way. In our example above, you could change the Macro so that all the files were loaded into a single bank, instead of the separate banks they had previously been loaded to.

Press **Delete** to remove the Macro entries from the Macro list. You will see the display:

## Delete macro entries?

**Yes****No**

If you answer **Yes**, the display returns to the Macro page and all of the previously selected entries will be gone from the list. If you answer **No**, the display will return to the Modify Macro Entries dialog.

Pressing **Cancel** in the Modify Macro Entries dialog will return to the Macro page with everything that was selected still selected, but with no parameter changes made to any Macro entries.

Here are the parameter values for Modify Macro Entries:

Parameter	Values
<b>Modify</b>	Drive, Bank / Mode
<b>Drive</b>	Floppy, SCSI 0, SCSI 1, SCSI 2, SCSI 3, SCSI 4, SCSI 5, SCSI 6, SCSI 7, Unspecified, Library
<b>Bank</b>	000's, 100's, 200's, 300's, 400's, 500's, 600's, 700's, 800's, 900's, Everything
<b>Mode</b>	Append, Merge, Fill, Overwrite, OvFill

Once you have made the necessary changes to the in-memory Macro, you can go to the Save Macro page to write selected (or all) entries to a new disk file (or replace over the original Macro file you may have loaded in for editing.)

**Executing the In-Memory Macro**

You can load any group of files listed in the in-memory Macro. This is done using the **Load** soft button on the Macro page:

```

Func: MHCRO      [Record]      Index: 1

1:\STRINGS.KRZ      200:F:
*1:\PIANO.KRZ      300:F:
1:\DRUMS.KRZ      400:F:
*1:\NOISE.AIF      500:F:
Select Modify Load Pause Off Exit

```

Pressing the **Load** button gives you the choice of loading either **all of the files** in the Macro list or loading **only the files that are selected**:

Load selected items or all items?

**All Selected Cancel**

If you don't have any items explicitly selected (with an asterisk), the message you see when you press load is slightly different:

Load current item or all items?

This word "current" means the highlighted entry in the display will be loaded.

The files that have been selected for loading will be loaded in their respective order in the Macro list, using the bank and mode parameters that are specified in the list. In the above screen, if you were to load selected items, first \PIANO.KRZ would be loaded in the 300's bank, and then \NOISE.AIF would be loaded in the 500's bank.

## Macro Insert

It is possible to insert new Macro entries into the middle of the Macro Table if necessary. This is done by pressing the **Insert** soft button at the "Load this file as:" prompt, when loading in a file (if Macro Record is enabled):

```
Load this file as: 200...299
                  300...399
                  400...499*
                  500...599*
Insert           Macro OK Cancel
```

When you press Insert, you will see a dialog displaying the current in-memory Macro list:

```
Set Macro Insert Point      Index: 2
5:\ARCEILE.KRZ              200:F:
5:\METALIC.KRZ              200:F:
5:\STEREOZX.KRZ             300:F:Obj
5:\STRINGS\DBLBASS.KRZ      400:F:
5:\STRINGS\CELLOS.KRZ       400:F:
                           OK Cancel
```

Scroll the Macro list until the entry before which you want to insert is highlighted. In the above display, any new macro entries added by this load operation will be inserted in the Macro list just before the entry for METALIC.KRZ.

Press **OK** to enable the insert point. Press **Cancel** to disable the insert point.

The display will return to the "Load this file as:" dialog. If a Macro insert point has been set, an indicator will appear at the top left of the screen:

(Macro insert)

```

Load this file as:200...299
                  300...399
                  400...499*
                  500...599*
Insert           Macro  OK  Cancel

```

The insert point can be disabled before loading, by pressing Insert again and pressing Cancel from within the "Set Macro Insert Point" dialog. The "(Macro insert)" indicator will disappear. Similarly, the insert point can be changed before loading by pressing Insert again (the screen will highlight the current insert point,) scrolling to a different insert point, and pressing OK.

## Saving and Loading a Macro Table in a .KRZ file

Macro Table objects can be explicitly saved or loaded (without being "executed") using Save Individual Object and Load Individual Object. If for some reason you wanted to save a Macro Table that was being worked on "in-progress", and then be able to load it back in later on to be worked on some more, you would use this method. The Macro Table can be selected for saving and loading just like any other object. When you load in a Macro Table using Load Individual Object, it will overwrite any Macro Table already in memory. Once it is loaded in, you may have to go to the Macro page and enable Macro Record mode to continue to record further load operations into the Macro.

The "Save Everything" feature of the Save dialog will not include the Macro Table. This is done to prevent inadvertent distribution of what would most likely be a meaningless Macro Table to other users.

## Aborting a Macro Load

There is a way to abort the process of loading a Macro file. Aborting can only be done "in between" files that are being loaded, and not during the load of any one file (short of powering off or soft-resetting the machine with [simultaneously pressing [+/- 0 CLR]...but this is not recommended!).

Aborting a Macro load is done by pressing and holding down either of the Plus (+) or the Minus (-) buttons that are located just below the Alpha wheel. This should be done at least one-half second before you anticipate the current file to finish loading, or else the K2500 will not sense that you wish to abort the load.

You will see the following question after the current file being loaded is completed:

Abort the macro?

Yes No

It may be a good idea to practice using this method of aborting a macro file load, so that when the time comes that you load in an out of date Macro with 25 entries all at the wrong SCSI ID. you will remember how to abort the process. This same method (of holding the Plus or Minus buttons down) is also used to abort the Backup feature and the multiple file load feature.

If you run out of object or sample RAM, you will have one opportunity to abort the Macro as explained above. However, if you continue from that point you may end up seeing the same error message "Memory is full" for each file that you had selected. This can be a rather tedious process, however it is still possible to abort out of this by holding down the Plus or Minus button simultaneously while pressing "Yes" when you see the following question:

Abort this Partial load?

Yes

No

If the Macro that you abort was loaded by multiple selection together with other files (that may or may not be Macro files), you will have to "abort twice", once to get out of the current Macro file load, and a second time to get out of the multiple file load process.

If you are aborting a Macro because the Disk Drive ID is incorrectly specified (as evidenced by lots of "**Problem mounting disk**" errors) you will need to hold down either the Plus or the Minus button simultaneously while pressing OK to satisfy the error prompt. The screen may "blink" while holding down the repeating Plus or Minus button, but as soon as you have pressed OK you will see the "Abort the macro?" question.

## Remote Macro Load

You can remotely load a macro into the K2500 from a sequencer. This can be useful, for example, if the K2500 is inaccessible or inconveniently situated.

Once you've created the macro that you will be remotely loading from the sequencer, set the sequencer to record, then dump the macro object to the sequencer using the **Dump** soft button on the Master mode Object page. Then, add the SysEx LOADMACRO (10h) command to the sequence, following the macro object. Although some sequencers allow you to record a SysEx command directly into a sequence, the K2500 provides a convenient shortcut, described in the next paragraph.

To add the LOADMACRO command to a sequence (after dumping the macro object to the sequencer), leave the macro object highlighted on the Master mode Object page, then press the left or right cursor button. The K2500 will display: "Send SysEx Load Macro command?" Press the **Yes** soft button and the K2500 will add the LOADMACRO SysEx command to the sequence.



**NOTE:** You cannot remotely load a macro to the K2500 while the K2500 is on the Disk mode page or in an edit mode.

## Disk Utilities

The Disk Utility functions provide certain necessary information about disks and their files and directories. These functions are useful when you want to know how your disks are organized and how much disk space you have available. They also help you to locate files and directories.

To access the Disk Utilities page, press the **Util** button from the Disk Mode page. The Disk Utilities page looks like this:

```
DiskUtil: SCSI 2
```

```
Select utility function:
```

```
Find List Free Done
```

The functions on the Disk Utilities page are used for finding out information about the Current Disk. The Current Disk is always indicated on the top line of this page. If you want to use the utility functions for a different disk, you must first set it to be the Current Disk on the Disk Mode page.

Here is a brief description of each soft button:

- Find** Find files utility. Allows you to search for files that match a certain character string in their file names.
- List** Lists an expanded directory tree from any level of the hierarchy. This function can be used to determine the total size of files within any tree of subdirectory. It is also helpful for finding files on the disk.
- Free** Calculates the available space on the disk.
- Done** Exits back the Disk Mode page.

### Find Files

The Find files utility first prompts you to enter a character string to be searched for:

```
Search string: RAT_
```

```
Delete Insert >>End Choose OK Cancel
```

You can use the **Choose** button to grab the text of a file name from the current disk, as described previously.

If you press **OK**, the K2500 will begin to search the disk for any files or directories that contain the search string in their names. The search starts in the root directory and searches the entire disk. When a matching file or directory is found, you'll see one of the following:

Found file:

\BABYTOYS\RATTLE.KRZ

FindNext Go To Cancel

Found directory:

\CRATES

FindNext Go To Cancel

If the search string is found anywhere within a file name it will be matched. The search algorithm independently checks both the file name and the extension. For instance, if you wanted to find any Macro file on the disk, you could enter in "MAC" for the search string. This would find any Macro files as well as any files or directories that have "MAC" in their filename.

When a match is found, there are three choices displayed:

**FindNext** Continue searching the disk for another file or directory that matches the search string.

**Go To** Exit to the Disk Mode page, setting the current directory and file index of the K2500 to the location of the found file or directory.

The next disk function you choose will display the current directory with the found file already highlighted. If a directory was found, then the first file in the directory list will be highlighted.

**Cancel** Exit to the Disk Mode page.

When the search has checked all of the items on the disk, the K2500 displays this screen:

Search completed.

OK

If no matching files were found, you will also see

(No files found)

The K2500 will remember the last search string that you entered. This makes it easy to repeat the same search. If you press **Util->Find** again, the "Search string :" dialog will contain the previously used string.

## List

The List utility allows you to view directories on a disk with the expanded contents of all sub-directories. Each line is indented according to its directory level, so that you can get sense of how your directory tree is organized.

```
Dir:\          [ 1968K]  Index: 1

ELEPHANT.KRZ          148K
  BSOUNDS\
    TRUMPETS\
      JSBACH.KRZ          712K
Root  Up  Down        Go To  Exit
```

The **Dir:** parameter shows the directory that is being listed. The size value displayed on the top line of the display represents the total size of the directory subtree that is currently being viewed. The **Root**, **Up**, and **Down** soft buttons control going from one directory to another in the hierarchy:

- |               |   |
|---------------|---|
| <b>Root</b>   | Display the disk from the top level, meaning that all of the files on the disk will appear in the scrollable list.  |
| <b>Down</b>   | Set the display to the level of the highlighted file or directory.<br><br>For example, scrolling to JSBACH.KRZ in the above list, and pressing Down , would focus the list on the contents of the TRUMPETS directory. |
| <b>Up</b>     | Set the display up one directory level.   |
| <b>Go To</b>  | Exit to the Disk Mode page, setting the current directory and file index to the location of the highlighted file or directory.  |
| <b>Cancel</b> | Exit back to the Disk Mode page. The current directory is unchanged.  |

The files are listed in the order that they appear on the disk, unalphabetized. The traversal order of the directories is the same one that is also used for the Backup function.

## Free

The Free utility calculates the amount of free space on the current disk and displays the result on the Disk Utilities page. This may take a small amount of time to calculate, depending on the drive.



Computing free space on SCSI 2  
Please wait...

```

DiskUtil: SCSI 2 (DOS)
Used:23% Free:94814K Total:124396K

Select utility function:

```

```

Find List Free Done

```

The parameters tell you the following:

**Used** The percentage of the disk that is taken up by the existing information stored on it.

**Free** The amount of disk space available for new files.

**Total** The size of the disk. This will be the size of the usable partition if MS-DOS format.

For SCSI drives, if the current disk was formatted on a PC or a Mac in MS-DOS format and contains at least one primary partition, you will see the "(DOS)" indicator on the top line. Using this format is described in the section on "MS-DOS compatibility".

For floppy drives, the density (HD or DD) will be displayed. All K2500 floppy disks are MS-DOS format compatible.

## Moving Files Between Directories

Files and directories can be moved from one directory to another on a given disk. You can either choose multiple files to move using the Select soft button, or just move the single highlighted file or directory (if no other files are selected). As you would probably expect, moving a directory also moves all the files within the directory. To use this function, press the **Move** soft button from the Disk Mode page. Then choose the file or files that you want to move:

```

Dir:\ Sel:0/15 Index: 1

File to move:ATOMTOM .KRZ 98K
              BLOWFISH.KRZ 36K
              COLORS .KRZ 109K
              DRUMHITS (dir)
Total:6846K
Select Root Parent Open OK Cancel

```

Press **OK** when you have made your selection. Press **Cancel** to return to the Disk Mode page.

The K2500 remembers the most recent destination directory that a file was moved to. If the current directory is different from the most recent destination directory, you will see the question:

```
Use default directory for ATOMTOM.KRZ?
(Path = \DRUMHITS\)
```

**Change** **OK** **Cancel**

Press **OK** to use the default.

Press **Change** if you want to select a different destination from the default shown. The K2500 will then display a directory selection dialog (described early in this chapter), and you can select the move destination directory from there:

```
Dir:\                               Sel:0/15      Index:    4
                                BLOWFISH.KRZ      36K
                                COLORS  .KRZ      109K
Select dest dir:DRUMHITS          (dir)
                                EARTHLNG.KRZ     144K
                                HANDCLAP.KRZ     645K
Total:6846K                       INDUSTRY      (dir)
Root  Parent  Open  Current  SetDir  Exit
```

The move operation begins when you press either **Current** or **SetDir**.

If the default destination directory is the same as the source directory, the K2500 will skip the "Use default directory?" question and instead go right to the "Select dest dir:" dialog.

A good way to organize files into subdirectories, with a disk drive that was being used with previous versions of K2500 software, is by using the Move command. First, create the subdirectories you need, using the **NewDir** function. Then, use multiple file selection to select the files to be grouped into a particular subdirectory. The files can be moved to their new location in one operation.

For each file that is moved, you will see a confirmation message:

```
\ATOMTOM.KRZ moved to
\DRUMHITS\ATOMTOM.KRZ
```

Note the following:

- You can select multiple files for moving within a directory. However, you cannot move files from more than one directory at a time. If you select any files and then switch to another directory, the files you had chosen will be deselected.
- If you are moving a directory, you cannot move it in to a subdirectory of itself.
- If the source and destination directories are the same, the file will not be moved, and an error message such as the one below is displayed. This would happen if you pressed **Current** above.
- The same message will be displayed if there is a file in the destination directory with the same name as the file to be moved.

```
File \ATOMTOM.KRZ not moved.
```

## Renaming Files

Press the **Rename** soft button (from Disk Mode) to rename files or directories without loading them. When you press **Rename**, the K2500 will prompt you to select the file to be changed, by showing you a list of the files found on the current disk.

When you've selected the file to be renamed, press **OK**, and the K2500 will ask you to enter the new file name. When you've done this, press **OK**, and the file name will be changed.

```
Dir:\          Sel:0/1      Index: 1
```

```
File to rename:FILE9876 .KRZ  2048K
```

```
Total:2048K
```

```
Root Parent Open OK Cancel
```

This function can only be used to change the 8-character file name and not the extension. When you press **Rename**, the File List dialog is displayed and you can navigate through the directories to choose the file or directory you wish to rename. Unlike the other disk functions that use the File List dialog, you will not see the **Select** soft button. This is because you can only rename one file at a time. Therefore you simply choose the file you want and press **OK**. The K2500 will then ask you to enter a new name. Once you've done this, press **OK** again, and the file name will be changed.

## Deleting Files and Directories

Press the **Delete** soft button (from Disk Mode) to delete files and directories. The Delete function supports multiple selection of files for deletion. Select the file(s) and/or directory(s) to be deleted, and press **OK** (or **Cancel** to abort). Be careful! You don't get a second chance to change your mind once you've pressed **OK**. Once a file is deleted, it's gone. Remember the fundamental directive of computer users: Save early, save often; make backups.

```
Dir:\          Sel:2/4      Index: 2
```

```
File to delete:BLUES      (dir)
                MOTOR     .KRZ*  98K
                QUACKS    .KRZ   344K
                ZAPPER    .KRZ*  802K
```

```
Total:1244K
```

```
Select Root Parent Open OK Cancel
```

A directory must be empty in order to delete it (i.e. it must not contain any files or subdirectories.)

## Backup and Copy Functions

### File Backup

To access the Backup function from the Disk Mode page, first make sure that the Current Disk is set to be the drive that you want to make a backup of. Next, press the **Backup** soft button:

```

Dir:\                               Sel:0/15       Index:    1

Set backup dir:ANIMALS.KRZ        1097K
                        BREAKAGE      (dir)
                        LOWINST        (dir)
Total:9040K                    PLANKTON  (dir)
Root Parent UPen              Current Exit

```

On this screen, choose the directory tree that you want to backup by selecting a directory (see Selecting Directories, near the beginning of this chapter.) Backup allows to you copy all of the files within a directory tree from one disk to another. All of the files within the directory that you choose, plus all of its subdirectories and files within them will be copied to the new drive. If you want to backup the entire disk, then make sure the current directory is the root directory (as in the picture), and press **Current**.

Next, you will see a dialog for choosing the destination disk:

```

Destination disk:SCSI 0
                  SCSI 1
                  SCSI 2
                  SCSI 3
                  OK Cancel

```

Select the disk you wish to transfer files to. It must be a different disk than the current disk.

Next, you can select a directory on the destination disk that will receive the transferred files.

```

Use default directory on SCSI 0?
(Path = \)

```

```

Change OK Cancel

```

The default is always the root directory on the destination disk. Press **OK** to select the default. To select a different directory, press **Change**.

Next, select the **Backup mode** when you see the following question:

## Replace or increment mode?

**Help** **RePlace** **Increm** **Cancel**

**Replace** Any files to be transferred that already exist in the destination directory will be replaced (overwritten.)

**Increm** Any files to be transferred that already exist in the destination directory will be skipped (not transferred.)

**Help** Displays a reminder about the meaning of Replace and Increment modes.

After you select the Backup mode, you will see a confirmation screen with all of your selections so far:

```
Press OK to start backup:SCSI 4->SCSI 0
Mode =RePlace
Source=\
Dest  =\
```

**SetFile** **OK** **Cancel**

**OK** Begin the Backup function according to the parameters on this page.

**Cancel** Exit to Disk Mode.

**SetFile** Set Backup starting file. This is mainly used when you are backing up files from a larger SCSI drive to several smaller removable media disks. Once one of the removable destination media is full, note the last file that was successfully copied. Enter the Backup function again, setting up the source and destination path parameters exactly as before. But this time use the **SetFile** button to start from the file after the last file copied.

You will notice that the file list displayed when SetFile is pressed is not in alphabetical order, but in the actual order the files occur in the directory on the disk:

```
Dir:\          Sel:0/15  Index: 1
```

```
Set start file:ZYXWVUTS .KRZ 418K
                PLANKTON (dir)
                ANIMALS .KRZ 1097K
Total:9040K      VEGGIES .KRZ 2801K
                Root  Parent  UPen  OK  Cancel
```

Find the last file copied from the previous partial backup. Set the list index to one entry past the location of this file. If this file is already the last entry in the file list, the "next" file to continue the multi-part backup would be the file or directory that comes after the file list entry

for the currently viewed directory. To find this file or directory, you will need to look in the Parent directory.

A large SCSI drive backed up in this way can be restored by individually backing up the removable media in the reverse direction.

Backup and Copy will transfer files much faster if you have cleared your object RAM first using the Delete Everything command (save any objects in RAM to disk first.)

When using the Backup function to transfer files from SCSI to Floppy, directories are ignored (i.e. not created on the floppy.) This "flattens" out any directory structure so that multi-disk files can be created on the floppy disks.

## **File Copy**

The Copy function lets you copy a single file from a directory on one disk to a specified directory on the destination disk.

## Creating a Startup File

You can create a Macro file that will automatically be loaded in when you power-on your K2500. This file, called the **Startup file**, or **Boot Macro**, can be on a disk drive at any SCSI ID or else on the Floppy disk. See the section on Macros for background information.

The steps needed to create a Startup file are:

First, create a Macro file called **BOOT.MAC** in the root directory of the SCSI or Floppy disk that you will use as the Startup disk. Specify in the Macro the exact ordering of files that you would like to have loaded into the K2500 when powered on. When you save the Macro file, just name the file "BOOT", and the K2500 will add the ".MAC" extension.

Second, set the **Startup** parameter on the Disk Mode page to be the drive ID of the Startup disk. So, if your BOOT.MAC file was on a floppy disk, set the Startup parameter to **Floppy** and make sure that you have the correct floppy in the drive when you next turn the K2500 on.

When the K2500 is powered on, it will display the following message (after the introductory VAST logo):

About to load startup file...

**Cancel**

The K2500 looks for a file BOOT.MAC in the **root directory** on the drive specified by the Startup parameter. If the file is not found, or the drive cannot be accessed, you will get an error message. The Startup load can be bypassed in the first few seconds after the K2500 is turned on, by pressing the **Cancel** button that is displayed on the screen.

If BOOT.MAC is found by the K2500, it will begin to load the Macro file as if you had loaded it explicitly from the Load function in Disk Mode. When the Macro has completed, you will see the following:

Macro BOOT.MAC completed...

The K2500 will go directly to Program Mode afterwards.

## Deleting Banks in a Startup File

You may want the Startup file to clear out one or all banks in the K2500 before loading files. This could help overcome the problem of having "silent" copies of programs in your RAM that depend on samples that are no longer there (because they disappeared the last time the K2500's power was turned off.) The following trick will allow a Macro entry to essentially function as a Delete Bank or Delete Everything command:

Create a file somewhere on (preferably) your Startup disk, by saving an empty bank from the K2500. Call the file NULL.KRZ. Now, insert this file at the beginning of a boot Macro you are creating - load the file, specify the bank you want to delete in the Startup file (choose

'Everything' to clear out all of the object RAM), and specify **Overwrite** for the load mode. Make sure you press **Macro** and not **OK**, so that the overwrite doesn't take place until you use the Startup file.

Here is what that Macro entry for this file might look like on the Macro page, if you were doing a "Delete Everything"

**3: \NULL.KRZ**

**E:O:**

The "E:O" stands for "Load as Everything, using Overwrite mode".

## MS-DOS File System Compatibility

The K2500 is compatible with fixed and removable disk drives that use the MS-DOS hard disk and floppy disk formats. If you want to use this feature, **you must first format the disk media on a computer** such as a PC compatible or a Mac running appropriate MS-DOS conversion software.

The MS-DOS hard disk format is structured so that the disk can be split up into multiple partitions. **The K2500 can only use the first partition that it finds on the disk.** Therefore, it is usually best to format the media with only one partition taking up all usable space on the disk. Working from the K2500 front-panel with an MS-DOS formatted drive will appear the same as working with a drive that has been formatted with the K2500's own Format function (on the Disk Mode page.) The Free utility (Disk Mode->Util->Free) can be used to identify whether a disk is DOS format or standard K2500 format. If the "DOS" indicator is displayed, it means the K2500 has determined that the disk is a DOS format hard drive with at least one primary DOS partition. Floppy disks will not display the DOS message.

Some advantages of working with an MS-DOS compatible drive format over the standard K2500 format are:

- easier sharing of K2500 files with other users over computer communications lines,
- being able to use graphical file management interfaces for organizing files and directories,
- being able to back up K2500 data using a PC compatible or Mac with commercially available software, and
- easier transfer of data using standard file formats such as AIFF, WAVE, and MIDI Type 0, for importing and exporting samples and sequences.

### File Name Compatibility

**DOS format does not support space characters in file names.** The K2500, though, allows spaces to be used within file names. If you plan to transfer files between the K2500 and a DOS compatible computer, it is recommended that you use only file names without space characters in them. Otherwise, a computer may have trouble identifying the files.



## Importing and Exporting Data using Standard File Formats

The K2500 supports three common data interchange file formats, Apple Interchange File Format (AIFF), Microsoft RIFF WAVE, and MIDI Type 0. The first two are used to transfer sample data, and the latter is used for sequences.

The K2500 can recognize these file types automatically on loading, regardless of the file extension. You can load these files as you would any standard K2500 file, and also as part of a Macro file load. The most recent sample file loaded will become the "preview" sample, which means you can quickly access it for playing or editing on the Sample page (**MASTER->Sample**). Similarly, the most recent MIDI Type 0 file loaded in will become the current song on the Song Mode page.

You can save files in these formats on the **Export** page. This page is accessible from the Disk Mode page by pressing **Save->Export**.

```
Save AIFF file          Index: 1

Sample 200 NewSample    C 4S 100K
Sample 201 Babaganoush  C 4S 444K
Sample 202 YaYaYa       C 4S 190K
Sample 203 GregorianChant C 4S 6515K
AIFF    WAVE    MIDI    OK    Cancel
```

The Export page allows you to save one sample or song object per file. Choose the format you wish to save in, and press the corresponding soft button. For AIFF and WAVE, only sample objects are listed. For MIDI Type 0, only song objects are listed. Scroll to the object that you wish to save, and press **OK**. The dialog will proceed the same as if you were saving a K2500 file. You will be prompted for a file name, and will have the option to select a different default directory to save the file in.

The K2500 will automatically place a standard extension on the file when it is created on the disk. These extensions are sometimes necessary when transferring files to external programs that can recognize the file format based on the extension. They also help you to recognize what files are of a what format when looking at a directory listing. You can use the Find Files utility (Disk Mode->Util->Find) to search for files that match a certain extension. The standard extensions used on the files are:

```
.AIF    AIFF
.WAV    WAVE
.MID    MIDI Type 0
```

The first time you enter the **Export** page after powering on (or after a soft-reset of the K2500), the format defaults to AIFF. After that, the K2500 remembers the most recent format that you used. For example if you save a MIDI Type 0 file, and then go back to the Song Mode page to record more sequences, the next time you return to the Export page, the file format will still be set to "MIDI", and all of the song objects will be listed. You can audition the samples and songs the same way as you would on the Save Object or Object Utility pages (by pressing the Left or the Right cursor buttons on the front-panel.).

### AIFF and AIFF-C Files

The K2500 can read 8 or 16 bit AIFF files, mono or stereo. The sample rate, sustain loop, loop mode, base note, sample name, and sample detuning are supported. **AIFF-C** files that do not use compression can also be read by the K2500. The K2500 will save 16-bit AIFF files, either mono or stereo.

## **WAVE Files**

The K2500 can read 8 or 16 bit WAVE files, mono or stereo. WAVE files do not support looping. The K2500 will save 16-bit WAVE files, either mono or stereo.

## **Standard MIDI Files (MIDI Type 0 and Type 1 Files)**

The K2500 reads MIDI Type 1 files (multi-track) and can read or write MIDI Type 0 files (single track). The K2500 supports all musical timestamp resolutions, and automatically scales imported information to the K2500's internal sequencer resolution of 768 ticks per beat, if necessary.

Tempo information is supported, defaulting to 120 beats per minute if no tempo is specified.

If there is a time signature event in a file, the first one becomes the song's time signature. Otherwise, the time signature defaults to 4/4. Time signature changes are not supported.

Apart from tempo and time signature events, all other meta events, as well as System Exclusive data, are skipped in order to preserve RAM.

## Chapter 14

# DSP Functions

This chapter presents explanations of the DSP functions that can be inserted into the algorithms in the Program Editor. As you configure each algorithm, the DSP functions you select determine the type of synthesis you apply to your sounds. Deciding which algorithm to use depends on what you want to do; there's no hard and fast rule. If you want to create a classic analog sound, for example, you'll choose one of the algorithms containing one or more blocks that can have filter functions assigned to them. If you want realtime panning effects, choose an algorithm that includes the PANNER function in the F3 block. Your best approach is to study the algorithm charts in the *Reference Guide*, and choose the algorithm that includes the functions you want to work with.

Before we get to the explanations of the DSP functions, we've included a brief discussion of a few general concepts of sound synthesis. This should help you understand the workings of the DSP functions. We'll refer to these concepts repeatedly as we go along.

Any single sound waveform is composed of numerous sine wave components, each at a different frequency. These components are called partials. The lowest frequency is perceived by the ear as the pitch of the sound, and is called the fundamental. The other components are called harmonics. The relative amplitudes (volume) of each of the partials in a sound determine its timbre, its most recognizable characteristic. When you think of the difference between the sound of a piano and a saxophone, you're thinking about their different timbres. A dull sound has a strong fundamental and weak harmonics, while a bright sound has strong harmonics.

Sound synthesis can be most simply described as the manipulation of either the amplitude or phase of one or more of the partials constituting a sound. The K2500's various DSP functions give you a variety of methods for manipulating those partials. We've grouped our explanations of the DSP functions according to the types of specialized manipulation they enable you to perform on a given sound. The categories are:

- FILTERS
- EQUALIZATION (EQ)
- PITCH / AMPLITUDE / PAN POSITION
- MIXERS
- WAVEFORMS
- ADDED WAVEFORMS
- NON-LINEAR FUNCTIONS
- WAVEFORMS WITH NON-LINEAR INPUTS
- MIXERS WITH NON-LINEAR INPUTS
- SYNCHRONIZING (HARD SYNC) FUNCTIONS

## Introduction to Algorithm Programming

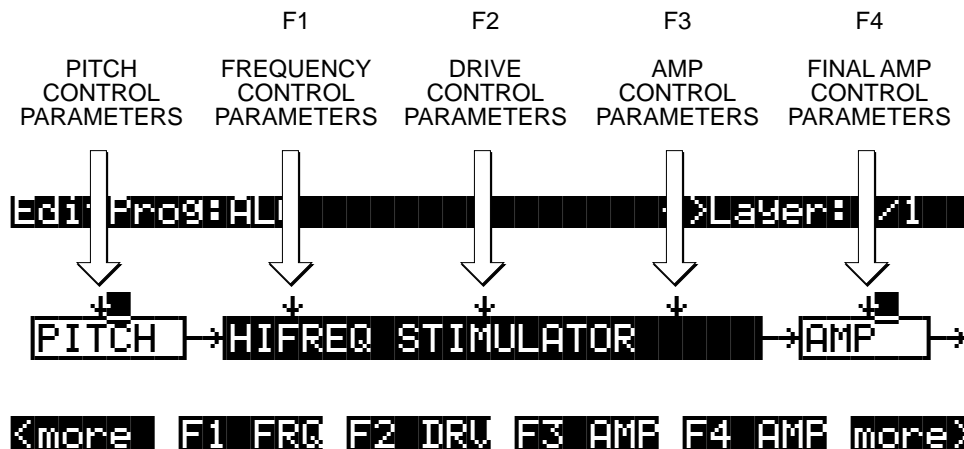
Programming the algorithms is a multi-step process. The first step is selecting an algorithm. Changing the algorithm of an existing program's layer is likely to alter the sound of the layer dramatically. As a rule, then, you won't want to change a layer's algorithm unless you're building a sound from scratch. Furthermore, when you change a layer's algorithm, the values for each of the DSP functions within the algorithm may be set at non-musical values; you should lower the K2500's volume slider before changing algorithms.

Deciding which algorithm to use for a new sound is primarily a process of planning a layer's signal path through the sound engine. The real sound manipulation is done by the DSP functions you insert into the algorithm. The algorithm simply lays a framework that determines how the DSP functions interact.

Once you know which algorithm you're going to work with, you'll assign various DSP functions to each of the stages of the algorithm. These stages, as you recall, are represented by the rectangular blocks you see on the ALG page. The arrows pointing down at the blocks represent control inputs that affect the behavior of the DSP functions. For each arrow, there's a page of parameters controlling some aspect of the DSP function's behavior. Every DSP function has at least one control input; several have two or three.

The ALG page is where you select algorithms and assign DSP functions to the algorithm's various stages. To assign a DSP function, move the cursor to select the stage you want to modify, then use any data entry method to scroll through the list of available DSP functions for that stage. You'll normally hear the effect of each selection as soon as you make it. If you don't hear a difference, it's because the function's control parameters aren't set to significant values. Once you adjust some of these parameters, the function will have a noticeable effect on the sound. Keep in mind that not all DSP functions are available at every stage of every algorithm.

When you have each stage of the current algorithm set up to your liking, you can begin to program the control inputs of each DSP function. This is done by selecting the control input page(s) for the currently selected DSP function, and adjusting the parameters on the page. There are two ways to select the control input pages: you can move the cursor to select the DSP function you want to tweak, and press EDIT. The selected DSP function's control input page will appear (if it's a multi-stage DSP function, its first control input page will appear). Or you can use the soft buttons to select the pages. The PITCH soft button always selects the pitch control input page, since the first stage of every algorithm is invariably the pitch control. The F1—F4 soft buttons select the control input pages corresponding to the remaining four arrows, which point down at the subsequent four variable control inputs.



Each control input page contains several parameters, which affect some aspect of the behavior of the DSP function named on the top line of the page. Most of these parameters are the common DSP control parameters; for a review, see "Common DSP Control Parameters" in Chapter 6.

The possibilities are truly enormous, given the number of different combinations of functions you can assign to any particular layer (not to mention multi-layer programs, each layer of which has its own algorithm). You can create completely new sounds just by tweaking the parameters on the control input page for a single DSP function. When you begin adjusting these parameters, it's a good idea to start with all of them set to 0 (or the value that minimizes their effects), then adjust them one by one. This will help you understand exactly what effect each parameter has, and will give you an idea of the variety of effects each parameter can produce. Then you can start combining the effects of multiple parameters, and quite possibly take your sound in a whole new direction. You'll quickly become familiar with the control input pages for the DSP functions, since most of them contain the same parameters, with just a few variations. You'll find that they all look much alike. The top line of each page, however, will indicate which DSP control input it represents (PITCH, or F1—F4).

For example, on the page below, the top line tells you that the currently selected DSP function is the HIGH FREQUENCY STIMULATOR—its name is abbreviated and enclosed in parentheses. You can also see that you’re looking at F1, which in this case controls the frequency of the HIGH FREQUENCY STIMULATOR. So the top line of these pages always shows three things: 1) the currently selected control input (PITCH or F1–F4); 2) the aspect of the current DSP function controlled by the input (this varies depending on the current DSP function); 3) the currently selected DSP function (usually abbreviated, and in parentheses). Items 1) and 2) match the label of the soft buttons that select each page. The page below, for example, is selected with the soft button labeled “F1 FRQ.”

```

EditProg:F1 FRQ(HIFRQ STIM)<>LAYER:1/1
Coarse:C 4 262Hz Src1 :OFF
Fine :Oct Depth :Oct
KeyTrk:Oct/key Src2 :OFF
VelTrk:Oct DptCt1:OFF
Pad :0dB MinDpt:Oct
MaxDpt:Oct
<more> F1 FRQ F2 DRU F3 AMP F4 AMP <more>

```

## Additional Parameters

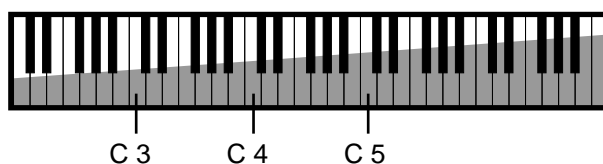
In addition to the common DSP control parameters you’ll find on each page, you’ll also see a few others on various pages. They’re described here, since programming them is the same regardless of the page on which they appear. Depending on the DSP function they affect, they’ll have different ranges of values and different units of measurement (% , dB , etc.).<sup>6</sup>

### Pad

Many of the DSP functions boost the signal as it passes through. Depending on the signal’s input level and the amount of gain (boost) introduced by any given DSP function, its output may clip, which will alter the sound considerably. Clipping may also occur as a result of phase shifting, but this is not as common as clipping caused by gain. While you may find clipping to be a useful component of some sounds, you’ll want to remove it from others. The Pad parameter, which appears on the control input pages of many DSP functions, lets you attenuate (reduce the amplitude of) the signal by 6, 12, or 18 dB at the input of those functions. Use the Pad parameter to reduce or eliminate any undesired clipping caused by the currently selected DSP function.

### Key Track Start (KStart)

This parameter appears on many control input pages, and gives you added control over the effect of Key Tracking. For each note you play, it multiplies the value of the KeyTrk parameter by a number that varies with the note’s MIDI key number. If KeyTrk is set to 0, this parameter will have no effect. When KeyTrk is a non-zero value, KStart will modify the normal key tracking curve, which is shown in the diagram below. The effect of normal key tracking reaches its minimum at C -1, and its maximum at C 9. You can use KStart to dampen the effects of key tracking at one end of the keyboard. If key tracking causes a sound to clip or distort toward the high end of the keyboard, for example, you can use KStart to reduce the effect of the key tracking at the upper end without changing its effect on the lower end. To do this you would set a negative value for KeyTrk, and a unipolar value for KStart.



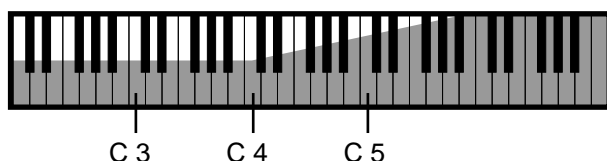
Normal KeyTrk curve  
(Positive KeyTrk value)

## UNIPOLAR KEYSTART

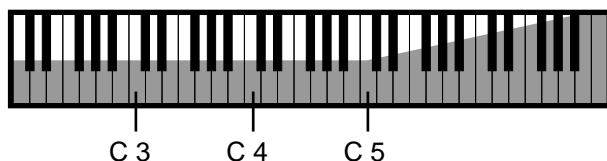
The range of values for KStart is C -1–C 9 unipolar, and C -1–C 9 bipolar. Unipolar and bipolar values have different effects on the key tracking. The next three diagrams illustrate the effect of three different *unipolar* keystart values on the key tracking curve when a positive value is assigned for the KeyTrk parameter. At a KStart value of C 4, for example, there is no key tracking effect below Middle C (it multiplies the key tracking amount by a key number value of 0). The key tracking value is multiplied by 0 at C 4 (normal key tracking), by 1 at C# 4, by 2 at D 4, and so on to a maximum of 64 at 5 1/3 octaves above the KStart value. For higher notes, the key tracking effect would still increase on its own, but the effect of keystart would not increase further. At a KStart value of C 3, the key tracking value would be multiplied by 0 for C 3 and all notes below, by 1 for C# 3, and so on. The key number value would reach its maximum of 64 before reaching C 9. When KStart is set above C 4, its effect on key tracking will continue to increase up to C 9, but will not reach full scale at C 9.

You'll use unipolar values for KStart when you want to cancel the key tracking effect on a DSP function over a sizable portion of the keyboard, but have it increase or decrease throughout the rest of the keyboard's range. Set high unipolar values for KStart when you want to remove key tracking from the lower notes, applying it only to the higher notes. If you have set a positive value for KeyTrk, set low unipolar values when you want to apply key tracking to the lower notes and pin it at its maximum throughout the upper range of the keyboard. You may want to use low values of key tracking in this case, depending on the DSP function you're applying.

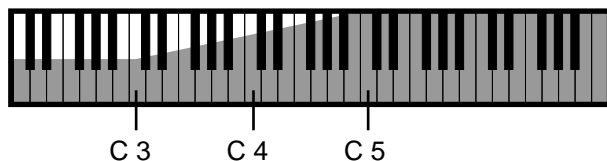
When the value of the KeyTrk parameter is negative, remember that the key tracking is at its minimum effect at C 9, and maximum at C -1. In this case, the key tracking effect will be reduced for notes above the KStart setting. For notes below the keystart note, the normal key tracking amount will apply.



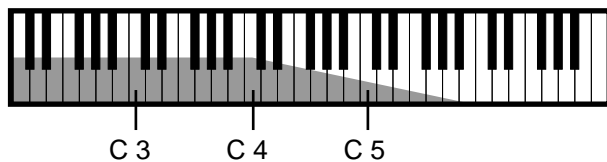
Positive KeyTrk value with  
KStart value at C4 Unipolar



Positive KeyTrk value with  
KStart value at C5 Unipolar



Positive KeyTrk value with  
KStart value at C3 Unipolar

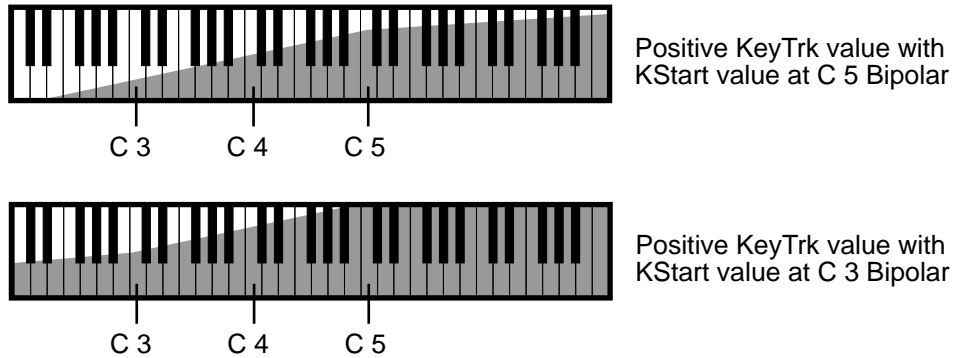


Negative KeyTrk value with  
KStart value at C4 Unipolar

## BIPOLAR KEYSTART

For bipolar KStart values with positive key tracking values, the effect on key tracking depends on whether the KStart parameter is set above or below C 4. When it's above, the effect on key tracking will be minimum at C -1, reaching its maximum effect on key tracking at the keystart

setting. The normal key tracking curve applies above the keystart setting. When KStart is set below C 4, the effect on key tracking is maximum at C 9, decreasing with each successive note closer to the keystart setting, and remaining constant at the keystart setting and below. The normal key tracking curve applies below the keystart setting.



Use bipolar settings for KStart when you want to gradually increase or decrease the key tracking effect of the currently selected DSP function across the entire keyboard range. With KStart at C 4 bipolar, playing C 4 will apply the DSP function at the level you set with the Adjust parameter, and will increase or decrease with higher or lower notes, depending on your settings for KeyTrk.

When KeyTrk is set to a negative value, the effect on key tracking is reversed. For keystart settings above C 4, the effect on key tracking will be maximum at C -1, decreasing with each note closer to the keystart setting, and remaining constant at and above the keystart setting. For keystart settings below C 4, the effect on key tracking will be minimum at C 9, increasing with each note closer to the keystart setting, and remaining constant for notes at and below the keystart setting.

KStart is available for many of the *non-linear DSP functions*, like SHAPER and WRAP. If you like the control it gives you, you can simulate its effect by using the FUNs. To simulate unipolar keystart, assign Key Number (KeyNum) as one of the inputs to a FUN, and select one of the diode equations for the FUN's Function parameter. To simulate bipolar keystart, assign Bipolar Key Number (BKeyNum) as one of the inputs of a FUN. Then assign those FUNs to some other control source parameter.

## The DSP Functions

### Filters

ONE-POLE LOWPASS  
 TWO-POLE LOWPASS  
 TWO-POLE LOWPASS, -6 dB RESONANCE  
 TWO-POLE LOWPASS, +12 dB RESONANCE  
 FOUR-POLE LOWPASS WITH SEPARATION  
 GATED LOWPASS  
 ONE-POLE HIGHPASS  
 TWO-POLE HIGHPASS  
 FOUR-POLE HIGHPASS WITH SEPARATION  
 ONE-POLE ALLPASS  
 TWO-POLE ALLPASS  
 TWO-POLE NOTCH  
 TWO-POLE NOTCH, FIXED WIDTH  
 DOUBLE NOTCH WITH SEPARATION  
 TWO-POLE BANDPASS  
 TWO-POLE BANDPASS, FIXED WIDTH  
 TWIN PEAKS BANDPASS

Filters are widely used in synthesis to change the timbre of a sound by manipulating the amplitude of specific partials. When using filters, you always set a reference point (cutoff or center frequency) that determines which partials the filters affect. Here's a quick summary of the effects of the filter functions.

**Lowpass** filters cut the levels of all partials above the cutoff frequency without affecting the partials at or below the cutoff frequency (the low frequencies pass through). **Highpass** filters do the opposite; they cut the levels of all partials *below* the cutoff frequency without affecting the partials at or above the cutoff frequency.

**Notch** filters, as the name implies, cut the levels of partials in a range between high and low frequency. Consequently the "cutoff" frequency is referred to as the center frequency. With notch filters, the levels of partials at the center frequency are cut, while the levels of partials above and below the center frequency are unaffected. **Bandpass** filters are the opposite of notch filters; they leave the levels of partials at the center frequency unchanged, and cut the levels of partials above and below the center frequency.

The use of lowpass, highpass, notch, and bandpass filters is often referred to as subtractive synthesis, since the timbre of a sound is changed by removing certain partials.

Allpass filters, instead of cutting or boosting the partials of a sound, change the phase of the partials as their frequencies pass through the center frequency.

### Filter Terminology

**Rolloff** - Filters do not usually cut all frequencies precisely at their cutoff point. Instead, the amplitude of the frequencies above (or below, in case of a hi pass filter) the cutoff decrease by a fixed amount per octave - for example, 6 dB per octave. This curve of lessening amplitude is called a rolloff.

**Poles** - The number of poles in a filter affect how sharp the rolloff is. The more poles there are, the sharper the rolloff, meaning that the cutoff will have a more dramatic effect on the sound. The K2500 has one pole, two pole, and four pole filters available. A one pole filter has a 6 dB per octave cutoff; a two pole is 12 dB per octave; and a four pole is 24 dB per octave.

**Resonance** - In a filter that has resonance, the frequencies near the cutoff are given an increase or decrease in amplitude. If you decrease these frequencies, you are essentially creating a longer rolloff. But if you increase those frequencies thereby emphasizing them, it creates a distinctive sound that you will very likely recognize. Resonance is also sometimes called Emphasis or Q on various synthesizers. Resonance on the K2500 is implemented in one of two ways. On some filters, the resonance is fixed, adding or subtracting a specific amount of dB to the affected frequencies (the ones near the cutoff). On other filters, you can control the amount of resonance applied. In the case of these filters, there will always be a separate control page for the resonance.

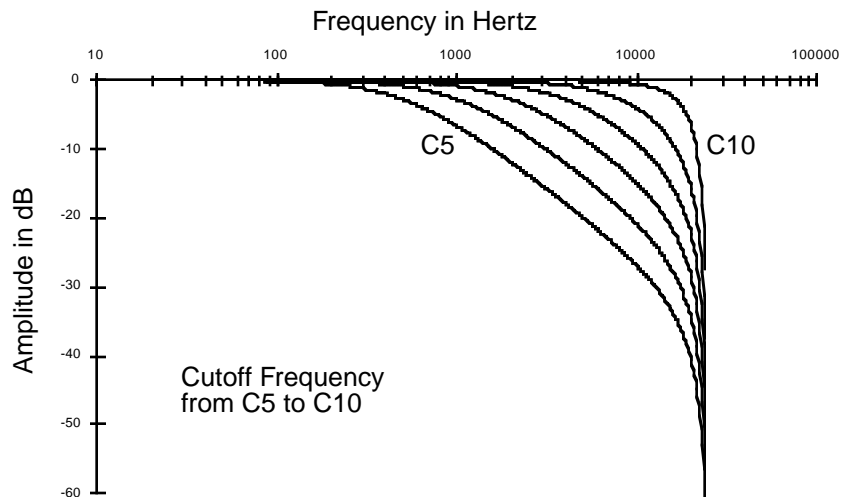
**Separation** - Four of the filters in the K2500 (both Four Pole filters, the Double Notch, and the Twin Peaks) are actually two filters combined into one DSP function. For these filters, you will find a control page called Separation. This allows you to shift the cutoff frequency of the second filter, creating a separation in the cutoff frequencies of the two filters. In the case of the Notch and Band Pass filters, this can be used to create two separate notches or band passes. In the case of the four pole filters, it affects the shape of the roll off. For the four pole filters, separation set to 0 creates sharp rolloff of 24dB per octave.

## How to read the graphs

The graphs show the rolloff curve, using several different values to show how they change the shape of the curve. Amplitude is always on the vertical axis. Frequency is always on the horizontal axis. You will notice on several graphs that the curve becomes more dramatic as the cutoff frequency is set at a higher value. This is because the highest frequency the K2500 can produce is 20Khz, so as the cutoff is set to higher values, there are fewer frequencies available before it is past the range of the K2500.



## One-pole Lowpass Filter (LOPASS)



Frequencies below the cutoff frequency are unaffected by this filter. At the cutoff frequency, the signal is attenuated 3 dB. There's a rolloff of 6 dB per octave above the cutoff frequency—that is, the signal is attenuated 6 dB with each octave above the cutoff. The resonance—the amount of cut or boost at the cutoff frequency—is fixed at -3dB. When the cutoff frequency is well below the the lowest-frequency partials of a sound, lowering the cutoff further will not affect the timbre of the sound, but will reduce its overall amplitude.

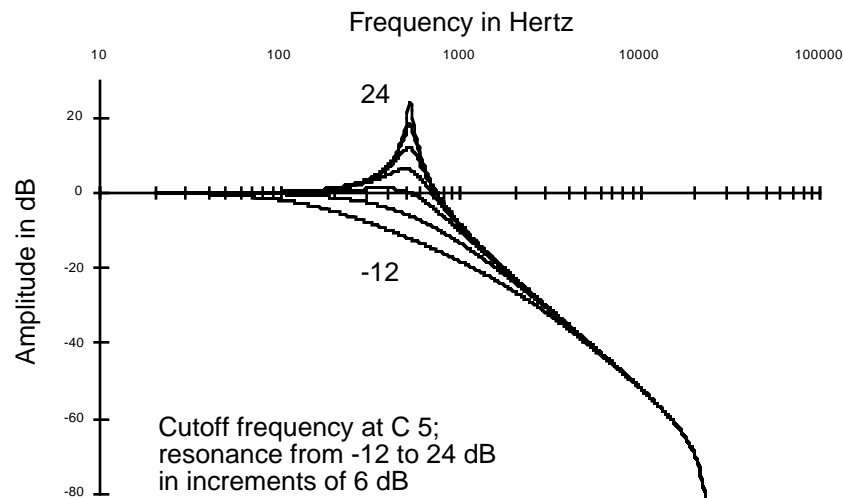
```

EditProg:F1 FRQ(LOPASS) <>Layer:1/1
Coarse:C 4 262 Hz Src1 :OFF
Fine :0ct Depth :0ct
KeyTrk:0ct/key Src2 :OFF
VelTrk:0ct DptCt1:OFF
Pad :0dB MinDpt:0ct
MaxDpt:0ct
<more F1 FRQ F2 F3 F4 AMF more>
  
```

PARAMETER	RANGE OF VALUES
COARSE ADJUST	C 0 16 Hz to G 10 25088 Hz
FINE ADJUST	± 100 cents
KEY TRACKING	± 250 cents per key
VELOCITY TRACKING	± 10800 cents
PAD	0, 6, 12, 18 dB
SOURCE 1	Control Source list
SOURCE 1 DEPTH	± 10800 cents
SOURCE 2	Control Source list
SOURCE 2 DEPTH CONTROL	Control Source list
MINIMUM DEPTH, SOURCE 2	± 10800 cents
MAXIMUM DEPTH, SOURCE 2	± 10800 cents

The Coarse Adjust parameter sets the cutoff frequency in terms of a key name. The remaining parameters (except Pad) alter the cutoff frequency in increments of cents. You'll notice that positive values for key tracking have an interesting effect on the function of lowpass filters; positive key tracking values raise the cutoff frequency for high notes and lower it for low notes. More specifically, a value of 100 cents per key on this page, when filtering a constant waveform like a sawtooth, would result in waveforms of exactly the same shape for all pitches of the waveform. The cutoff frequency moves in sync with the frequencies of the waveform's partials as different pitches are generated. Negative key tracking values will steepen the rolloff of lowpass filters above the cutoff. The Pad parameter, as always, attenuates the signal at the input to the function. These parameters affect all the lowpass filters similarly.

### Two-pole Lowpass Filter (2POLE LOWPASS)



The two-pole lowpass filter has a rolloff of 12 dB per octave above the cutoff frequency. This is a two-stage function, so it has two control input pages. The first affects the cutoff frequency, and has the same parameters as the one-pole lowpass. The second control input page (F2 RES) affects the resonance of the filter. Resonance is a cut or boost in amplitude of the partials in the vicinity of the cutoff frequency.

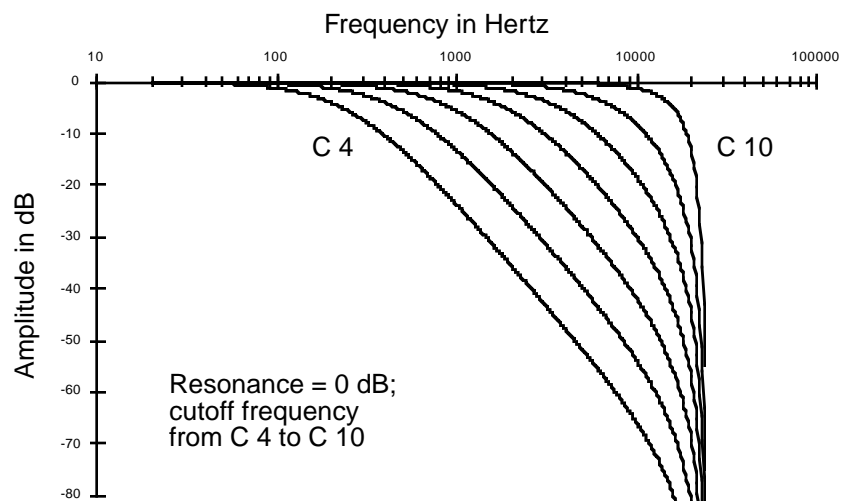
Set the resonance with the Adjust parameter on the F2 RES control input page, and use the other parameters to assign various controls to alter it. If a boost is applied at frequencies where there are high-amplitude partials, the signal may clip. The Pad parameter on the F1 FRQ page will reduce the clipping, but there's no harm in keeping it if you like the sound.

```

EditProg: F2 RES(2P LOWPASS) <> Layer: 1/1
Adjust: 0.0dB Src1 : OFF
Depth : 0.0dB Src2 : OFF
KeyTrk: 0.00dB/key DptCt1: OFF
VelTrk: 0.0dB MinDpt: 0.0dB
MaxDpt: 0.0dB
<more F1 FRQ F2 RES F3 F4 AMP more>
  
```

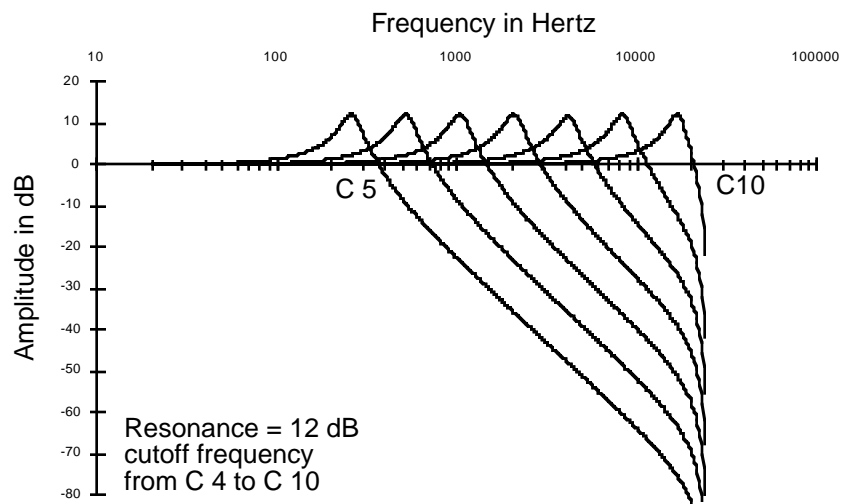
PARAMETER	RANGE OF VALUES
ADJUST	-12 to +24 dB
KEY TRACKING	$\pm 2.00$ dB per key
VELOCITY TRACKING	$\pm 30$ dB
SOURCE 1	Control Source list
SOURCE 1 DEPTH	$\pm 30$ dB
SOURCE 2	Control Source list
SOURCE 2 DEPTH CONTROL	Control Source list
MINIMUM DEPTH, SOURCE 2	$\pm 30$ dB
MAXIMUM DEPTH, SOURCE 2	$\pm 30$ dB

### Two-pole Lowpass Filter, -6 dB Resonance (LOPAS2)



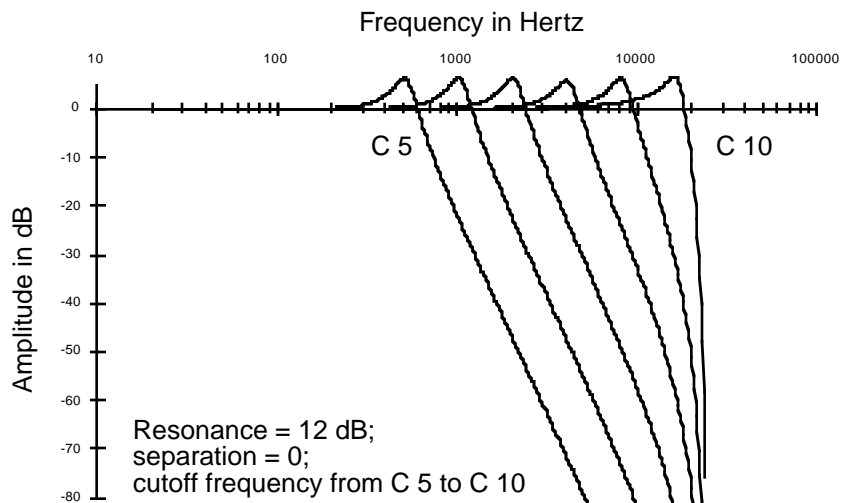
Using this filter is the same as using two one-pole lowpass filters in successive algorithm blocks. Since its resonance is fixed at -6 dB, it's also the same as using 2POLE LOWPASS with the resonance set to -6 dB. You'd use this filter when you want a 12 dB per octave rolloff but don't need to set a resonance level. This would leave you free to use another DSP function in the algorithm, since it's a one-stage function.

### Two-pole Lowpass Filter, +12 dB Resonance (LP2RES)

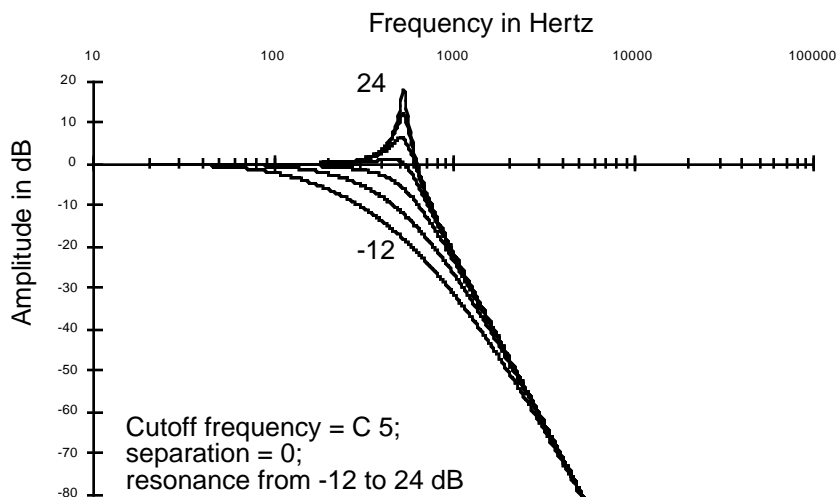


This is similar to LOPAS2; the only difference is that its resonance is fixed at +12 dB.

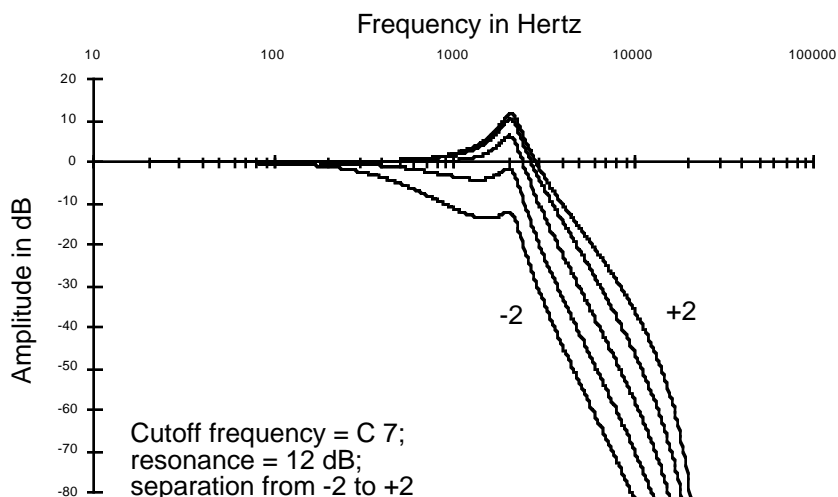
### Four-pole Lowpass Filter with Separation (4POLE LOPASS W/ SEP)



4-Pole Lowpass Filter:  
Resonance



4-Pole Lowpass Filter:  
Separation in Octaves



This combines 2POLE LOWPASS and LOPAS2 in one three-stage function. The parameters on the F1 FRQ control input page affect the cutoff frequencies of both filters. The parameters on the F2 RES page affect the resonance of 2POLE LOWPASS. The parameters on the F3 SEP page shift the cutoff frequency of LOPAS2, creating a separation between the cutoff frequencies of the two filters. Positive values raise the cutoff frequency of LOPAS2, while negative values lower it. If no separation is applied, there's a 24 dB per octave rolloff above the cutoff frequency.

```

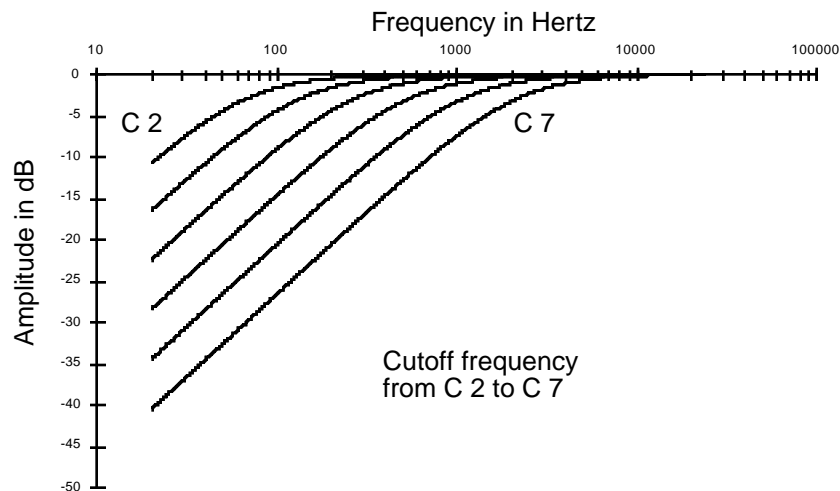
EditProg: F3 SEP(4P LOWPASS) (<>) Layer: 1/1
Adjust: 0ct Src1 : OFF
Fine : 0ct Depth : 0ct
KeyTrk: 0ct/key Src2 : OFF
VelTrk: 0ct DptCt1: OFF
MinDpt: 0ct
MaxDpt: 0ct
<more F1 FRQ F2 RES F3 SEP F4 AMP more>
    
```

PARAMETER	RANGE OF VALUES
COARSE ADJUST	$\pm 10800$ cents
FINE ADJUST	$\pm 100$ cents
KEY TRACKING	$\pm 250$ cents per key
VELOCITY TRACKING	$\pm 10800$ cents
SOURCE 1	Control Source list
SOURCE 1 DEPTH	$\pm 10800$ cents
SOURCE 2	Control Source list
SOURCE 2 DEPTH CONTROL	Control Source list
MINIMUM DEPTH, SOURCE 2	$\pm 10800$ cents
MAXIMUM DEPTH, SOURCE 2	$\pm 10800$ cents

### Gated Lowpass Filter (LPGATE)

You may be familiar with gates as applied to effects like reverb, where the effect shuts off abruptly after a specified time. The gated lowpass filter produces a somewhat similar effect in terms of the sound's amplitude. The filter's cutoff frequency is controlled by the AMPENV. When the AMPENV is at 100%, the cutoff frequency is high, so most of the partials are heard. When the AMPENV decays or releases to 0%, the cutoff frequency is low, so only the lowest partials are heard. You'll hear the distinct effect of the filter closing as the amplitude envelope releases.

### One-pole Highpass Filter (HIPASS)

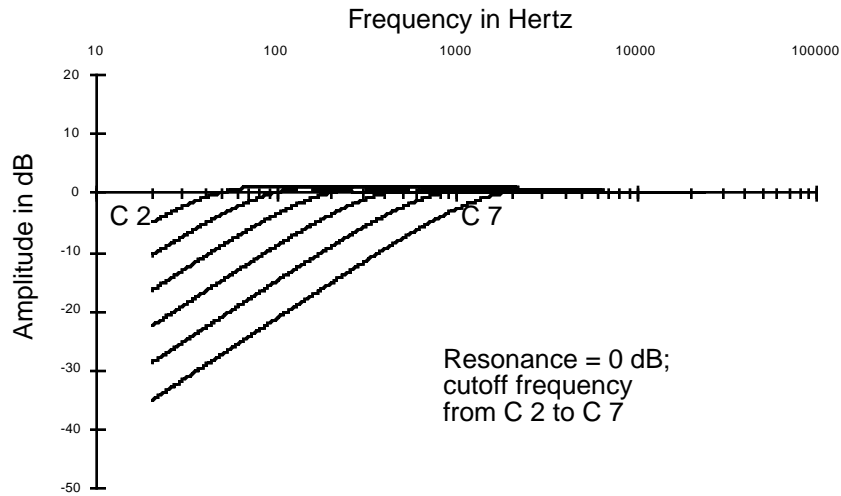


High-frequency partials pass through this filter unaffected. At the cutoff frequency, the signal is attenuated 3 dB. There's a roll-off of 6 dB per octave below the cutoff frequency. The resonance is fixed at -3dB. When the cutoff frequency is well above the the highest-frequency partials of a sound, raising the cutoff further will not affect the timbre of the sound, but will merely attenuate it further.

The Coarse Adjust parameter sets the cutoff frequency in terms of a key name. The remaining parameters (except Pad) alter the cutoff frequency in increments of cents. Positive key tracking values raise the cutoff frequency for high notes and lower it for low notes. More specifically, a value of 100 cents per key on this page, when filtering a constant waveform like a sawtooth, would result in waveforms of exactly the same shape for all pitches of the waveform. The cutoff frequency moves in sync with the frequencies of the waveform's partials as different pitches are generated. Negative key tracking values will steepen the rolloff of highpass filters below the

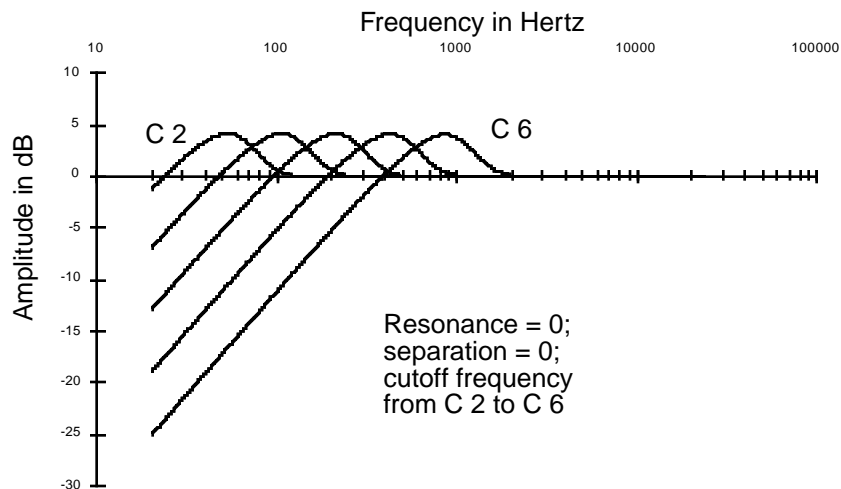
cutoff. The Pad parameter, as always, attenuates the signal at the input to the function. These parameters affect all the highpass filters similarly.

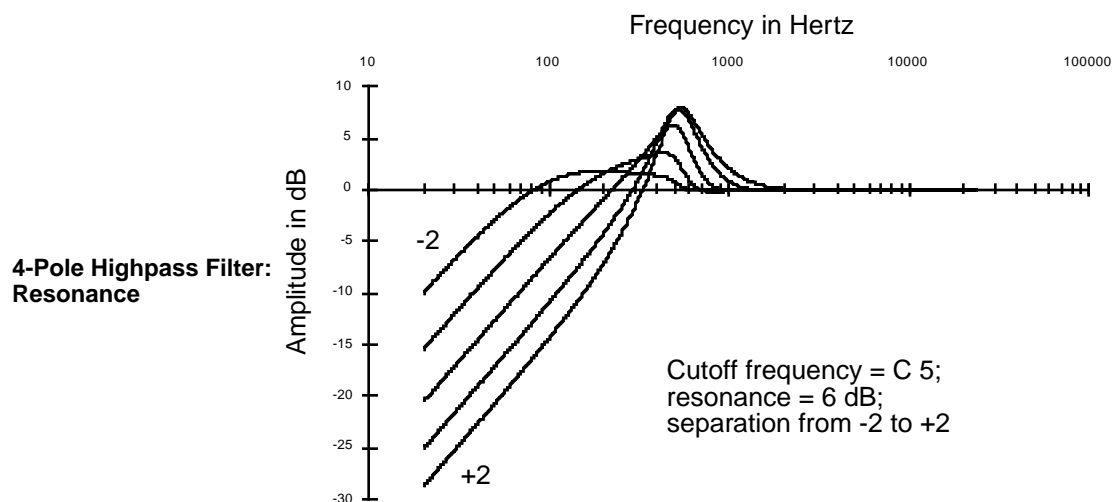
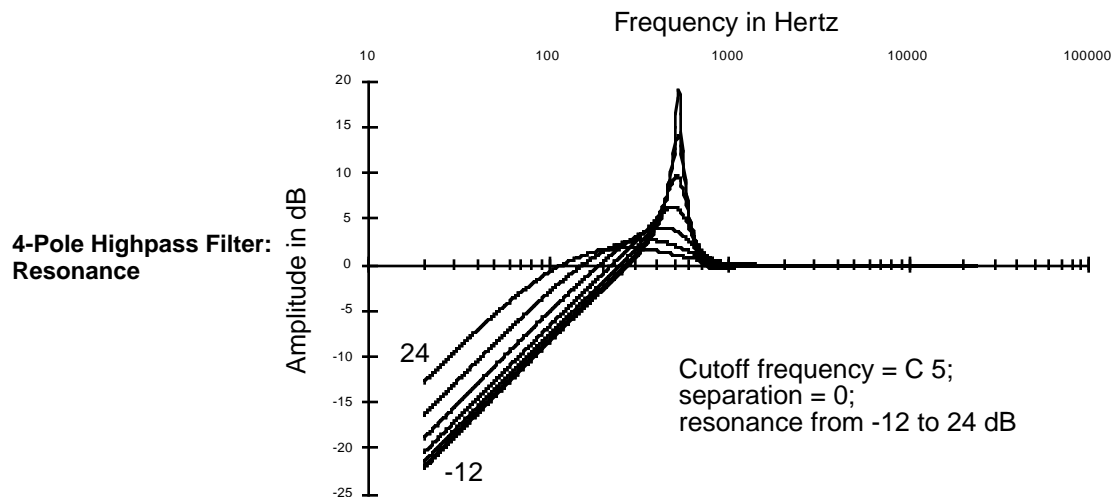
### Two-pole Highpass Filter (HIPAS2)



This is very similar to HIPASS. The primary difference is in the steepness of the rolloff at the cutoff frequency. Below the cutoff frequency, the rolloff is similar to that of HIPASS, except that there's a one-octave shift—that is, HIPASS with a cutoff frequency of C 3 will sound nearly the same as HIPAS2 with a cutoff of C 4. In other words, HIPASS gives you greater attenuation of low frequencies when set to the same cutoff frequency as HIPAS2.

### Four-pole Highpass Filter with Separation (4POLE HIPASS W/ SEP)

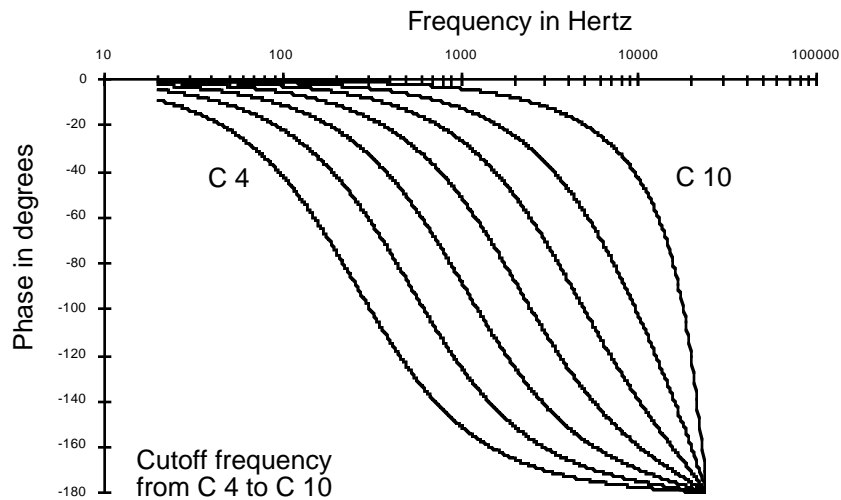




This combines two of the 2POLE HIPASS filters into one three-stage function. It has a rolloff of 6 dB per octave below the cutoff frequency. The parameters on the F1 FRQ control input page affect the cutoff frequencies of both filters. The parameters on the F2 RES page affect the resonances of the first filter. There will always be a slight extra boost of partials at the cutoff frequency, even at low resonance settings. The parameters on the F3 SEP page shift the cutoff frequency of the second 2POLE HIPASS, creating a separation between the cutoff frequencies of the two filters. Positive values raise the cutoff frequency of the second 2POLE HIPASS, while negative values lower it. If no separation is applied, there's a 24 dB per octave rolloff below the cutoff frequency. A variety of responses can be produced by adjusting the resonance and separation settings.



## One-pole Allpass Filter (ALPASS)

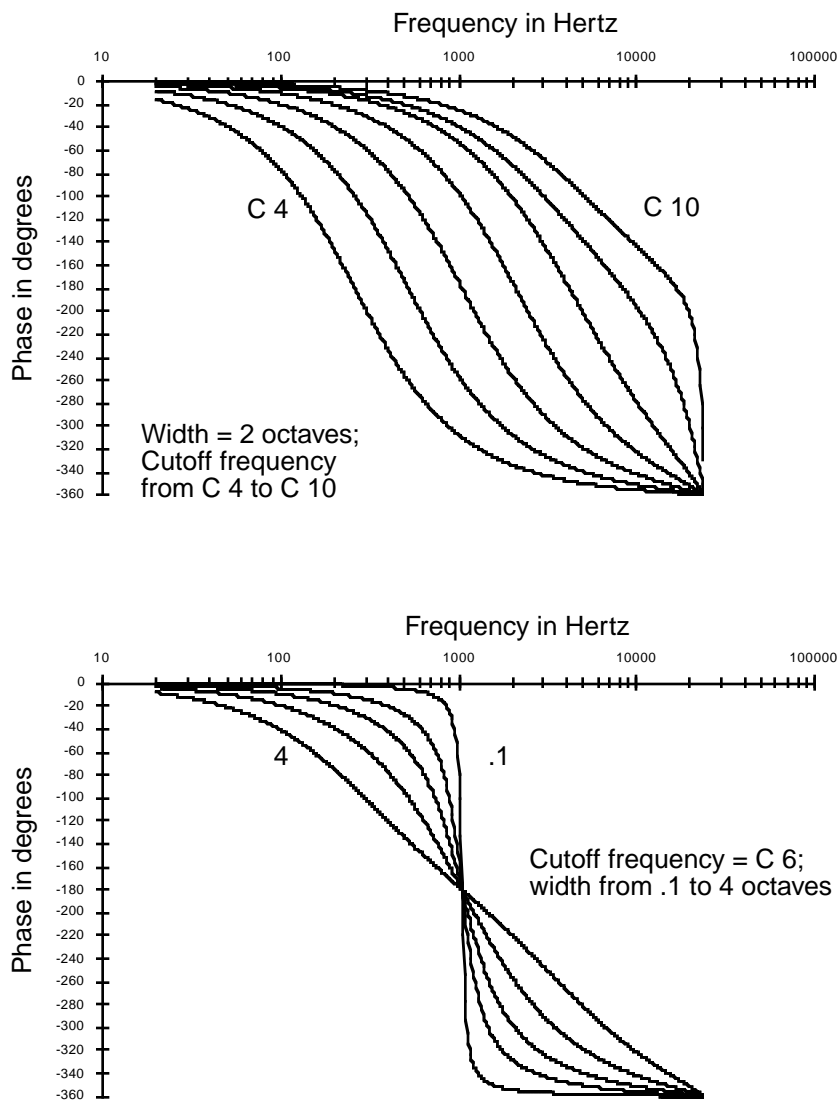


Allpass filters do not affect a sound's frequency response (the amplitude of partials at various frequencies), but change the phase of each partial depending on its proximity to the center frequency. The phase shift is -90 degrees for partials at the center frequency. It rises toward 0 degrees for partials at frequencies below the center, and falls toward -180 degrees for partials at frequencies above the center. With low-frequency waveforms, you'll be able to hear this phase shift. As a rule, however, the ear is not sensitive to phase shifts unless they're changing, so you'll usually want to use Source 1 or 2, and assign an LFO to sweep the center frequency up and down.

Periodic phase shifts like those induced by an LFO on the center frequency will cause a vibrato-like variation in the pitch of a sine wave input. This vibrato effect will be less regular for more complex partials. The greater the depth setting of the control source using the LFO, the greater the vibrato effect.

The amount of vibrato effect also depends on the speed and amount of the phase shift. Try adjusting the rate of the LFO controlling the center frequency. Another way to increase the amount of phase shift is to use the two-pole allpass filter, or to use the one-pole allpass filter in more than one algorithm block.

## Two-pole Allpass Filter (2POLE ALLPASS)



Using 2POLE ALLPASS is very similar to using ALPASS, with two differences. First, the phase shift is -180 degrees for partials at the center frequency, approaching 0 degrees for partials at low frequencies, and approaching -360 degrees for partials at high frequencies.

Second, since this is a two-stage function, there's an additional control input page (F2 WID) which controls the filter width. The parameters on this page affect the frequency range, measured in octaves, where most of the phase shifting occurs. Small values cause a drop from 0 to -360 in the phase shift to occur near the center frequency, while large values spread the drop in the phase shift over a broader frequency range. Small values tend to affect just a few partials, leaving others mostly untouched. The affected partials seem to become detached from the others, creating the illusion of an additional sound source.

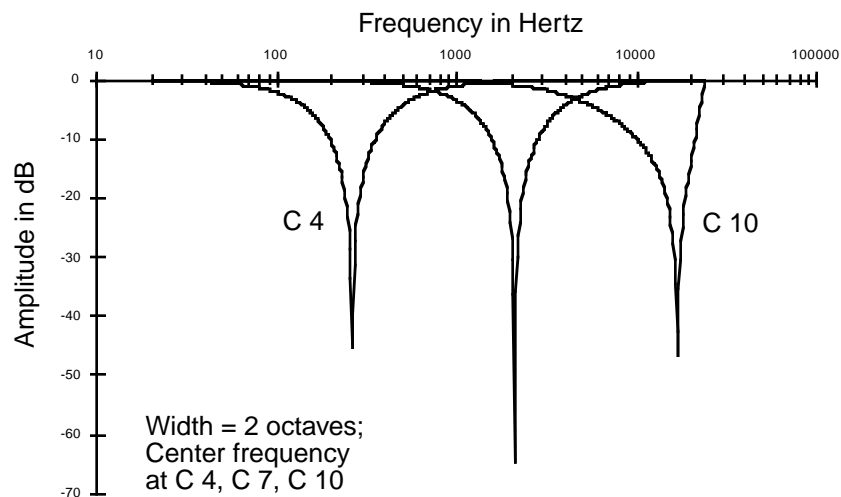
If you leave the center frequency constant and assign an LFO to vary the width, partials with frequencies above the center will shift their pitches in the opposite direction of partials below the center frequency.

```

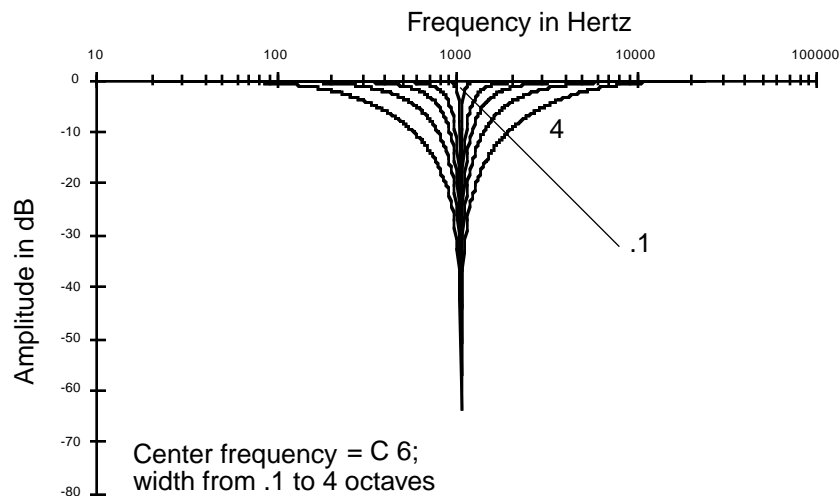
EditProg:F2 WID(2P ALPHSS) <>Layer:1/1
Adjust:0.010oct Src1 :OFF
Depth :0.00oct Src2 :OFF
KeyTrk:0.000oct/key DptCt1:OFF
VelTrk:0.00oct MinDpt:0.00oct
MaxDpt:0.00oct
<more F1 FRQ F2 WID F3 AMP F4 AMP more>
  
```

PARAMETER	RANGE OF VALUES
ADJUST	0.010 to 5.000 octaves
KEY TRACKING	± .200 octaves per key
VELOCITY TRACKING	± 5.00 octaves
SOURCE 1	Control Source list
SOURCE 1 DEPTH	± 5.00 octaves
SOURCE 2	Control Source list
SOURCE 2 DEPTH CONTROL	Control Source list
MINIMUM DEPTH, SOURCE 2	± 5.00 octaves
MAXIMUM DEPTH, SOURCE 2	± 5.00 octaves

### Two-pole Notch Filter (NOTCH FILTER)

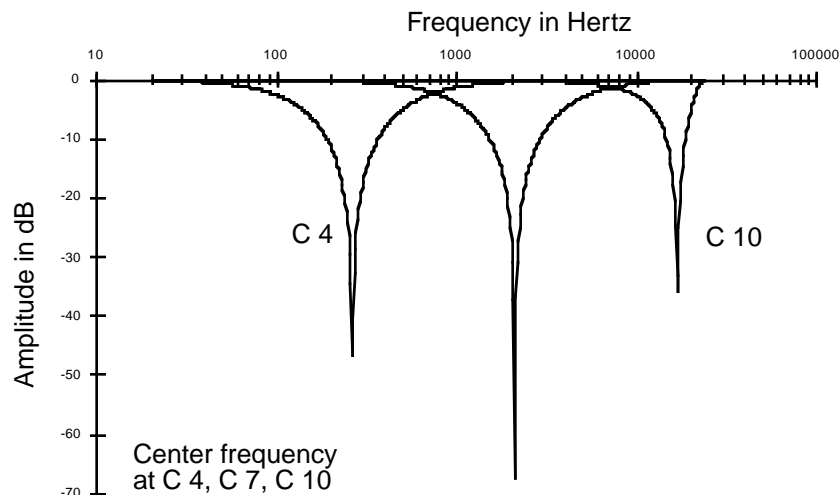


**2-Pole Notch Filter:  
Width in octaves**



The two-pole notch filter has two control input pages, one for center frequency, one for width. Partial with frequencies above or below the notch will be unaffected. Within the notch, partials will be attenuated according to the width of the notch. The width is defined in terms of the number of octaves between the points on the signal's attenuation curve where the attenuation is 3 dB (see the explanation of F2 WID for the PARAMETRIC EQ function). For example, if the width is set at four octaves, then the attenuation will be 3 dB at two octaves in either direction from the center frequency. There's no attenuation of partials at more than two octaves in either direction from the center frequency.

**Two-pole Notch Filter, Fixed Width (NOTCH2)**



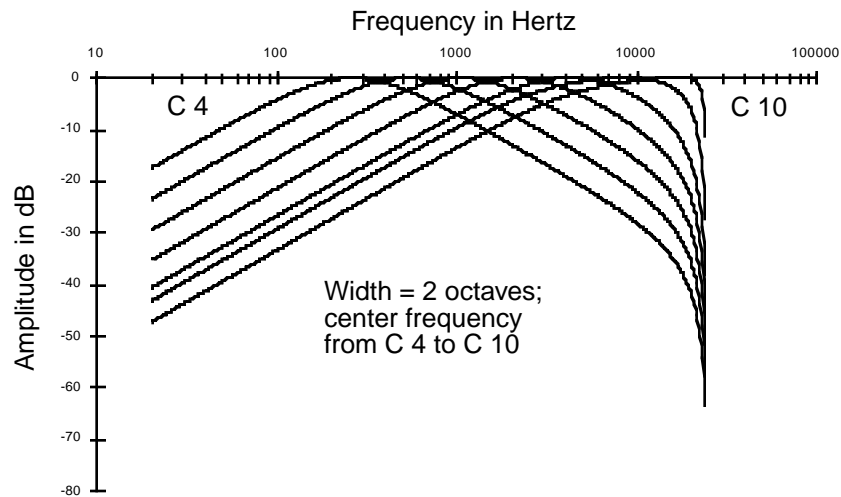
The only functional difference between NOTCH2 and NOTCH FILTER is that the width of NOTCH2 is fixed at 2.2 octaves. This gives you a one-stage notch filter function.

**Two-pole Bandpass Filter (BANDPASS FILTER)**

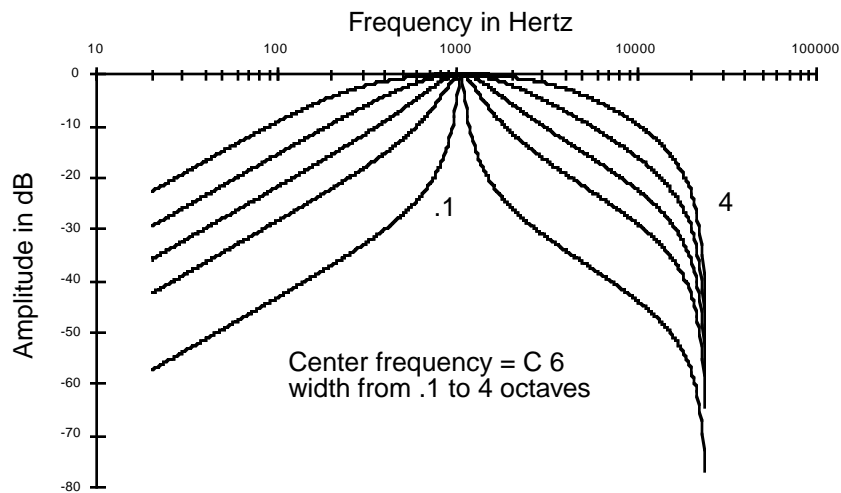
This is essentially the opposite of a notch filter; it passes all partials at the center frequency, and cut the levels of partials above or below the center frequency. The width is defined the same as for the double notch filter.

The gain at the center frequency is 0 dB. Small values for width (a narrow bandpass) may produce a very quiet signal unless the center frequency matches the frequency of a strong sine wave partial. Wide bandpasses may result in a quiet signal if they're centered in a region of the sound where the partials are weak. You can easily boost these quiet signals with the parameters on the F4 AMP page.

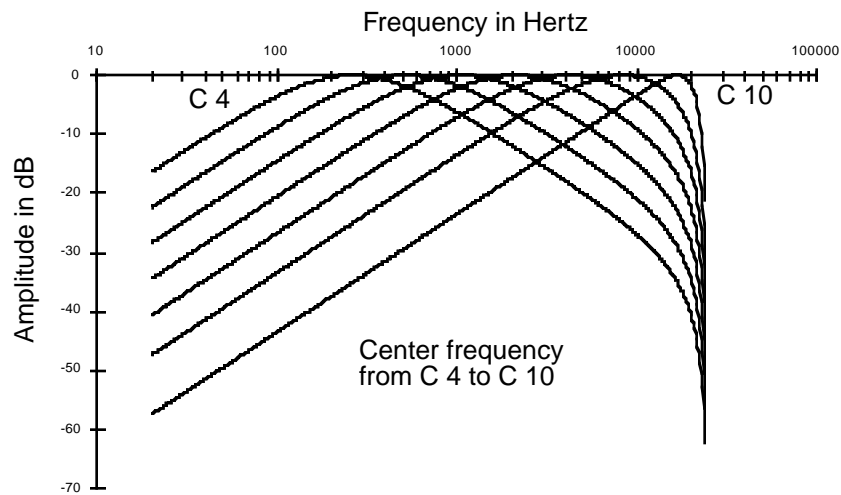
**Bandpass Filter:  
Frequency**



**Bandpass Filter:  
Width in octaves**

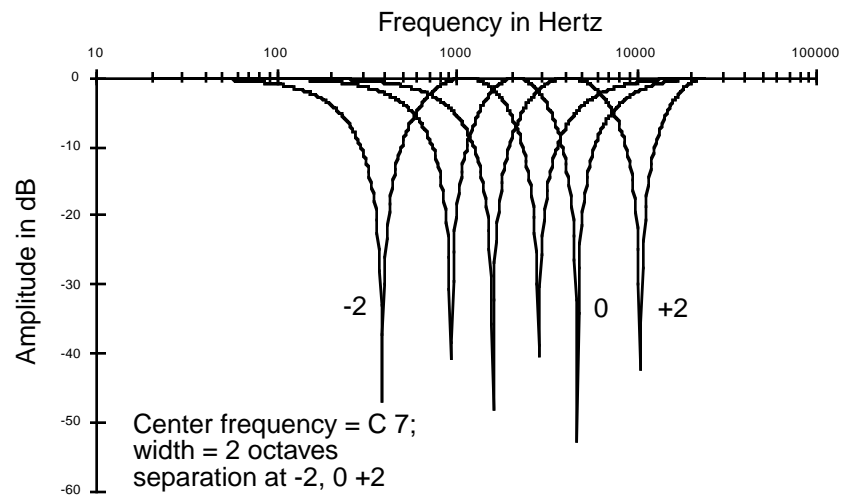


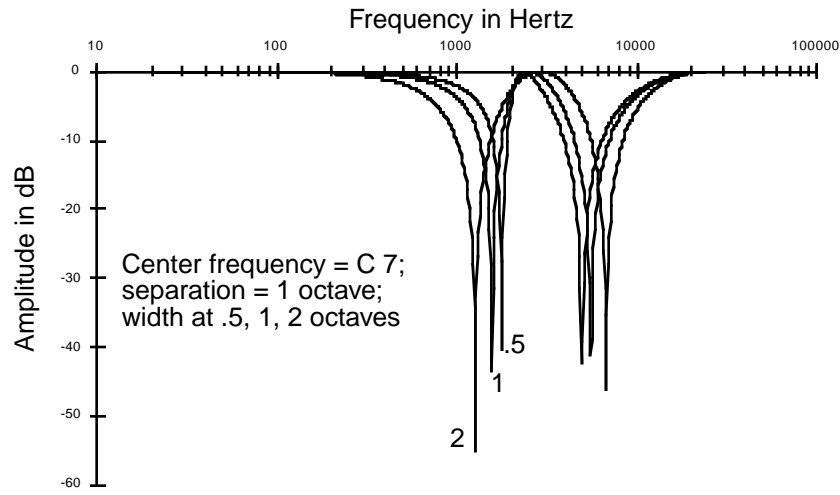
### Two-pole Bandpass Filter, Fixed Width (BAND2)



The only functional difference between BAND2 and BANDPASS FILTER is that the width of BAND2 is fixed at 2.2 octaves. This gives you a one-stage bandpass filter function.

### Double Notch Filter with Separation (DOUBLE NOTCH W/ SEP)





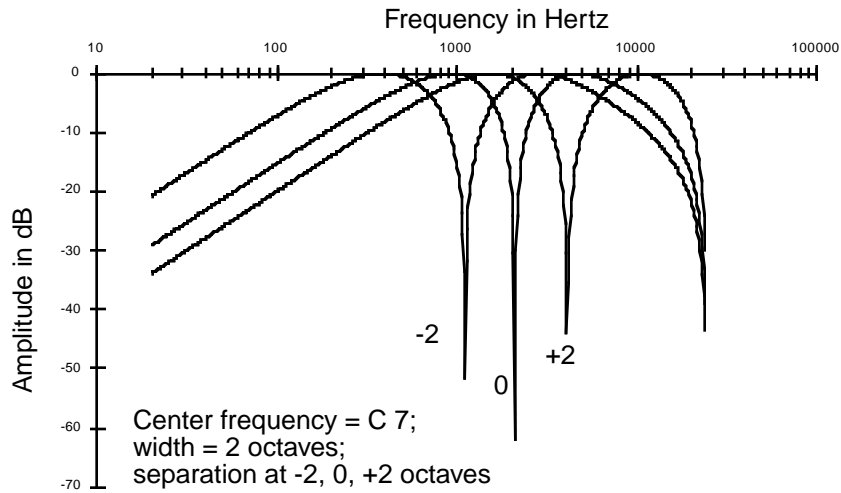
This is a three-stage function that puts two notches in the frequency response. As with NOTCH FILTER and NOTCH2, there are control input pages for frequency and width. A third control input page affects the separation of the notches.

Setting the center frequency on the F1 FRQ page defines the frequency that's halfway between the notches. The settings for the separation affect the behavior of the width control parameters. When the separation is 0, the notches are close to the center frequency, and the width control parameters control the widths of both notches equally. Positive values for separation move the notches apart, and cause the width control parameters to affect the width of the higher-frequency notch more than the width of the lower-frequency notch. Negative values for separation will move the notches apart to the same extent, but will cause the width control parameters to affect the width of the lower-frequency notch more than the width of the higher-frequency notch.

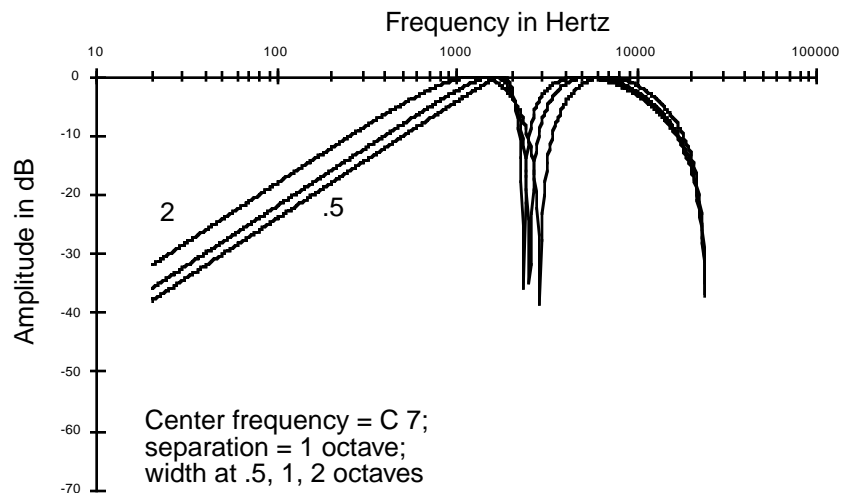
### **Twin Peaks Bandpass Filter (TWIN PEAKS BANDPASS)**

The control parameters for TWIN PEAKS BANDPASS work the same way as for DOUBLE NOTCH FILTER, but of course, you get peaks instead of notches—that is, the amplitudes of partials near the center frequency are high, and the amplitudes are increasingly attenuated at frequencies farther from the center.

**Twin Peaks  
Bandpass Filter:  
Separation**



**Twin Peaks  
Bandpass Filter:  
Width**



## Equalization (EQ)

Equalization is a specialized filtering process that lets you boost or cut the amplitude of a specified range of frequencies.

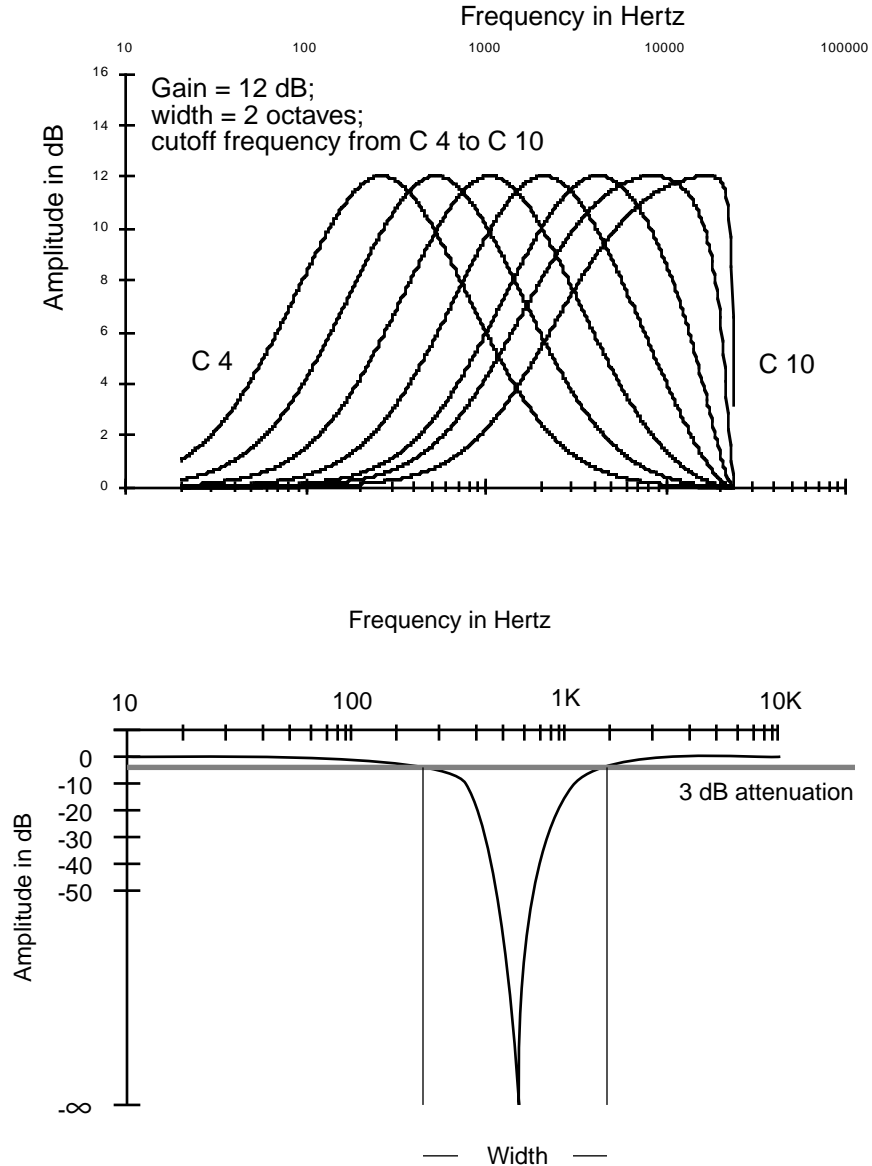
PARAMETRIC EQ  
MID-RANGE PARAMETRIC EQ  
BASS TONE CONTROL  
TREBLE TONE CONTROL  
STEEP BASS TONE CONTROL

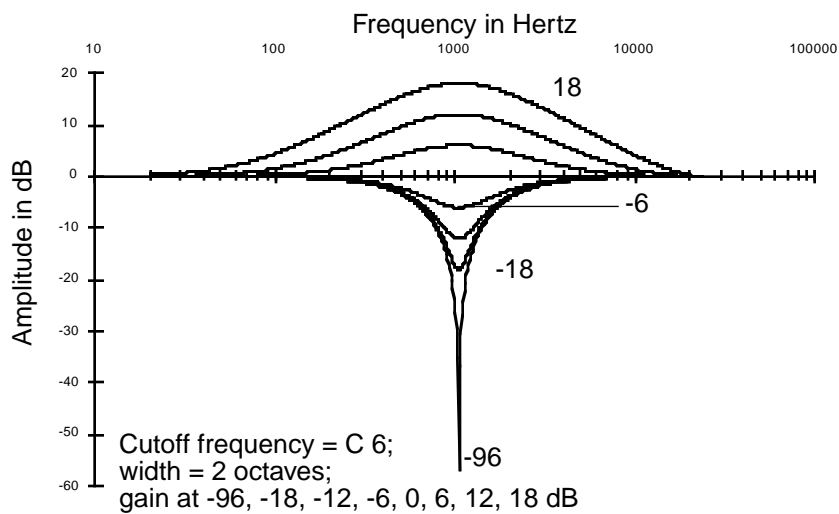
### Parametric Equalizer (PARAMETRIC EQ)

This function has three interacting parameters, each with its own control input page: center frequency, width, and amplitude. The center frequency is the center of the range of frequencies that will be boosted or cut by the amplitude setting. The width is the entire range of frequencies

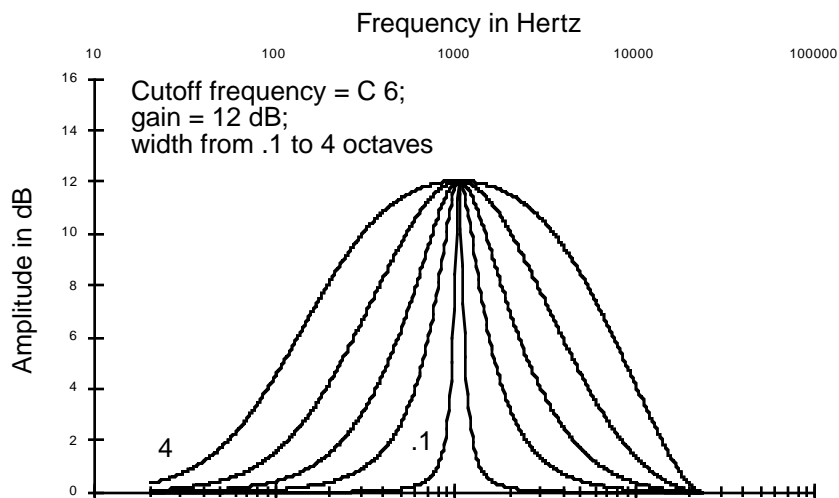


that will be affected by the amplitude setting. For the K2500, the width is defined by imagining an amplitude curve with a level (in dB) of -infinity (minus infinity) at the center frequency, then measuring the distance (in octaves) between the points on the curve where the amplitude is attenuated by 3dB. See the diagram below.





When you're using the Parametric EQ, you might use the following sequence. Set the center frequency (press the F1 FRQ soft button to select its control input page). The frequency is measured in terms of each note of the keyboard. The frequency in Hertz of each note appears with the note name as the value for the Adjust parameter. Next, select the width control input page (the F2 WID soft button) to determine how wide a range will be affected by the amplitude adjustment. Then select the amplitude control input page (F3 AMP soft button), and adjust the amplitude of the range you specified with the center frequency and width settings. You'll probably jump back and forth between these three pages until your ear is satisfied with the sound.



```

EditProg:F1 FRQ(PARH EQ) <>Layer:1/1
Adjust:C 4 262Hz Src1 :OFF
Fine :0ct Depth :0ct
KeyTrk:0ct/key Src2 :OFF
VelTrk:0ct DptCt1:OFF
Ped :0dB MinDpt:0ct
MaxDpt:0ct
<more F1 FRQ F2 WID F3 AMP F4 AMP more>
  
```

PARAMETER	RANGE OF VALUES
COARSE ADJUST	C 0 16 Hz to G 10 25088 Hz
FINE ADJUST	± 100 cents
KEY TRACKING	± 250 cents per key
VELOCITY TRACKING	± 10800 cents (9 octaves)
PAD	0, 6, 12, 18 dB
SOURCE 1	Control Source list
SOURCE 1 DEPTH	± 96 dB
SOURCE 2	Control Source list
SOURCE 2 DEPTH CONTROL	Control Source list
MINIMUM DEPTH, SOURCE 2	± 10800 cents
MAXIMUM DEPTH, SOURCE 2	± 10800 cents

The Fine Adjust parameter gives you one-cent precision in setting the center frequency.

```

EditProg:F2 WII(PARA EQ) <>Layer:1/1
Adjust:0.010oct Src1 :OFF
Depth :0.000oct
Src2 :OFF
KeyTrk:0.000oct/key DptCt1:OFF
VelTrk:0.000oct MinDpt:0.000oct
MaxDpt:0.000oct
<more F1 FRQ F2 WII F3 AMP F4 AMP more>

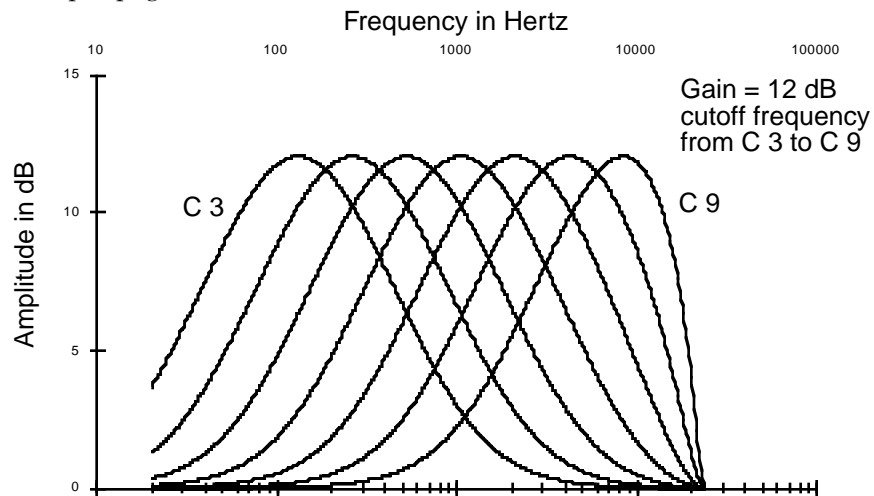
```

PARAMETER	RANGE OF VALUES
ADJUST	0.010 to 5.000 octaves
KEY TRACKING	± .200 octaves per key
VELOCITY TRACKING	± 5.00 octaves
SOURCE 1	Control Source list
SOURCE 1 DEPTH	± 5.00 octaves
SOURCE 2	Control Source list
SOURCE 2 DEPTH CONTROL	Control Source list
MINIMUM DEPTH, SOURCE 2	± 5.00 octaves
MAXIMUM DEPTH, SOURCE 2	± 5.00 octaves

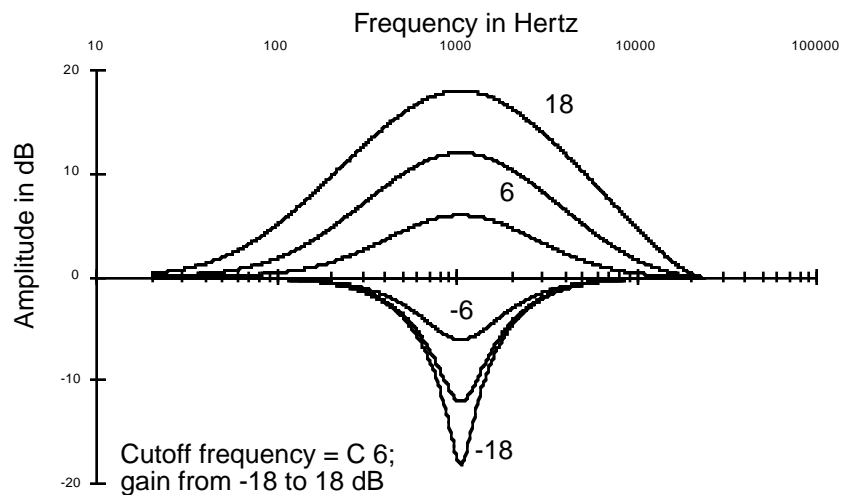
The control input page for the amplitude stage (F3 AMP) is identical to the AMP page described previously, except that there's no Pad parameter.

### Mid-range Parametric EQ (PARA MID)

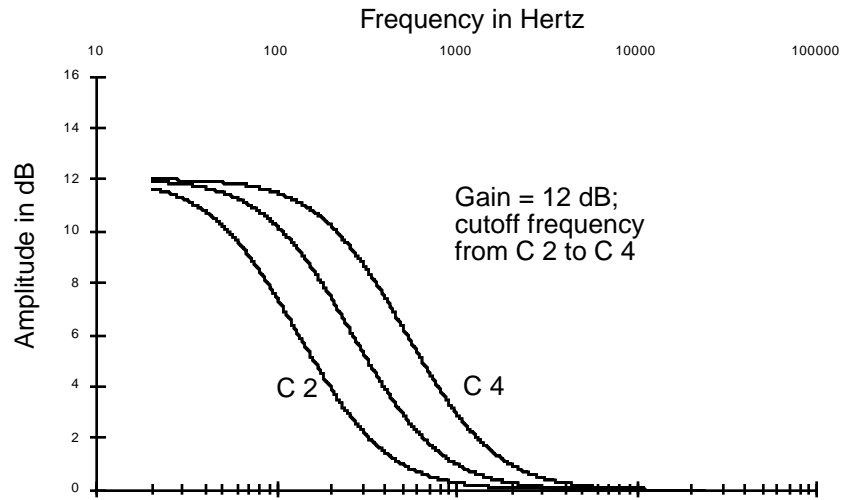
This two-stage function is almost identical to the three-stage Parametric EQ function. The only difference is that the width of PARA MID is fixed at 2.2 octaves. Consequently there's no control input page for the width.



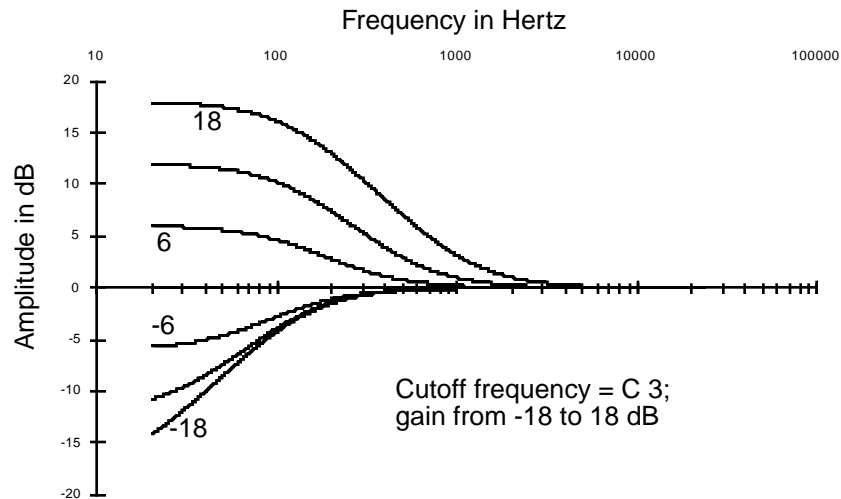
Para Mid: Gain



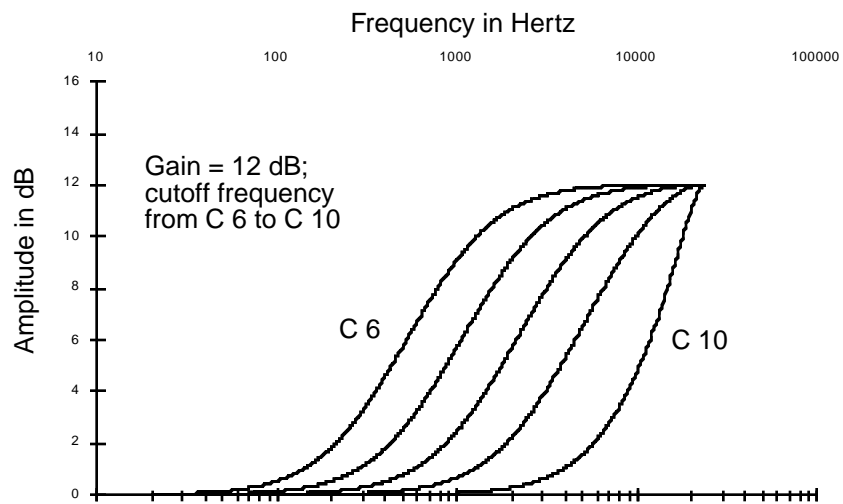
## Bass Tone Control (PARA BASS)



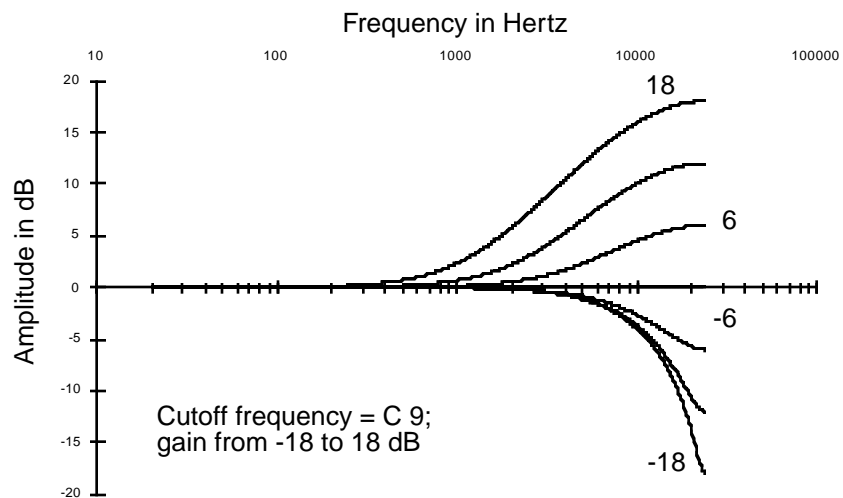
This is a two-stage function, with control input pages for frequency and amplitude. These pages are the same as those for frequency and amplitude in PARA EQ. On the frequency control input page, you'll set the cutoff frequency. For notes above this frequency, the amplitude setting has a diminished effect. On the amplitude control input page, you'll set the amount of cut or boost that's applied to notes below the cutoff frequency. There's a gradual increase in the bass response for each successively lower note. The location of the cutoff frequency will change somewhat as you change the amplitude settings, although the value for the Adjust parameter on the frequency control input page will not reflect the change.



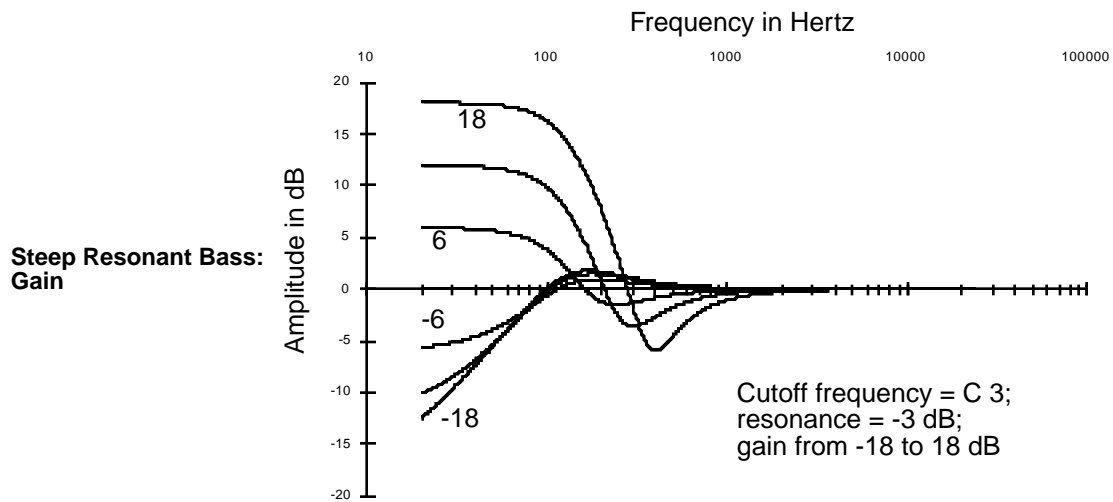
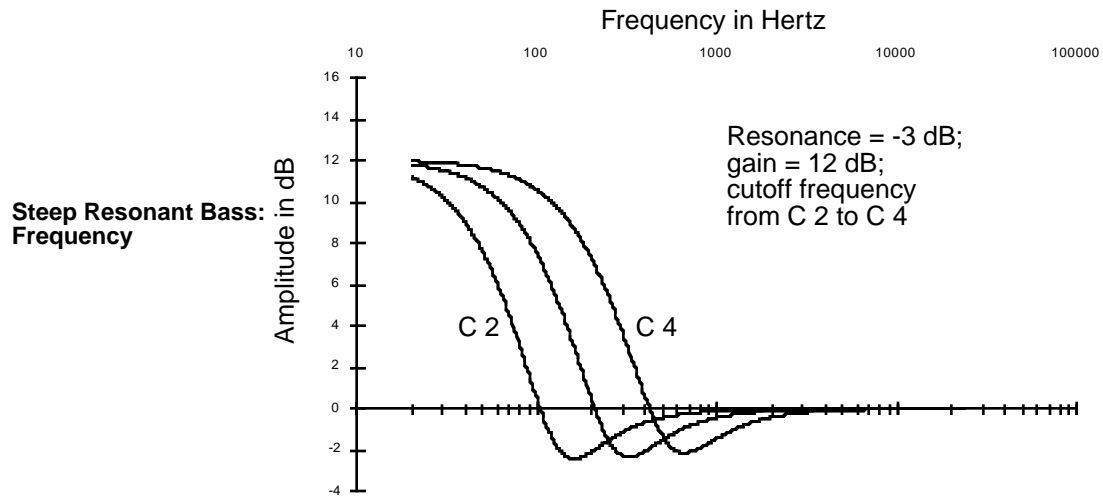
### Treble Tone Control (PARA TREBLE)

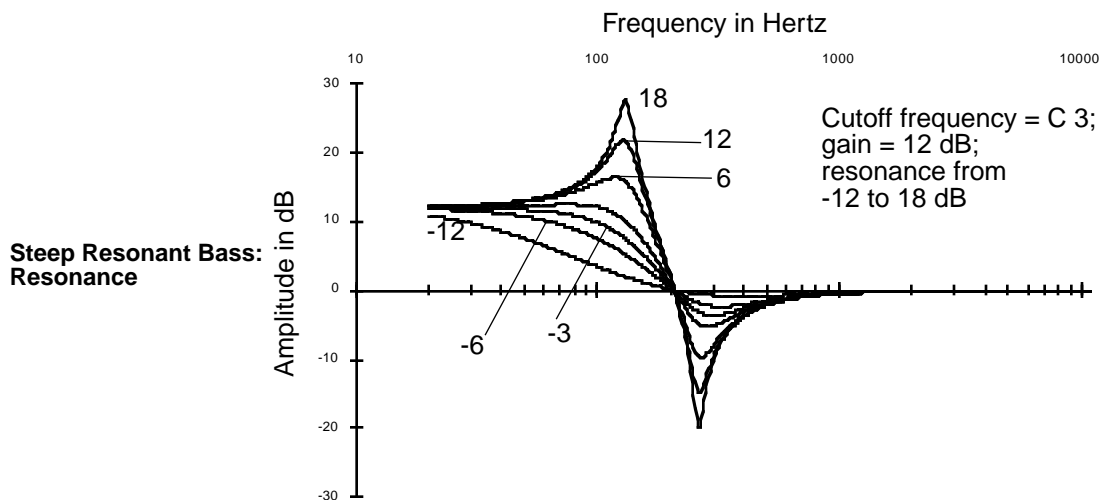


PARA TREBLE is very similar to PARA BASS; the only difference is that the amplitude setting affects notes *above* the cutoff frequency.



### Steep Bass Tone Control (STEEP RESONANT BASS)





This function uses a two-pole lowpass filter to give you a sharper transition in bass response than PARA BASS. Like PARA BASS, there are control inputs pages for cutoff frequency and amplitude, which are identical to those for PARA BASS. There is also a control input page for resonance (also known as “q”), which can boost or cut the amplitude of the partials near the cutoff frequency.

```

EditProg: F2 RES(StEEP BASS) Layer: 1/1
Adjust: 0dB Src1 : OFF
Depth : 0.0dB
Src2 : OFF
KeyTrk: 0.00dB/key DptCt1: OFF
VelTrk: 0.0dB MinDpt: 0.0dB
MaxDpt: 0.0dB
<more F1 FRQ F2 RES F3 AMP F4 AMP more>
  
```

PARAMETER	RANGE OF VALUES
ADJUST	-12 to 24 dB
KEY TRACKING	± 2.00 dB per key
VELOCITY TRACKING	± 30 dB
SOURCE 1	Control Source list
SOURCE 1 DEPTH	± 96 dB
SOURCE 2	Control Source list
SOURCE 2 DEPTH CONTROL	Control Source list
MINIMUM DEPTH, SOURCE 2	± 30 dB
MAXIMUM DEPTH, SOURCE 2	± 30 dB

You’ll get the best transition in bass response with a resonance setting of -6 dB. There’s a small frequency range above the cutoff frequency where the response reverses direction (if you’re cutting the amplitude, for example, you’ll get a slight boost just above the cutoff frequency). The higher you set the resonance, the larger this reversal will be, resulting in unusual—but possibly useful—response curves at high resonance values.



## Pitch / Amplitude / Panner

PITCH  
AMP  
PANNER  
UPPER AND LOWER AMP  
BALANCE AND AMP  
GAIN

### PITCH

We used the PITCH control input page as an example to introduce the common DSP control parameters in Chapter 6, so we won't add much here. The PITCH function modifies the pitch of the layer's keymap as it passes through the sound engine. The PITCH stage of each algorithm is always the first stage. Algorithms 26—31, the Sync algorithms, don't show the PITCH stage on the ALG page, since these algorithms generate their own sawtooth waves, and do not use keymaps.

### AMP

The AMP function is the final stage in every single-output algorithm, and controls the overall amplitude (volume) of the layer. This is an easy way to boost the signal to a more desirable level if it's not loud enough for your purposes. Large values for the Adjust parameter can cause a sound to clip, which will distort most sounds considerably. You may want this effect, and using it won't damage anything, but as a rule, you'll want to avoid clipping your sounds with the AMP (or GAIN) function. There are many other ways to distort your sounds, like DIST, SHAPER, and WRAP, to name a few.

The settings for the parameters on the F4 AMP page affect the gain level for the currently selected layer. So do the settings on the AMPENV page. Compare this with the effect of GAIN, described later in this section.

```

EditProg:F4 AMP(FINAL AMP) <>Layer:1/1
Adjust:0dB Src1 :OFF
Depth :0dB
Src2 :OFF
KeyTrk: 0.00dB/key DptCtl:OFF
VelTrk:0dB MinDpt:0dB
Pad :0dB MaxDpt:0dB
<more F1 F2 F3 F4 AMP more>
  
```

PARAMETER	RANGE OF VALUES
ADJUST	-96 to 48 dB
KEY TRACKING	± 2.00 dB per key
VELOCITY TRACKING	± 96 dB
PAD	0, 6, 12, 18 dB
SOURCE 1	Control Source list
SOURCE 1 DEPTH	± 96 dB
SOURCE 2	Control Source list
SOURCE 2 DEPTH CONTROL	Control Source list
MINIMUM DEPTH, SOURCE 2	± 96 dB
MAXIMUM DEPTH, SOURCE 2	± 96 dB

## PANNER

This single-stage function converts a single wire at its input into a double wire at its output, splitting the signal between an “upper” and “lower” wire. This creates the double-output algorithm we discussed in Chapter 6. The parameters on the PANNER page enable you to modify the signal’s routing through the upper and lower wires. By itself the PANNER doesn’t change the pan position of the sound. It just defines what percentage of the currently selected layer’s sound goes to each wire. When you select one of these double-output algorithms, the OUTPUT page for the layer changes to enable you to make pan settings for each wire independently. So when you use the PANNER function, you’ll also want to adjust the Pan parameters on the OUTPUT page, setting the upper wire’s pan fully right, and the lower wire’s pan fully left. This will enable you to hear the effect of the PANNER function.

The PANNER function is available only in algorithms 2, 13, 24, and 26, and always appears in the block before the final AMP function. Consequently, it will always be selected with the F3 soft button, which is labeled “F3 POS” (position).

```

EditProg: F3 POS(PANNER) <> Layer: 1/1
Adjust: 0% Src1 : OFF
Depth : 0%
Src2 : OFF
KeyTrk: 0.0%key DptCt1: OFF
VelTrk: 0% MinDpt: 0%
Pad : 0dB MaxDpt: 0%
<more F1 F2 F3 POS F4 AMP more>
  
```

PARAMETER	RANGE OF VALUES
ADJUST	± 100%
KEY TRACKING	± 16%/key
VELOCITY TRACKING	± 200%
PAD	0, 6, 12, 18 dB
SOURCE 1	Control Source list
SOURCE 1 DEPTH	± 200%
SOURCE 2	Control Source list
SOURCE 2 DEPTH CONTROL	Control Source list
MINIMUM DEPTH, SOURCE 2	± 200%
MAXIMUM DEPTH, SOURCE 2	± 200%

The Adjust parameter sets the initial routing of the layer to the upper or lower wire. -100% is the lower, and 100% is the upper. The KeyTrk parameter lets you shift the layer’s sound from one wire to the other based on the MIDI key number of each note. For positive values of KeyTrk, the higher above Middle C, the more sound goes to the upper wire.

The remaining parameters have ranges from -200% to 200%. This lets you start with a sound that’s fully on the lower wire, for example, and shift it completely to the upper wire. The VelTrk parameter shifts notes between wires based on the attack velocity of each note. For positive values, the higher the attack velocity, the more sound goes to the upper wire. The Src1 and Src2 parameters let you assign controls to reroute the sound relative to the initial routing. Setting their depth parameters to positive values will shift the sound to the upper wire when the controls assigned to them approach their maximum values.

## Upper and Lower Amp (AMP U AMP L)

This two-stage function is similar to the AMP function described above, but it appears in algorithms that have split the signal to two wires and has sent them through different DSP functions in the F2 and F3 blocks. This function enables you to set the final amplitude

independently for each wire, and keeps the two signals separate at its output, giving you added flexibility for mixing and panning. Like the AMP function, UPPER AND LOWER AMP always appears as the last block in an algorithm. Since it's a two-stage function, it has two control input pages. F3 selects the control input page for the lower wire, and F4 the control input page for the upper wire.

```

EditProg:F3 AMP(AMP U/L) <>Layer:1/1
Adjust:0dB Src1 :OFF
Depth :0dB
Src2 :OFF
KeyTrk: 0.00dB/key DptCt1:OFF
VelTrk:0dB MinDpt:0dB
Pad :0dB MaxDpt:0dB
<more F1 F2 F3 AMP F4 AMP more>

```

PARAMETER	RANGE OF VALUES
ADJUST	-96 to 48 dB
KEY TRACKING	± 2.00 dB per key
VELOCITY TRACKING	± 96 dB
PAD	0, 6, 12, 18 dB
SOURCE 1	Control Source list
SOURCE 1 DEPTH	± 96 dB
SOURCE 2	Control Source list
SOURCE 2 DEPTH CONTROL	Control Source list
MINIMUM DEPTH, SOURCE 2	± 96 dB
MAXIMUM DEPTH, SOURCE 2	± 96 dB

### Balance and Amp (BAL AMP)

This function has a two-wire input and a two-wire output. The parameters on its control input page affect the amount of gain applied to each wire between input and output. A value of 0% applies equal gain to both the upper and lower wires; at a value of 100% only the upper wire's sound will be audible, at -100%, only the lower wire's sound will be audible. This works like the balance control on any stereo system; as the gain increases for one wire, it decreases for the other. It's also similar to the PANNER and XFADE functions. The F3 soft button selects the control input page for the balance stage of this function.

```

EditProg:F3 POS(BAL/AMP) <>Layer:1/1
Adjust:0dB Src1 :OFF
Depth :0%
Src2 :OFF
KeyTrk: 0.0%/key DptCt1:OFF
VelTrk:0% MinDpt:0%
Pad :0% MaxDpt:0%
<more F1 F2 F3 POS F4 AMP more>

```

PARAMETER	RANGE OF VALUES
ADJUST	± 100%
KEY TRACKING	± 16.00% per key
VELOCITY TRACKING	± 200%
PAD	0, 6, 12, 18 dB
SOURCE 1	Control Source list
SOURCE 1 DEPTH	± 200%

<b>SOURCE 2</b>	<b>Control Source list</b>
<b>SOURCE 2 DEPTH CONTROL</b>	<b>Control Source list</b>
<b>MINIMUM DEPTH, SOURCE 2</b>	<b>± 200%</b>
<b>MAXIMUM DEPTH, SOURCE 2</b>	<b>± 200%</b>

The AMP stage sets the overall amplitude applied to both wires, and is programmed exactly like the AMP function described above. Their control input pages are almost identical, including the ranges of values. The only difference is that there's no Pad parameter for the AMP stage of the BAL AMP function. The F4 soft button selects the control input page for the AMP stage.

### GAIN

This function, like AMP, can boost or cut the amplitude of the signal as it passes through the algorithm. Unlike AMP, however, the layer's amplitude envelope doesn't effect the gain settings. GAIN can be used to introduce clipping into a signal, or for adjusting the amplitude of an added waveform. Use a gain function when you want to boost or cut all of a sound's partials uniformly. The control input page for GAIN is identical to the one for AMP.

## Mixers

+AMP  
+GAIN  
CROSSFADE

The functions in the Mixer category combine two-wire signals in various ways. They have double wires at their inputs, and they mix and amplify the signals from the two wires, then combine them for output to a single wire. Depending on where you assign one of these functions, they can be used to combine two-wire signals for the F4 AMP block, or to enable you to apply another DSP function to the combined signals before the F4 AMP block.

There's a Pad parameter on the control input pages for these functions, which attenuates the lower wire's signal at its input to the function.

### +AMP

The two input signals to this function are multiplied by .5 (to reduce the likelihood of clipping), then added together. The resulting signal is then multiplied by a gain factor (the combined values for the parameters on the control input page), and multiplied by 2. Any clipping that occurs can be eliminated by lowering the value of the Adjust parameter. If the Adjust value is -6 dB or lower, the signal will never clip. The control parameters are affected by the settings and controls on the AMPENV page.

### +GAIN

This function operates in almost the same way as +AMP, the only difference being that the signal is not affected by the settings on the AMPENV page, since it occurs before the final AMP block.

### Crossfade (XFADE)

This function adds the signals from the upper and lower wires after evaluating the combined values of the parameters on its control input page. If those values add up to -100%, only the lower wire's signal is sent to the output. If they add up to 100%, only the upper wire's signal is sent to the output. If they add up to 0%, both signals are attenuated 6 dB, then added and sent to the output.

## Waveforms

SINE  
LOW FREQUENCY SINE  
SAWTOOTH  
LOW FREQUENCY SAWTOOTH  
SQUARE  
LOW FREQUENCY SQUARE

In this category of DSP functions are three standard synth waveforms—Sine, Sawtooth, and Square—with high- and low-frequency variations of each. These are all one-stage functions. They can be assigned in several different positions and combinations in many of the algorithms.

One important fact to keep in mind is that assigning one of these waveforms to a layer's algorithm may remove the original sample from the signal, since they don't have input signals to send to their outputs (they send only the waveform that they generate themselves). If, for example, you were editing the Acoustic Piano program, and you assigned SINE in the F1 block, you would no longer hear the piano timbre, only the sine wave (unless the signal splits before the F1 block, as in Algorithm 10). Consequently, you'll tend to use these waveforms when you want to build a sound from scratch. If you want to add a waveform to the original timbre of a sound, use one of the added waveform functions described in the next section, or use one of the split signal algorithms.

These waveforms can range in frequency from .1 Hz to 20 KHz. They're not samples like the instrumental sounds and other waveforms; they're generated by oscillators. Since the DSP function waveforms aren't produced by playing back multi-sample keymaps, there are no sample root transitions as you play notes in different keyboard ranges. This makes them especially suitable for use with portamento and wide pitch bend ranges.

Since these waveform functions generate an output signal only, and don't receive an input signal to pass along, the algorithms are arranged so you won't inadvertently assign a series of waveforms that interfere with each other. You'll usually find, for example, that if you can assign a waveform in the F1 block, all subsequent blocks will allow you to assign only the added waveforms. Or, if the subsequent blocks allow you to assign the "regular" waveforms, it's because the wiring of the algorithm is split so that the two waveforms pass through in parallel (as in Algorithm 10).

This next point is a small one, but important, and may make it easier for you to understand the way the waveform functions operate, especially if you've been carefully studying the wiring paths of the algorithms. In several algorithms where the waveforms are available the wiring paths of the algorithms (the horizontal arrows) appear to send a signal to an input of the waveforms. This is *not* the case, and anywhere one of these waveforms is assigned, you should view the algorithm as if there were no horizontal arrow pointing to the left (input) side of the block where the waveform is assigned. The diagrams below will clarify this point. The only difference in the DSP function assignments is in the F1 block, where the first and second diagrams show the SAW waveform, and the third diagram shows the SAW+ added waveform (described in the next section). In the first diagram, the PITCH function's output (passing the sample signal from the keymap) appears to be connected to the input of the F1 block (the SAW function), as well as splitting and passing to the +GAIN function in the F2 block. This is what you would see on the ALG page.

In fact, the actual signal path does *not* pass from the PITCH function through the SAW function; it splits and bypasses the SAW function, as shown in the second diagram. The third diagram shows the same algorithm with the SAW+ added waveform assigned to the F1 block. In this case, the diagram is accurate; the signal passes from the output of the PITCH function, and splits into a two-wire signal. The upper wire passes through the F1 block where the sawtooth wave is added, and into the +GAIN function in the F2 block. The lower wire bypasses the F1 block, and passes directly to the F2 block, where it is combined with the upper wire signal.

Algorithm 24



Algorithm 24



Algorithm 24



The six waveforms in this category are Sine, Sawtooth, Square, Low Frequency Sine, Low Frequency Sawtooth, and Low Frequency Square. The control input pages for all six waveforms affect the pitch of the waveforms. The control input pages for the first three waveforms are identical, as are the control input pages for the three low frequency waveforms are identical.

### SINE, Sawtooth (SAW), SQUARE

There's only one parameter on this control input page that may still be unfamiliar to you: Fine Hz. We discussed this in Chapter 6. It can tune the pitch of the waveform in terms of its actual frequency in Hertz, as opposed to the usual method of tuning by key names. The advantage to using the Fine Hz parameter is that you can maintain constant beat frequencies across much of the keyboard when you have a program with slightly detuned multiple layers (or multiple waveforms in one layer).

```

EditProg:F1 PCH(SINE) <>Layer:1/1
Coarse:0ST Src1 :OFF
Fine :0ct Depth :0ct
FineHz: 0.00Hz Src2 :OFF
KeyTrk:0ct/key DptCt1:OFF
VelTrk:0ct MinDpt:0ct
Pad :0dB MaxDpt:0ct
<more F1 PCH F2 F3 F4 AMP more>
  
```

#### PARAMETER

#### RANGE OF VALUES

COARSE ADJUST	−120 to 60 semitones
FINE ADJUST	± 100 cents
FINE ADJUST IN HERTZ	± 6.00 Hertz
KEY TRACKING	± 2400 cents per key (2 octaves)
VELOCITY TRACKING	± 7200 cents (6 octaves)
PAD	0, 6, 12, 18 dB
SOURCE 1	Control Source list
SOURCE 1 DEPTH	± 7200 cents
SOURCE 2	Control Source list
SOURCE 2 DEPTH CONTROL	Control Source list
MINIMUM DEPTH, SOURCE 2	± 7200 cents
MAXIMUM DEPTH, SOURCE 2	± 7200 cents

If you want the waveform to play with standard twelve-tone octaves, set the KeyTrk parameter to 100 cents per key. Different values for KeyTrk will result in non-standard tunings.

### Low Frequency Waveforms: Sine (LF SIN), Sawtooth (LF SAW), Square (LF SQR)

These can be used like the waveforms above, since their frequency ranges are similar, but they're intended to be used not for their timbres, but for the shape of their waveforms. By using low frequency values for these waveforms, you're basically getting extra LFOs with very precise control parameters. They're intended to be used as inputs to drive the DSP functions in the subsequent algorithm blocks. They're especially useful with the non-linear DSP functions, such as xAMP.

The parameters on this page affect the pitch of the low frequency waveform in a slightly different manner. They're all tied to the value of the Coarse Adjust parameter, so when you're working with this page, you'll want to set the Coarse Adjust first, then set the values of the other parameters to modify the initial setting. The Coarse Adjust value is multiplied by the values of the other parameters to determine the effect on the pitch, as indicated by the "x" after the parameters' values. More parameter descriptions follow below.

```

EditProg: F1 PCH(LF SIN) <> Layer: 1/1
Coarse: 100.0Hz Src1 : OFF
Fine : 4.00x Depth : 1.000x
Src2 : OFF
KeyTrk: 2.00x/oct DptCt1: OFF
VelTrk: 1.000x MinDpt: 1.000x
Pad : 0dB MaxDpt: 1.000x
<more> F1 PCH F2 F3 F4 AMF <more>

```

PARAMETER	RANGE OF VALUES
COARSE ADJUST	0.1, 1.0, 10.0, 100.0, 1000.0 Hertz
FINE ADJUST	1.00 to 20.00 x
KEY TRACKING	0.1 to 10.0 x per octave
VELOCITY TRACKING	0.010 to 32.000 x
PAD	0, 6, 12, 18 dB
SOURCE 1	Control Source list
SOURCE 1 DEPTH	0.010 to 32.000 x
SOURCE 2	Control Source list
SOURCE 2 DEPTH CONTROL	Control Source list
MINIMUM DEPTH, SOURCE 2	0.010 to 32.000 x
MAXIMUM DEPTH, SOURCE 2	0.010 to 32.000 x

### Coarse Adjust (Coarse)

There are only five values to choose from. They're expressed in terms of their frequencies in Hertz. Each value has a frequency ten times higher or lower than adjacent values.

### Fine Adjust (Fine)

Set this parameter's value to 1.00 x if you don't want it to affect the Coarse Adjust setting. Doubling this value (2.00 x, 4.00 x, etc.) will raise the pitch in octaves. By using Fine Adjust in tandem with Coarse Adjust, you can achieve frequencies from .1 Hz (well below the audible range) to 20 KHz.

### Key Tracking (KeyTrk)

A value of 1.00 x per octave will keep the waveform's pitch uniform across the entire keyboard. A value of 2.00 x per octave will give you the normal twelve-tone octave. Other values will give you non-standard tunings.

## Added Waveforms

SINE+  
SAW+  
NOISE+

There are three DSP functions that add waveforms to a layer's existing sample: SINE+, SAW+, and NOISE+.

The parameters on the control input page for the SINE+ function affect the pitch of the sine waveform without affecting the pitch of the existing sample. The control input page for the SINE+ function is similar to those for the regular waveforms above. There are parameters for coarse adjust, key tracking, velocity tracking, Source 1 and Source 2, and a pad. There are also parameters for fine adjust and fine Hertz adjust.

The SAW+ function is virtually identical to the SINE+ function; the only difference is in the shape of the waveform.

The NOISE+ function is tied to the level of the sample to which it's added. It will generate fairly white noise (that is, nearly equal amplitude at all audible frequencies) as long as the amplitude of the sample is non-zero. The amplitude of the noise is multiplied by its gain control (the Adjust parameter on its control input page), then added to the signal. To add a short burst of noise at the beginning of a sound, assign ENV2 as the value of one of the Source parameters, then edit ENV2 to produce an envelope with a rapid decay.

The control parameters for NOISE+ are similar to those for SINE+ and SAW+, except that there are no parameters for fine adjust or fine Hertz adjust.

## Non-linear Functions

HIGH FREQUENCY STIMULATOR  
DISTORTION  
SHAPER  
DOUBLE SHAPER  
TWO-PARAMETER SHAPER  
WRAP  
LOWPASS FILTER WITH CLIPPING  
PULSE WIDTH MODULATION

The functions in this category have a variety of effects on the signal. What they have in common is that they can add partials to the signal that were not present at their inputs.

The non-linear functions can produce dramatic changes in timbre, resulting in all sorts of new and modified sounds. One thing to keep in mind is that sounds with a large number of high-frequency partials can be subject to distortion at the high end of the keyboard, especially when you're using more than one of the non-linear DSP functions. You might also hear a bit of *aliasing* with some sounds. Aliasing refers to unintended partials that occur below the fundamental pitch of a sound. The easiest way to remove this distortion or aliasing is to reduce the level of the Adjust parameter on the control input page for whichever non-linear DSP functions you're working with. When you're using PWM followed by DIST or SHAPER, you'd reduce the level of the Adjust parameter for the DIST or SHAPER function. You can also use key tracking (KeyTrk, usually with a negative value), and key tracking in combination with the Keytrack Start (KStart) parameter that's described at the beginning of this chapter.

Even with the damping effects of KeyTrk and KStart, you may come up with sounds that are fantastic in the low range, but gritty in the high range. You can transpose the keymap down to



counteract this, but that's the nature of the non-linear functions. In extreme cases, you can lower the HiKey of the layer to disable the high end completely.

### High Frequency Stimulator (HIFREQ STIMULATOR)

The overall effect of this three-stage function is to boost the high frequency partials of the signal, and depending on the settings of the control inputs, it can add high-frequency partials to the signal as well. It's useful for building sounds that cut through the mix, and have a bright crisp nature.

There's more to the High Frequency Stimulator than meets the eye. It works like this: the signal is run through a high-pass filter, then through a distortion function, then through a second high-pass filter. Finally, it's mixed with the original signal after passing through the final AMP stage of the algorithm. The three control input pages let you adjust the cutoff frequency of the first high-pass filter (F1 FRQ), the amount (drive) of the distortion function (F2 DRV), and the mix (relative amplitude) of the stimulated signal with the original (F3 AMP).

```

EditProg:F1 FRQ(HIFREQ STIM)<>LAYER:1/1
Coarse:C 4 262Hz Src1 :OFF
Fine :0ct Depth :0ct
KeyTrk:0ct/key Src2 :OFF
VelTrk:0ct DptCt1:OFF
Pad :0dB MinDpt:0ct
MaxDpt:0ct
<more F1 FRQ F2 DRV F3 AMP F4 AMP more>
  
```

PARAMETER	RANGE OF VALUES
COARSE ADJUST	C -1 16 Hz to G 10 25088 Hz
FINE ADJUST	± 100 cents
KEY TRACKING	± 250 cents per key
VELOCITY TRACKING	± 10800 cents
PAD	0, 6, 12, 18 dB
SOURCE 1	Control Source list
SOURCE 1 DEPTH	± 10800 cents
SOURCE 2	Control Source list
SOURCE 2 DEPTH CONTROL	Control Source list
MINIMUM DEPTH, SOURCE 2	± 10800 cents
MAXIMUM DEPTH, SOURCE 2	± 10800 cents

```

EditProg:F2 DRU(HI FRQ SUM)<>LAYER:1/1
Adjust:0dB Src1 :OFF
Depth :0dB
KStart:C -1 unipola Src2 :OFF
KeyTrk: 0.00dB/key DptCt1:OFF
VelTrk:0dB MinDpt:0dB
MaxDpt:0dB
<more F1 FRQ F2 DRU F3 AMP F4 AMP more>

```

PARAMETER	RANGE OF VALUES
ADJUST	-96 to 48 dB
KEYTRACK START	C -1 to C 9 unipolar, C -1 to C 9 bipolar
KEY TRACKING	± 2.00 dB per key
VELOCITY TRACKING	± 96 dB
SOURCE 1	Control Source list
SOURCE 1 DEPTH	± 96 dB
SOURCE 2	Control Source list
SOURCE 2 DEPTH CONTROL	Control Source list
MINIMUM DEPTH, SOURCE 2	± 96 dB
MAXIMUM DEPTH, SOURCE 2	± 96 dB

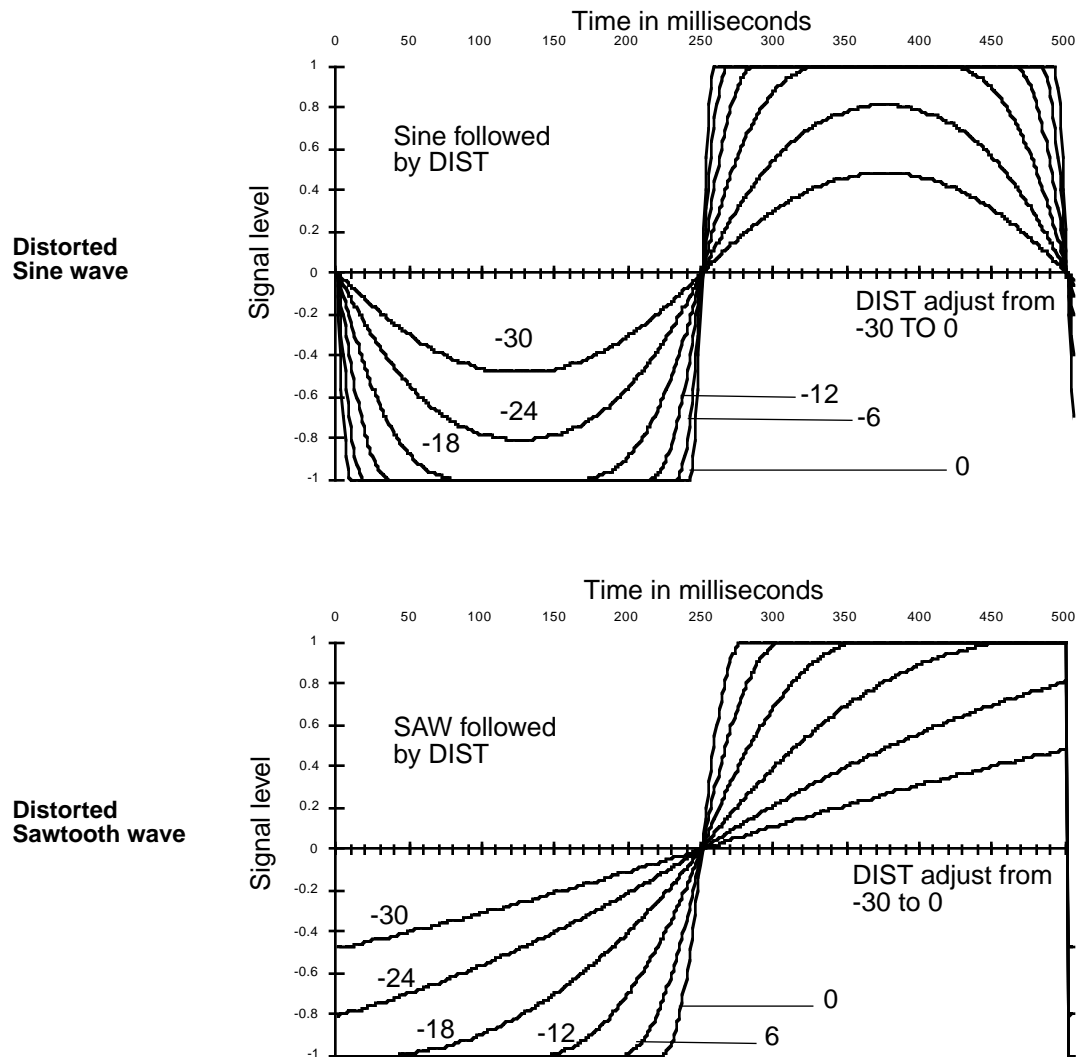
```

EditProg:F3 AMP(HI FRQ SUM)<>LAYER:1/1
Adjust:0dB Src1 :OFF
Depth :0dB
Src2 :OFF
KeyTrk: 0.00dB/key DptCt1:OFF
VelTrk:0dB MinDpt:0dB
MaxDpt:0dB
<more F1 FRQ F2 DRU F3 AMP F4 AMP more>

```

PARAMETER	RANGE OF VALUES
ADJUST	-96 to 48 dB
KEY TRACKING	± 2.00 dB per key
VELOCITY TRACKING	± 96 dB
SOURCE 1	Control Source list
SOURCE 1 DEPTH	± 96 dB
SOURCE 2	Control Source list
SOURCE 2 DEPTH CONTROL	Control Source list
MINIMUM DEPTH, SOURCE 2	± 96 dB
MAXIMUM DEPTH, SOURCE 2	± 96 dB

## Distortion (DIST)



Using this function is much like overdriving an ordinary keyboard or guitar amplifier. The input signal is multiplied by the gain control (the combined values of the parameters on its control input page labeled DRV, for Drive), then passes into a distortion mapper. Large Adjust values will cause serious amounts of distortion.

Different sounds are affected differently by DIST. Waveforms that are static (waveforms with shapes that repeat regularly and are not evolving) when they enter the DIST function will undergo more of a timbre change than the familiar sound of distortion.

The DIST function distorts each note separately, unlike a fuzz box, which adds several notes together then applies a uniform amount of distortion to all of them. Consequently your power chords may sound a little different from your expectations, but you can also get some great effects with key and velocity tracking (not to mention Sources 1 and 2!) that aren't possible with other distortion devices.

The page below shows the DIST function in the F1 block, but it can appear in other blocks as well.

```

EditProg:F1 DRU(DIST) <>Layer:1/1
Adjust:0dB Src1 :OFF
Depth :0dB
KStart:C -1 unipola Src2 :OFF
KeyTrk: 0.00dB/key DptCtl:OFF
VelTrk:0dB MinDpt:0dB
Pad :0dB MaxDpt:0dB
<more F1 DRU F2 F3 F4 AMP more>

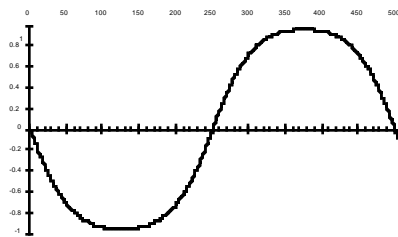
```

PARAMETER	RANGE OF VALUES
ADJUST	-96 to 48 dB
KEYTRACK START	C -1 to C 9 unipolar, C -1 to C 9 bipolar
KEY TRACKING	± 2.00 dB per key
VELOCITY TRACKING	± 96 dB
PAD	0, 6, 12, 18 dB
SOURCE 1	Control Source list
SOURCE 1 DEPTH	± 96 dB
SOURCE 2	Control Source list
SOURCE 2 DEPTH CONTROL	Control Source list
MINIMUM DEPTH, SOURCE 2	± 96 dB
MAXIMUM DEPTH, SOURCE 2	± 96 dB

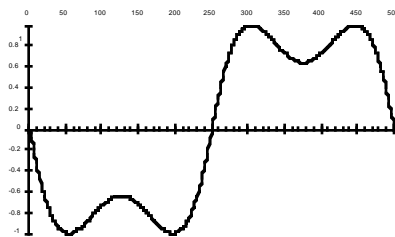
## SHAPER

The effect of SHAPER can be very unpredictable, and the mechanics of its operation lend themselves toward explanations that are more numerical than verbal. The best way for you to get a feel for the SHAPER is to start with single-cycle waveform keymaps and experiment with different values for the parameters on its control input page (labeled AMT, for Amount), and listen to the results. SHAPER tends to work best with the single-cycle waveform sounds (keymaps with IDs 112—166), and is usually less effective with acoustic instrumental sounds. SHAPER often produces numerous peaks throughout the frequency range, even at frequencies that didn't have much amplitude to begin with. These peaks can sound like resonant filters, and can even sound voice-like.

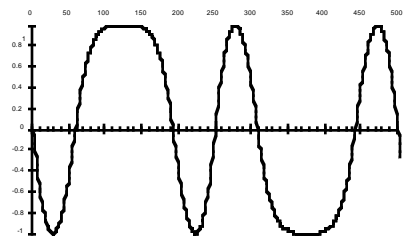
The two series of graphs that follow show the effect of SHAPER on two typical single-cycle waveforms. The first set of six graphs just below shows the evolution of a sine wave input as the value of the Amount parameter is increased. The following set of six graphs shows the effect when the Adjust parameter is increased. Each graph plots a 500-millisecond segment of waveforms cycling at frequencies of 2 Hz. Of course, these are just a few of the countless modulations you can apply to different waveforms at different frequencies.



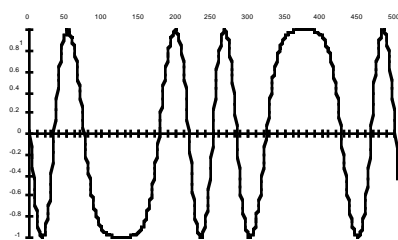
Amount = .1



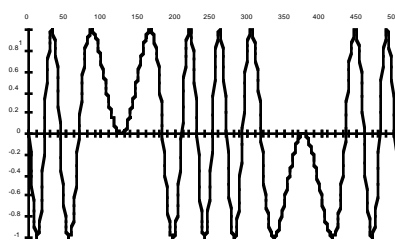
Amount = .2



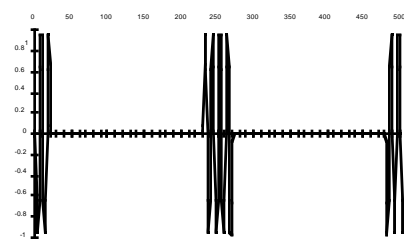
Amount = .375



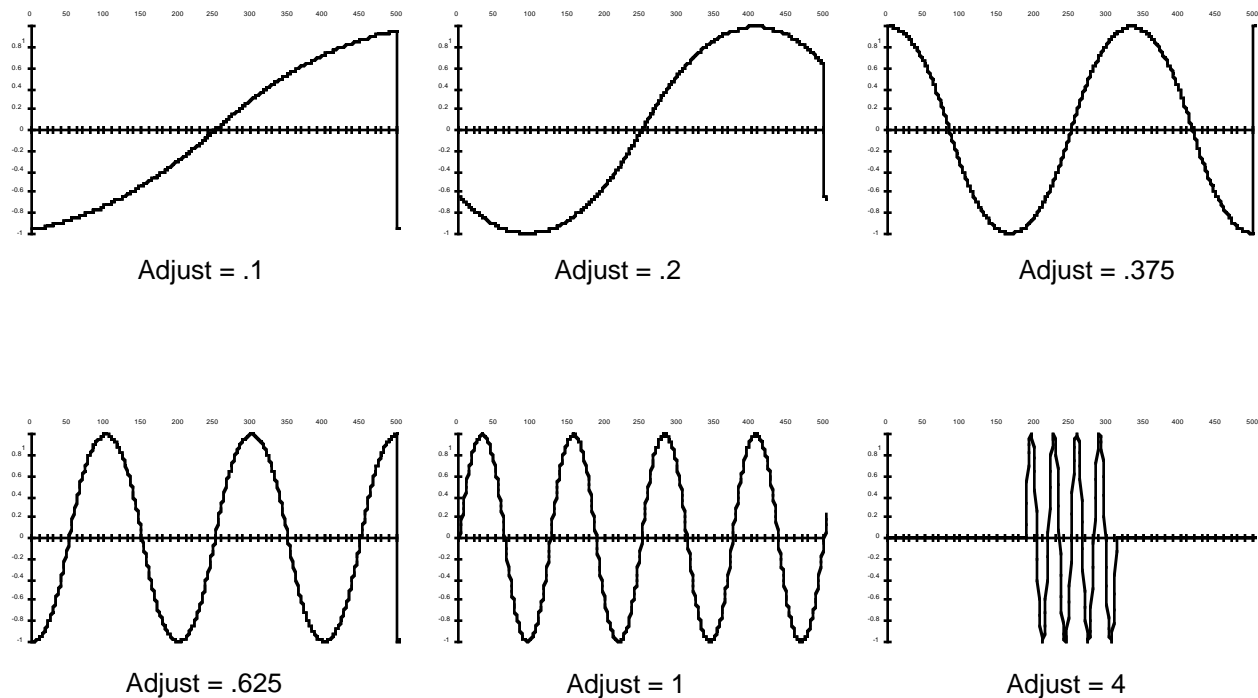
Amount = .625



Amount = 1



Amount = 4



As the SHAPER receives input signals, it evaluates the signal's level according to its own internal scale. When the SHAPER's Adjust value is at .25, an input signal moving from negative full scale to positive full scale (a sawtooth) will map to an output curve with a single-cycle sine wave shape. At an adjust value of .5, the same input signal would map to a 2-cycle sine wave output signal. Adjust values of .75 and 1.0 for the SHAPER would map to 3-and 4-cycle sine wave output signals, respectively. Beyond values of 1.0, some portions of the output will pin at zero-scale.

Small Adjust values for the SHAPER can sound much like the DIST function, but larger values will introduce dramatic changes in timbre, while DIST will have a less pronounced effect on timbre.

The following LCD page shows the SHAPER function in the F1 block, but it can appear in other blocks as well.

```
EditProg:F1 AMT(SHAPER) <>Layer:1/1
Adjust: 0.100x Src1 :OFF
Depth : 0.00x
KStart:C -1 unipola Src2 :OFF
KeyTrk: 0.000x/key DptCtl:OFF
VelTrk: 0.00x MinDpt: 0.00x
Pad :0dB MaxDpt: 0.00x
<more F1 AMT F2 F3 F4 AMF more>
```

PARAMETER	RANGE OF VALUES
ADJUST	0.100 x to 4.000 x
KEYTRACK START	C -1 to C 9 unipolar, C -1 to C 9 bipolar
KEY TRACKING	± 0.200 x
VELOCITY TRACKING	± 4.00 x

PAD	0, 6, 12, 18 dB
SOURCE 1	Control Source list
SOURCE 1 DEPTH	$\pm 4.00 \times$
SOURCE 2	Control Source list
SOURCE 2 DEPTH CONTROL	Control Source list
MINIMUM DEPTH, SOURCE 2	$\pm 4.00 \times$
MAXIMUM DEPTH, SOURCE 2	$\pm 4.00 \times$

The values for each of the parameters on the SHAPER's control input page are expressed in arbitrary quantities that represent a multiplication of the amount of shaping applied.

### Double Shaper (SHAPE2)

This is simply a series of two SHAPERS. The first is programmed exactly like SHAPER. The values of the control parameters for the second are fixed at .75 times those of the first. This can produce effects that the single SHAPER can't. If, for example, you set the Adjust parameter of SHAPE2 to 1.000, it will process the input signal with a value of 1.000, then again with a value of .75. This is not the same as processing the input signal once with an Adjust value of 1.75.

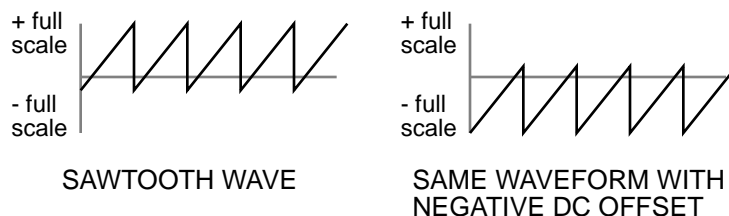
### Two-Parameter Shaper (2PARAM SHAPER)

This function is similar to the SHAPERS described above, but it has two control input pages instead of one. The F1 EVN control parameters enable you to add distortion to sine wave partials that are even harmonics of the input signal, and the F2 ODD control parameters let you add distortion to sine wave partials that are odd harmonics of the input signal.

In simpler terms, the control parameters behave like those of the regular SHAPER, but they can shape the signal about six times more than SHAPER can. 2PARAM SHAPER works by separately multiplying the input signal by the combined values of the two sets of control parameters, adding the resulting signals, and multiplying that sum by a constant, then wrapping the signal values that exceed positive or negative full scale (see the WRAP function below).

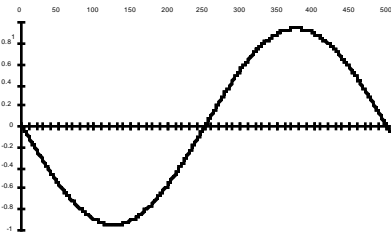
Experimentation is the key here. Start with very low values for each of the Adjust parameters, and increase them until you begin to hear an effect. Some values will create a DC offset in the signal—that is, the signal won't oscillate around the normal zero-point of the scale, but will be shifted toward positive or negative full scale. This may cause a click or thump in sounds with rapid attacks, decays, or releases. To reduce the click or thump, you can edit the AMPENV to produce a more gradual envelope.

2PARAM SHAPER works best with the single-cycle waveform keymaps (IDs 112—166).

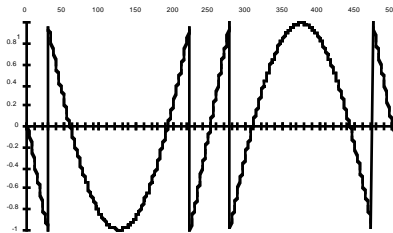


## Waveform Wraparound (WRAP)

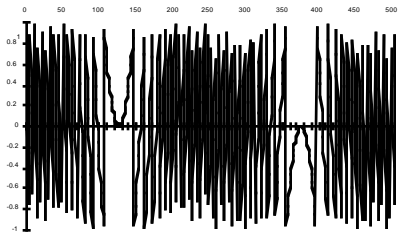
The next three graphs show the effect of various amounts of WRAP on a sine wave.



Adjust = -30

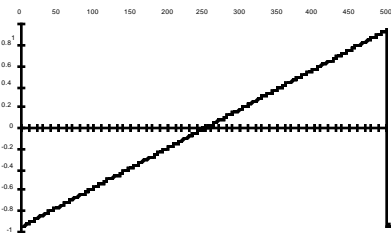


Adjust = -20

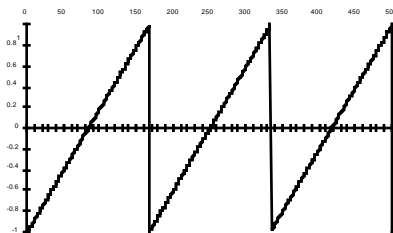


Adjust = 0

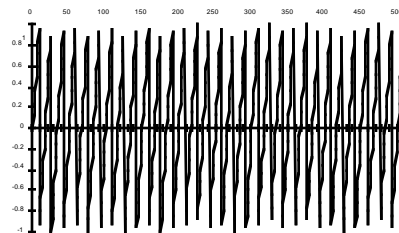
The following three graphs show the effect of WRAP on a sawtooth wave at the same frequency.



Adjust = -30



Adjust = -20



Adjust = 0

With this function you can completely mutilate a sound, and with large amounts of wrap, turn anything into white noise. At the input of the WRAP function, the signal is multiplied by the combined values of the parameters on the WRAP control input page, then multiplied by an additional gain factor of 30. If the resulting value is greater than full scale (in other words, if it's sufficiently high to clip), then instead of clipping, the waveform "wraps" back around to negative full scale, and it continues to evolve from that point. Likewise, if the resulting value is less than negative full scale, it wraps to positive full scale and proceeds from there. For any waveform, several of these wraparounds can take place before the waveform fits into the allowable range.

You'll want to try different values of the Adjust parameter to get a feel for the results of different amounts of wraparound. Look for the value that introduces a very slight amount (it will tend to be well below 0). The sound will start to buzz here and there, as a few segments of the input wrap around. As you increase the Adjust value, the buzz will increase, and the pitch of the sound will begin to disappear. Keep adding to the Adjust value, and you'll end up with white noise, regardless of the starting timbre.

The bright buzzy nature of wrapped sounds is due to the discontinuities in the waveforms of the partials as they wrap around from positive full scale to negative full scale and vice versa. If you want, you can reduce or eliminate the buzz by sending the signal through SHAPER after it goes through WRAP (assign SHAPER as the DSP function in the following algorithm block). Set the SHAPER's Adjust parameter to .25. This will map both positive full scale and negative full scale amplitudes to a level of zero, eliminating the wraparound discontinuities, but preserving the alterations in the waveform produced by WRAP. SHAPER will add its own effects as well.

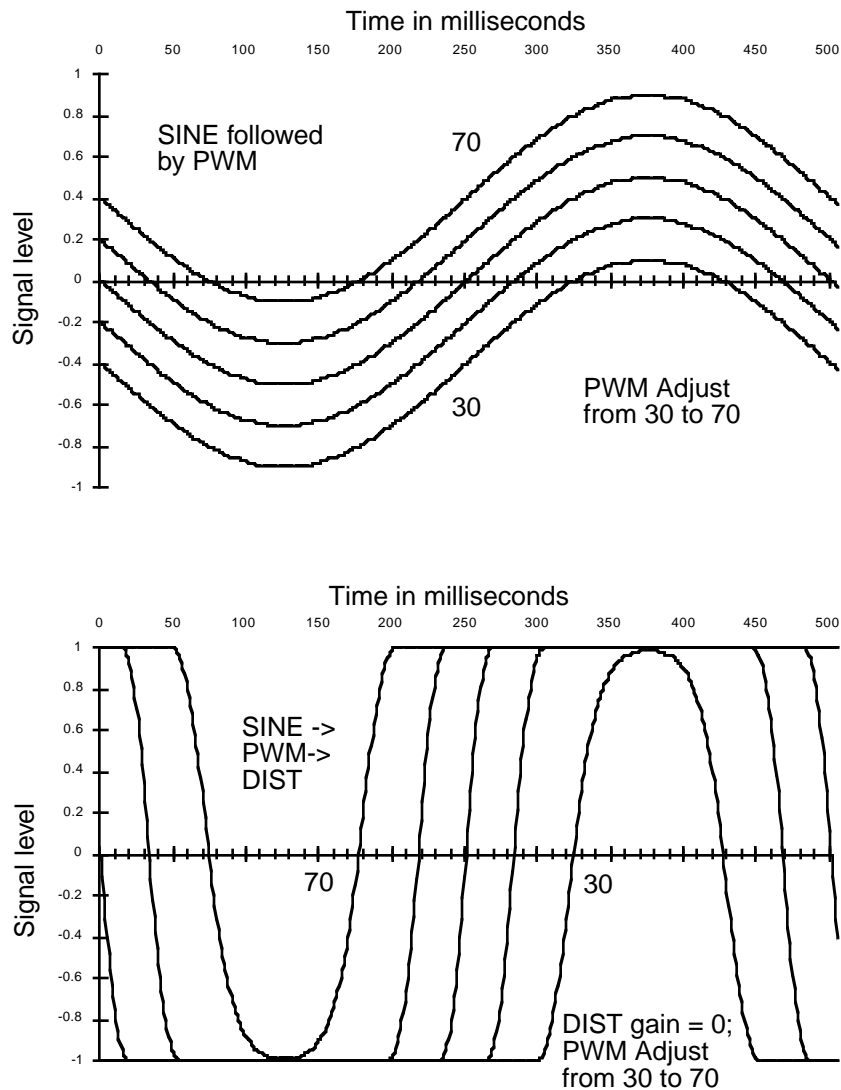
The control input page for WRAP uses the same set of parameters and ranges of values as DIST.



### Lowpass Filter with Clipping (LPCLIP)

This is a one-pole filter, which is programmed just like LOPASS. The difference with LPCLIP is that the amplitude of the input signal is multiplied by 4 before the filter. This can cause the signal to clip, which can produce interesting results. Naturally, you'd use this function only when you want to induce clipping intentionally as a component of your sound.

### Pulse Width Modulation (PWM)



Pulse width modulation can produce some classic synth sounds, and can break new sonic ground as well. Its operation is very simple; it adds an oscillating DC offset to the input signal (shifts it toward positive or negative full scale). Unless it causes the signal to clip, it won't have much effect on the signal. It's designed to be followed by DIST, with the Adjust parameter for DIST set to a fairly high level. The DIST function will drive all positive elements of the signal toward positive full scale, and negative elements toward negative full scale. The result is a rectangular wave with a width that varies according to the Adjust level of PWM.

PWM won't affect a square wave input. Sine and triangle waves will produce familiar PWM sounds. More complicated waveforms will result in discontinuous rectangular waveforms.

You can also follow a PWM algorithm block with SHAPER, since SHAPER's output is affected by the DC level of the signal.

The parameters on the PWM control input page affect the DC offset of the signal, in terms of the percentage of shift from no offset to maximum offset. At a value of 0%, there is an offset positive full scale. At 100%, the offset is negative full scale, and at 50%, there is no offset.

A typical control configuration for PWM has the Coarse Adjust set to 50%, an LFO assigned to Src1, and Src1's Depth parameter set to a value of 25%.

The page below shows the PWM function in the F1 block, but it can appear in other blocks as well.

```

EditProg:F1 WID(PWM) <>Layer:1/1
Adjust:50% Src1 :LFO1
Depth :25%
Src2 :OFF
DptCt1:OFF
KeyTrk: 0.0%/key MinDpt:0%
VelTrk:0% MaxDpt:0%
Pad :0dB
<more F1 WII F2 F3 F4 AMF more>
    
```

PARAMETER	RANGE OF VALUES
ADJUST	0 to 100%
KEY TRACKING	± 8.00% per key
VELOCITY TRACKING	± 100%
PAD	0, 6, 12, 18 dB
SOURCE 1	Control Source list
SOURCE 1 DEPTH	± 100%
SOURCE 2	Control Source list
SOURCE 2 DEPTH CONTROL	Control Source list
MINIMUM DEPTH, SOURCE 2	± 100%
MAXIMUM DEPTH, SOURCE 2	± 100%

## Waveforms Combined with Non-linear Functions

ADDED SAWTOOTH WITH NON-LINEARITY (SW+DST)  
 ADDED SAWTOOTH PLUS SHAPER (SW+SHP)  
 SHAPE-MODULATED OSCILLATOR  
 x SHAPE MOD OSC  
 + SHAPE MOD OSC  
 AMP-MODULATED OSCILLATOR

The six functions in this category do one of two things: they combine samples and waveforms with a non-linear DSP function, or they use a waveform or sample as inputs to drive non-linear functions.

### Added Sawtooth Wave with Non-linearity (SW+DST)

This function starts by adding a sawtooth wave to the layer's sample input. When the input signal from the sample is added to the sawtooth, the signal may exceed full scale, so a wraparound function similar to WRAP is performed. The result is then squared to remove any

discontinuities from the wraparound. The resulting signal has a large DC offset, so a constant of  $3/8$  is subtracted.

The parameters on the control input page for SW+DST control the pitch of the sawtooth wave.

### Added Sawtooth Wave Plus SHAPER (SW+SHP)

For this function, the sample input is combined with a sawtooth wave, then passed into the SHAPER function. The SHAPER has a constant Adjust value of .25. First, the sample input is multiplied by a constant, which may cause it to clip. Any clipping becomes part of the signal. This result is added to the sawtooth wave, which may cause the waveform to exceed full scale. If it does, the signal wraps around as in the WRAP function. This result is then put through the SHAPER.

You may want to use the Pad parameter with this function, depending on the sample you use. Pastorius fans should try this function using one of the Electric Bass keymaps. Use Algorithm 8, and start with the second and third algorithm blocks set to NONE. Set the fourth block to SW+SHP. Try setting the Keymap transposition to -12, and the SW+SHP key tracking to 100 cents per key.

### Shape-modulated Oscillator (SHAPE MOD OSC)

This function combines the sample input with a sine wave at  $1/4$  scale, does a combination of additions and multiplications on both signals, then puts the result through a SHAPER function. The amount of shaping depends on the levels of both the input signals. First, the SINE value is multiplied by the sample input value, then multiplied by a constant—any samples exceeding full scale will wrap around. The result is added to the wrapped product of the SINE value times a constant. The entire resulting waveform is then passed through the SHAPER, whose Adjust value is set by the level of the sample input. You might think of this function as an oscillator whose shape is controlled by the sample input signal.

Even if the depth value goes to zero, there will still be a non-zero final output, due to the addition of the multiplied SINE value. In this case, the output would be a slightly distorted sine wave. As the value of depth increases, the harmonic content of the output signal will rise, assuming the pitches of the sample input and the sine wave are simply related—unison, octaves, etc. Slight detuning between the two pitches will cause a slow beat frequency.

The parameters on the F2 PCH control input page affect the pitch of the sine wave. Those on the F3 DEP page affect the level of the sample input, and consequently the amount of SHAPER that's applied. If the DEPTH values exceed +5 dB, then the product of SAMPLE INPUT x DEPTH may clip, adding further harmonics through a mechanism that differs from the addition of harmonics below the +5 dB DEPTH level.

### x SHAPE MOD OSC

Available only in Algorithm 18, this function is similar to SHAPE MOD OSC, except that it multiplies its two input signals and uses that result as its input.

### + SHAPE MOD OSC

Also available only in Algorithm 18, + SHAPE MOD OSC is similar to x SHAPE MOD OSC, except that it adds its two input signals and uses that sum as its input. With this and all the modulated oscillators, let your ears be your guide.

### Amp Modulated Oscillator / Final Amp (AMP MOD OSC)

This function is available only in Algorithm 17. The sample input is multiplied by the output of a sine wave oscillator. This result is scaled by the parameters on the F3 DEP control input page, and the result is added to the original sample input and sent to the final AMP function. The parameters on the F2 PCH control input page affect the pitch of the sine wave, and consequently all subsequent results.

## Mixers with Non-linear Inputs

x AMP  
x GAIN  
! AMP  
AMPLITUDE MODULATION

### x AMP

This function can be used in the final algorithm block when it mixes two input wires into a single output. The two input signals are multiplied. The control input parameters affect the gain of the multiplied signals. The final amplitude is also affected by the settings for AMPENV and ENVCTL. Multiplying the two signals can result in outputs that differ dramatically from the input signals. You can get a wide variety of effects from this function, for example, turning an acoustic sample and a waveform (or two waveforms) into a sound that has little resemblance to the input sounds.

When two signals are multiplied, the resulting signal consists of the sums and differences of the frequencies of each partial of each signal. The frequencies of the original signals do not come through, unless they have one or more DC components (non-oscillating partials). And of course, if one of the signals has zero amplitude, the resulting signal also has zero amplitude.

If the fundamental frequencies of the two input signals are related by simple fractions (that is, if the ratios between their frequencies are something like  $1/1$ ,  $2/1$ ,  $3/1$ ,  $4/1$ ,  $1/2$ ,  $1/3$ ,  $1/4$ ,  $2/3$ ,  $3/2$ ) the resulting signal will be a harmonic sound. Its partials will be multiples one of the original fundamentals, or possibly a new fundamental. If the frequency ratios of the original signals are nearly but not quite one of these fractions, some beat frequencies will be perceived, which may or may not be useful. Of course, with equal temperament, the ratios given above are not perfectly precise (a perfect fifth, for example, has a frequency ratio of 1.4983, not  $3/2$ ). If the frequencies of the original signals are not at least closely related, the result of X AMP will be, shall we say, less than harmonious.

If the frequency of one of the original signals is below the audible range, then the result of X AMP is not a matter of harmony, but of amplitude. In this case, a tremolo effect (amplitude modulation) would be heard, because the resulting signal would periodically dip below the audible range. In fact, when you're using X AMP in the final algorithm block, you can use any sample as an LFO source by setting the Adjust parameter on the PITCH page to its minimum. To make this work, the algorithm must use one of the waveform functions in one of the blocks, and the sample signal must be routed to the x AMP block. Results will vary.

### x GAIN

This function operates like X AMP, except that it is not affected by the settings for AMPENV, since it occurs before the F4 AMP block.

### SHAPER / Final Amp (! AMP)

This function also appears in the final algorithm block when it mixes two input wires to a single output wire. The two inputs are added, then put through the SHAPER function with a fixed Adjust value of .25, then amplified according to the values for the parameters on the F4 AMP control input page.

### Amplitude Modulation (AMP MOD)

The AMP MOD function multiplies its two input signals, and the result is multiplied by a gain value that is determined by the parameters on the AMPMOD's control input page. This result determines the balance between the upper and lower wires. AMP MOD can clip the signal, so you may need to use the Pad parameter.

## Hard Sync Functions

### SYNC M AND SYNC S

These two functions appear in Algorithms 26—31, and always work in tandem. Each is a rising sawtooth oscillator. SYNC M is the “master” waveform, and SYNC S is the “slave.” These terms stem from the fact that the pitch (frequency) of the master waveform determines the repetition rate, and thus the shape, of the slave’s waveform. These functions generate their own waveforms, and do not pass the sample input through the algorithm. Consequently the PITCH function does not appear for these algorithms.

Every time the master waveform falls from positive full scale to negative full scale, the slave waveform is forced to negative full scale. You can create a wide variety of timbres by adjusting the pitches of the waveforms relative to each other. This is done with the parameters on the F1 PCH and F2 PCH control input pages. F1 is for the master, and F2 is for the slave. Pitch control is really a bit of a misnomer for the slave waveform, because its pitch is determined by the pitch of the master waveform. The fundamental of the slave waveform is forced to be the same as that of the master, since they always have the same frequencies, although the shapes of their waveforms differ.

To clarify this, assume for now that the pitch of the master waveform remains constant. When you trigger a note, both waveforms start at negative full scale. If the slave’s “pitch” control is set to a much lower value than that of the master, the master waveform will reach positive full scale before the slave. So the shape of the slave waveform will be that of a more slowly rising sawtooth wave with a relatively large negative DC offset (most of the waveform will be in the negative portion of the scale).

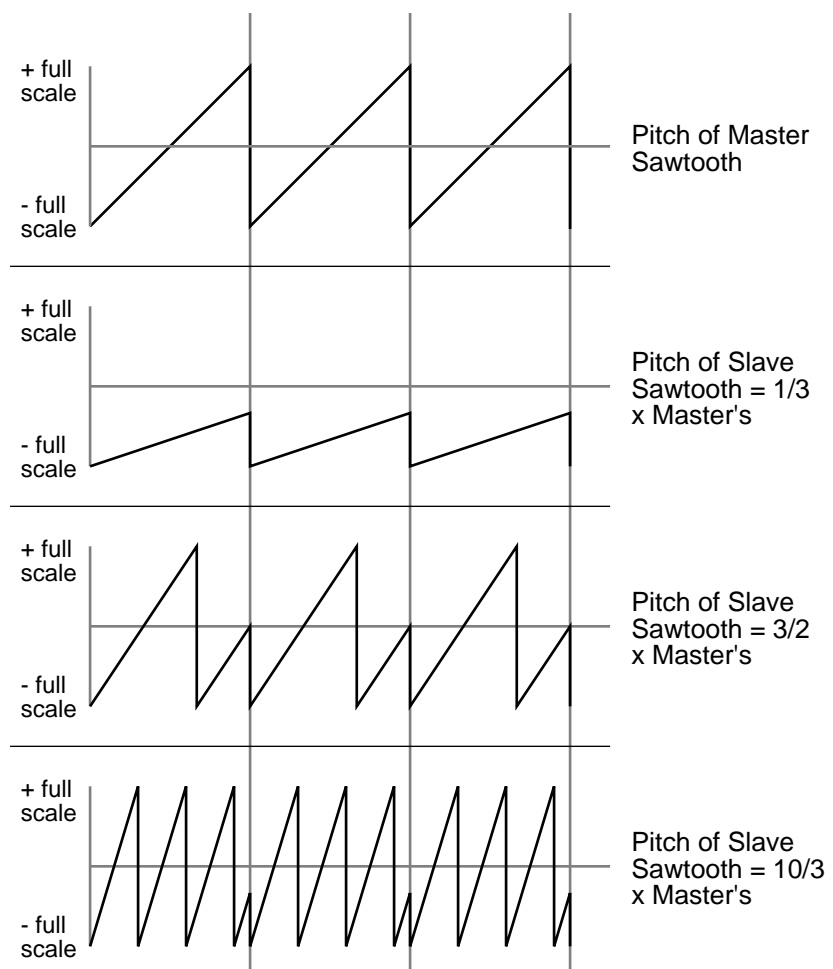
If the slave’s “pitch” control is set to a value only slightly lower than that of the master, the waveforms will be very similar, and the slave waveform will have a small negative DC offset. When the pitch settings are identical, the waveforms are identical.

If the slave’s pitch setting is higher than that of the master (which usually gives more interesting results), the slave’s waveform will alternate between a complete sawtooth cycle and a fraction of the subsequent cycle. At twice the frequency of the master, the waveform will have twice the frequency, and only the even harmonics of the master frequency will be pronounced. When the slave/master frequency ratio is nearly, but not exactly three, all harmonics will be present, but the 3rd, 6th, 9th (all multiples of 3) harmonics will be much louder than the others. This will sound like a resonant filter with multiple resonance peaks.

Because the pitch of the slave waveform is forced to be nearly that of the master, you can adjust the key tracking of the slave to values less than 100 cents per key without affecting the pitch. This will help reduce some of the harshness at the high end of the keyboard.

## DSP Functions

### Hard Sync Functions



# Chapter 15

## Sampling and Sample Editing

### Setting Up For Sampling

Before you begin sampling, you'll need to connect the proper cables from your sample source to your K2500. The cables and input jacks you use depend on the sample format you choose, and the output configuration of your sample source.

Note that sampling requires the K2500 Sampling Option. Even without the Sampling Option, however, you still have access to all of the sample editing features covered later in this chapter, provided you have some sample RAM in your K2500. Samples can be loaded from disk, or dumped into the K2500 via MIDI Sample Dump Standard (SDS) or over SCSI using the SMDI protocol. See the *Reference Guide* for information on the MIDI Sample Dump Standard and SMDI.

### Cables and Input Jacks

If you're going to be sampling from an analog source, connect a standard 1/4-inch mono or stereo cable (a typical guitar cable) from the output of your sample source to the 1/4-inch analog input jack on the K2500. If you have a K2500R or K2500RS, use the 1/4-inch jacks if you're sending an unbalanced signal, or the XLR jacks on the front panel if you're sending a balanced signal. Although it is possible to send a balanced signal on a 1/4-inch cable, avoid sending a balanced signal to the 1/4-inch jack when you're making stereo samples, since doing so can cause phase cancellation in your signals.

Using a mono cable will send the signal to the K2500's left channel. If you use a mono cable, be sure to set the Mode parameter on the Sample mode page to a value of Mono(L).

If you're using a digital sample source, you can use either a coaxial cable or an optical cable, depending on the output format of your sample source. The coaxial input on the K2500 accepts a standard male XLR fitting. The K2500's XLR input is configured as follows: Pin 1 is Common, Pin 2 is High, and Pin 3 is Low. Depending on your sample source (a commercial DAT deck, for example), you may need to use a cable (or a cable and adaptor) with an RCA connector on one end and an XLR connector on the other. In this case, you should tie the Common wire to the Low wire at the RCA end.

If your digital sample source has an optical output, connect your cable to the optical input jack above the coaxial jack on the K2500's rear panel. This jack is covered by a small plug which is easily removed. This plug should be kept in place whenever the optical input is not in use, since dust or dirt can cause the optical input to malfunction.

## Entering The Sampler - Two Different Ways

There are two different methods of entering the Sampling page. Which method you choose depends on the type of sampling you are doing - how many samples you are making and if you need custom keymaps.

The difference between the two methods primarily concerns the ease of accessing the keymap editor. Once you have made your samples, you will need to have them assigned to a keymap and have that keymap assigned to a layer in a program. Refer to the section entitled "Building a Keymap" on page 15-41, for a step by step explanation of how to create keymaps.

### From Program, Setup, or Quick Access Mode

The simplest way to enter the Sampling page is from Program, Setup, or Quick Access Mode. (Note: the **Sample** soft button is also on the Master page.) Press the soft button labelled **Sample** on one of these pages. This is a good method to use if you are making only a couple of samples, or if you want to assign each sample to its own keymap and program. Once you have created and saved your sample, you can press the **Preview** soft button. This button will allow you to quickly create a program and keymap, with that sample assigned across the entire range of the keyboard. The program is a one layer program which uses the settings from the Default program 199.

### From the Keymap Editor

This is a better method to use if you are going to be doing lots of multi-sampling, or if you need to create custom keymaps in which you have your new samples assigned across the keyboard in one keymap. Call up program 199, Default Program. Press **Edit**, then **Keymap**. Select Keymap 168, Silence, then press **Edit** again. This brings you to the Keymap editor. (In fact you can choose any program and keymap you want to start with, but by choosing these, you are starting with a "blank slate".) Now from the Keymap editor, press the MIDI Mode button. This will jump you to the Sampling page. Once you have created and saved your samples, press Exit. You will now return to the Keymap Editor page, where you can immediately assign those samples across the keyboard. Once you have created and saved your keymap, you can either exit the Keymap editor and create a program which uses your new keymap, or you can return to the Sampling page for another round of sampling

## Sampling Analog Signals

The K2500's analog sampling input is optimized for a low-impedance line level signal (-10 dBm). With a line level signal, an input gain setting of 0 dB should prevent any clipping of the sample even at maximum output from the source. You can compensate for lower input levels with the Gain parameter on the Sample mode page.

If you're sampling through a microphone, you'll probably want to use a preamp to optimize your signal-to-noise ratio. If you don't have a preamp, you can adjust the Gain parameter. A setting of 21 dB will give you reasonable results for many applications. This will increase the noise level as well, however.

Running your sample signal through a mixer before sending it to the K2500 will give you the most flexibility in controlling your signal level, since you can use its gain or pad if needed. This may add noise to the signal, however. For the cleanest possible signal, you'll want to connect your sample source directly to the K2500. The best results will be achieved by sampling from a digital source, using one of the K2500's digital sample inputs.

Assuming your connections are made, you're ready to set up your first sample recording. Select the Sample mode page (refer to "Entering The Sampler - Two Different Ways," above). The top line of the sample mode page gives you the amount of free sample memory, and the amount of free program memory.



## Input

On the Sample mode page, you'll set the conditions for your sample recording. Depending on the input type you select, a different set of parameters will appear on this page. When you've selected analog input, the page appears as in the diagram below. The differences between analog and digital sampling are discussed in the section called "Sampling Digital Signals" on page 15-10.

```

SampleMode  Samples:8192K  Memory:752K
Sample:None  Src:Int
Input :Analog  Time :1s  Mon:Off
Gain :0 dB
Rate :44.1KHz
Mode :Mono(L)
Thresh:Off  -dB 60 40 * 16 * 8 4 0
Record Auto Timer Preview

```

The digital meters at the lower right of the display give a good indication of your sample level. When you send a signal from your sample source, you should see the meters respond.

## Src

The possible values for the Src parameter are Internal(Int) or External (Ext). Choose a value of Ext when you want to sample the signal from an external source that's connected to one of the K2500's sampling inputs. Use a value of Int if you want to sample the K2500's own output. For sampling the K2500's own output, the Monitor parameter must be Off to prevent signal attenuation.

## Gain

The meters are calibrated in -dB units. A level of 0 dB indicates the maximum signal without clipping. The sample will be free of clipping as long as the meter levels don't exceed 0 dB. For optimum results, you should adjust the K2500's Gain parameter (or the gain from your sample source) so that the signal stays below 0 dB. Otherwise, the signal will be clipped, causing the loss of sample data, and usually resulting in audible distortion of the resulting sample. A few clips (fewer than 100) may not cause any appreciable distortion. You'll get the best signal-to-noise ratio with meter levels as close to 0 dB as possible, although you'll find that samples with maximum meter readings as low as -12 dB can sound remarkably noise-free.

The relatively slow LCD output of the meter levels cannot show every peak in the incoming signal. Therefore, you won't necessarily see every transient in every sample you take. You will be able to see any transient that is clipped, however, since whenever a clip occurs, the K2500 will display the word "CLIP" above the meters, and will flash the Master mode LED. It will also give you the number of clips in each sample before you save it.

The meters are inoperative during actual sampling, so make a few tests of your levels before you begin to record.

## Rate

After you've set your levels, you need to select the sample rate. You have four rates to choose from. The trade-offs that determine your best sampling rate are frequency response and storage requirements. Higher sample rates will capture more frequency content from your samples, but will take up more memory. Lower rates give you more sample time, but don't give the same frequency response as higher rates. Rates of 29.4 or 32 KHz will yield a flat response up to about 14 and 15 KHz, respectively. 44.1 and 48 KHz yield a flat response up to 20 KHz, which is

the upper limit of audibility for most humans. The lower rates may be adequate for most sounds, since many sounds have little content above 15 KHz. Sounds with a great deal of high-frequency content, such as cymbals, should probably be sampled at the higher rates. You can save memory by using lower sample rates for sounds without much high-frequency content—acoustic or electric bass, for example.

Another consideration in selecting sample rate is the K2500's transposition range during sample playback. The K2500 transposes samples by changing the sample playback rate; the higher the playback rate, the higher the pitch of the sample. The K2500 can achieve a maximum sample playback rate of 96 KHz. A sample made at 48 KHz can be transposed up only one octave, since the playback rate doubles for every octave of upward transposition. A sample made at 29.4 KHz can be transposed up approximately 21 semitones (an octave and a sixth). There is no limit on downward transposition, regardless of the sample rate.

Each portion of a sample (each individual sample element made by the K2500 during the sampling process) takes up two bytes of sample memory. A one-second stereo sample at 48 KHz consists of 96,000 individual samples (48,000 x 2), taking up 192,000 bytes (about 188K) of sample memory. The same sample taken at 32 KHz takes up about 125K. A one-second mono sample taken at 32 KHz takes up about 63K.

If you plan to do a lot of sampling, you may want to consider adding sample memory to your K2500. SIMMS (Single In-line Memory Modules) are available at your dealer, or at most computer stores or mail-order houses. Be sure to read "Choosing SIMMs for Sample RAM" in Chapter 8 of the *K2500 Reference Guide* before you go SIMM shopping, though.

At a sampling rate of 44.1 KHz, each megabyte you add increases your sample time by about 12 seconds. The chart below lists a few standard sample RAM configurations and their total sample time capacity (in seconds) at various sample rates.

		Sampling rate in KHz			
		29.4	32.0	44.1	48.0
2 x 1M	Mono	35	32	23	21
	Stereo	17	16	11	10
2 x 4M	Mono	140	128	92	84
	Stereo	70	64	46	42
2 x 16M	Mono	560	512	368	336
	Stereo	280	256	184	168

### Mode

Use the Mode parameter to select mono or stereo sampling. (Keep in mind that stereo samples take up twice as much memory as mono samples.) Use a value of Mono(L) for a mono signal. You can use either Mono(L) or Mono(R) to isolate either the left or right side of a stereo signal.

Audio Sampling Input doubles as a two channel "drum" trigger, allowing audio signals to trigger samples. On the Sample page, set Mode to Trigger. Adjust Thresh to control triggering sensitivity. This triggers the currently assigned click program. The left input will trigger click

key note number +1, right input will trigger click key +2. The click key and click program can be accessed on the Song Mode MISC page.

### Threshold (Thresh)

The Thresh parameter controls when the K2500 actually begins sampling incoming signals. If you set it to a value of Off, sampling begins immediately when you press the **Record** soft button. Otherwise the K2500 waits for the incoming signal to exceed a specified threshold before beginning to record. You can set the threshold from -90 to 0 dB, in 6 dB increments.

Sampler recording can also be triggered via the keyboard. On the Sample page, set Thresh to "key", then press **Auto**. Striking a MIDI note event now will trigger the sampler, set the sample placement on the keyboard and relative tuning all in one easy step making sample mapping easy and intuitive.

### Time

The Time parameter lets you determine how long the sample will be. The available sample time is a function of the sample rate and the amount of available sample memory. The K2500 calculates this automatically, and limits the maximum value of the Time parameter accordingly. At a value of zero for this parameter, the K2500 will not record.

### Sample

The Sample parameter lets you select any sample in memory for auditioning. This is a convenient way to listen to the samples you've made without having to manually create keymaps and programs for them. With a value of None for this parameter, the K2500 plays the last program or setup you selected before entering Sample mode. The list of values includes all ROM and RAM samples.

When you select a sample for auditioning, the K2500 automatically creates a temporary keymap and program, based on the settings for Default Program 199—which is a simple single-keymap program with few controller assignments, and the effects set to 0% wet (100% dry). Any edits you've made to Program 199 are reflected in the sample you audition. When you exit the Sample mode page, the temporary keymap and program disappear until the next time you audition a sample. You can create regular RAM keymaps and programs using the **Preview** soft button; see the discussion of the **Preview** button in the section below called "Recording Samples."

If you don't have enough free program RAM, you may be unable to audition samples, since the K2500 won't have enough RAM to create the temporary keymap and program. In this case, deleting a few objects from RAM will restore the audition feature.

### Monitor (Mon)

The Monitor parameter provides a convenient way to listen to what you're recording. When set to a value of On, any signal received at the analog sample input will appear at the K2500's MIX outputs and the headphone jack. Adjusting the input gain will affect the monitor gain as well. A clean monitor signal, however, does not guarantee a distortion-free sample. Always check the meters on the Sample mode page and look for the CLIP indicators to ensure that your sample is free of clipping. Note that the Mon parameter is not available when the Input parameter is set to a value of Digital. The Monitor feature applies only to the analog sampling inputs. You should monitor digital sources from the sources themselves.

For sampling the K2500's own output, the Monitor parameter must be Off to prevent signal attenuation.

## Recording Samples

Press the **Record** soft button to begin the sample recording process. If the Thresh parameter is set to a value of Off, recording will begin immediately, and will continue for the number of seconds indicated by the Time parameter. The display will indicate that recording is in process. Any other value for the Thresh parameter will cause the K2500 to wait until the specified threshold is exceeded, then recording will proceed normally. The display will indicate that you're making a threshold recording, but won't actually begin recording until the threshold is exceeded.

End the sampling process (either to save what you've done, or to abort) by pressing the **Stop** soft button.

When recording is complete, and you've pressed the **Stop** button, the K2500 will prompt you to strike a root key. The sample is assigned to the key you strike. This "root" is the key at which the sample will be played back without transposition. When sampling pitched sounds, it generally makes sense to assign a root key that matches the pitch of the original sample, although you can set the root key anywhere you like. If you press the **Default** soft button, the K2500 uses C 4. You can change the root key at any time on the MISC page in the Sample Editor.

When the root key has been assigned, the K2500 asks you if you want to save the sample. At this point the display will show one of two things—the number of clips, or if no clips occurred, the maximum level (in dB) of the sample signal.

You can listen to the sample before deciding whether to save it. If you decide not to keep the sample, press the **No** soft button, and you'll return to the Sample mode page. If you press **Yes**, you'll see the normal Save dialog. When you've saved the sample, you'll return to the Sample mode page. You'll also have the opportunity to name the sample. A recommended convention for naming samples is to include the root key as part of the name. This is particularly useful for pitched samples. Including the root key in the sample name helps when you are creating a keymap, because it tells you how much transposition of the sample you will hear depending on its key assignment.

Once the sample is recorded and saved, you may want to edit it, using the TRIM page, LOOP page, or any of the sample DSP functions.

### The Auto Soft Button

To save time when sampling with either the Analog or Digital inputs, you can use the **Auto** soft button. This will begin sampling immediately (or when the sample threshold is exceeded if you have the Thresh parameter set to a value other than Off), and will return you to the Sample Recording page when the sample is complete and you've pressed the **Stop** button. The root key is automatically assigned at C 4, and the sample is automatically saved at the next available ID above 199.

### The Timer Soft Button

If you need to delay the beginning of your sample recording, you can press the **Timer** soft button instead of the **Record** or **Auto** soft buttons. This will begin a ten-second countdown before sample recording actually starts. The display will show the countdown. When the countdown reaches zero, The Program, Setup, MIDI, and Master mode LEDs will flash three times. If you have the Thresh parameter set to a value of Off, sample recording will start immediately after the LEDs flash. If you have the Thresh parameter set to a value other than Off, sample recording will begin when the countdown has finished and the threshold level is exceeded.

### The Preview Soft Button

When you've finished taking a sample, you can press the **Preview** soft button to automatically create a keymap and program using the new sample. It uses the settings for the Default program 199 as a template. Unlike the temporary keymap that's created when you audition a sample (and disappears when you select another sample), the preview keymap and program are stored in RAM and can be selected at a later time. The program and keymap will have the same name as the sample.

When you press the **Preview** soft button, the Bank dialog appears, prompting you to select a bank where the preview program will be stored. Select a bank, then press the **OK** soft button. The K2500 will create a keymap and a program, using the lowest available ID numbers in that bank for both the keymap and the program. The display will tell you the ID number of the new program.

## Sampling the K2500's Output

You can sample the K2500's own sounds when in Analog sampling mode. To do so, set the Src parameter on the Sample mode page to a value of Int. Then, just press the **Record** soft button and start playing.

The K2500's "sample-while-play" capabilities offer a number of useful possibilities. It allows you, for example, to create composite sounds made up of several K2500 sounds or even sequences. This can help you make efficient use of the K2500's polyphony. By building composite sounds from other composite sounds, you could actually cause a frighteningly large number of K2500 sounds to become a single sample. Assuming that your imagination isn't limited, your only constraint would be the amount of sample RAM installed in your machine.

## Using the Sampler to Create Overdubs

One very exciting way that "sample-while-play" increases the power of your K2500 is by giving you the capability to add overdubs to a sequence you've created with K2500 Song mode. The procedure is a little tricky, so follow the steps below carefully. Also, remember when using a microphone that you'll need to boost the signal a bit before plugging into the K2500. The line output of a high-quality mixer will give you a properly pre-amped signal.

Here's how to do it:

1. **Create a sequence in Song mode and leave at least one track free for your overdub.**

For this example, we'll use track 5 for the overdub track.

2. **Find the lowest unused program number; let's say "213".**

3. **Now press the Song button to enter Song mode.**

Make sure that the current song is the correct sequence.

4. **Set the track you've allocated for the overdub to Record mode. Press the Record button; the "Rec Ready" indicator will flash on the top line of the screen, as shown below. Then set this track's program to be the lowest unused program number (as determined above). For this example, it's "213".**

You'll have to use the numeric keypad to enter the program number.

Record track is set to track 1

```

SongMode:MAIN  Used:  0%  REC READY
CurSong:202*NewSong  Tempo:120
RecTrk :5  Vol:127 Pan:64  Mode :Merge
Program:213 Not Found  Locat: 1:1
  □ □ □ □
Track  :P P P P R - - - - -
Channel:1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Record Play Stop Erase MISC MIXER

```

5. **Press the Stop button.**

6. **Now edit the overdub track.**

Make sure that the program parameter is not highlighted when you press the Edit button. Otherwise you will enter the Program editor.

7. **Press the STEP soft button to enter the Step editor.**

8. **Play C4.**

This "note" will be recorded by the Step editor on the overdub track.

9. **Press Stop, followed by the Done soft button to leave the Step editor.**10. **Now press the EVENT soft button to enter the Event editor.**11. **Move the C4 note event from 1:1.000 to 1:1.101 by entering "11101" in its Bar:Beat:Tick column. Then set the note's velocity to 127 and its duration to the maximum value.**

The simplest way to set the note's duration to its maximum value is to type "9999999" in the note duration column. When you press the Enter key, the value shown in this column will be updated to reflect the maximum possible duration for the note, as shown below.

Note duration set to maximum

```

EditSong:EVENT (Ch 2) <> track:2
1:1 1:1.000 CTRL BANK 2
1:1 1:1.000 PCHG 13
1:1 1:1.000 CTRL VOL 127
1:1 1:1.000 CTRL PAN 64
1:1 1:1.101 C 4 v127 ^ 64
21:1 21:1.000 CTRL PAN 64 79.378
Cut Copy Paste View AllOn Done

```

12. **Press Exit and save the changes to the song.**

Do not change the song's current record track.

13. **Press the Sample soft button (Program, Setup, Q Access, or Master page); enter sampler.**14. **Set the Sample to "None" and Source to "Ext". Then choose a sample rate and the longest available sample length. Finally, set Thresh to "Key".**15. **Press the Record soft button.**

Be careful not to press any keys yet.

16. **Press the Program button to enter Program mode.**17. **Double press the Left and Right cursor buttons to start the song and begin recording the sample (overdub).**

Alternatively, you can press the Play/Pause button on the K2500 keyboard models.

18. **Double press the Left and Right cursor buttons to stop the song (and recording).**19. **Return to the Sample page and press the Stop soft button to stop sampling.**20. **Save the sample, then press the Preview soft button, choosing the bank with the lowest unused program number (in this example, the 200's bank).**

The new sample will be made into the program with the lowest unused number (in this example, "213"), and it will play along correctly with the song, owing to the C4 note that you've recorded in the overdub track. To add more overdub tracks, copy the track you've just made to an empty track, then edit its program change command to be the next highest unused program. You can then begin recording the next overdub (sample) by restarting at step 13, above.

## Sampling Digital Signals

The process for sampling through either of the digital inputs is essentially the same as that for sampling analog signals, although there are a few additional parameters associated with digital sampling formats.

You'll notice that the Sample mode page changes considerably when you change the value of the Input parameter from Analog to Digital. There are a few more settings to be made before you start recording.

```

SampleMode Samples:8192K Memory:118K
Sample:None
Input :Digital Time :1s
Cable :Coaxial
Format:AES/EBU
Mode :Stereo
Thresh:Off -dB 60 40 * 16 * 8 4 0
Record Auto Timer Preview

```

The first difference is the fact that there are no parameters for gain and sample rate. There's no need for a gain parameter because with digital sampling, since you're making an exact digital copy of the source signal. The Rate parameter is excluded because the K2500 automatically recognizes the source sample's rate and sets its own rate accordingly. Also, the Mon parameter does not appear when sampling digitally. Any monitoring you wish to do must be done from the sample source.

### Cable

Set the Cable parameter to a value of Coaxial or Optical, depending on the type of cable you're using. Many consumer products with digital output provide an optical jack. There is a small plug covering the K2500's optical input; this plug must be removed before you can connect the optical cable to the K2500. The plug should be replaced whenever the optical input is not in use.

If you plan to use a coaxial cable, select a value of Coaxial for the Cable parameter. If you're sampling from a consumer product like a CD player, you'll probably need to get an adaptor, or a cable with an RCA fitting on one end and an XLR (male) fitting on the K2500 end. Some professional products have an XLR fitting for their outputs, so an ordinary microphone cable will be suitable in most cases. Longer cables can cause signal loss, however, so if you need a long cable, you may need to get a special cable designed for digital information transfers. There's detailed information about cables for sampling on page 15-1.

### Format

Use the Format parameter to tell the K2500 the format of the incoming sample. Most consumer products use SPDIF (Sony / Philips Digital Interface Format), while most professional machines use the AES / EBU (Audio Engineering Society / European Broadcast Union) format. Refer to the owner's manual of your sample source for information regarding its digital format.

The Mode, Time, and Thresh parameters function for digital sampling just as they do for analog sampling.

When you begin recording, a box appears in the display that indicates that the K2500 has locked on to the digital signal. You'll see the word "LOCK," and the rate at which the signal was sampled will be shown. The K2500 automatically sets itself to this rate. If you don't see this box, check to make sure that the signal is getting to the K2500.



The K2500 may not display the rate when you're using a rate other than the frequently-used rates of 32, 44.1 or 48 KHz. This is normal, and although the rate may not be displayed, the K2500 will still lock on to it.

## Using the Digital Outputs

The K2500 features digital audio outputs with XLR connectors. There are two points to keep in mind if you wish to use them. First, the Mode parameter on the Sample mode page must be set to a value of Analog to enable the digital outputs. Second, the analog input must be disconnected from any cables.

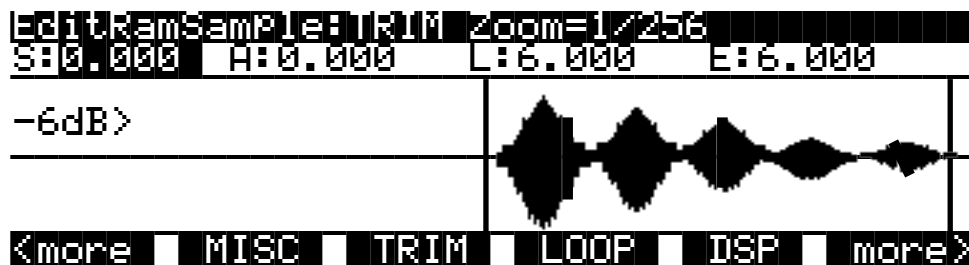
If you have a cable connected to the analog input, that input will be routed to the digital outputs. This effectively makes the K2500 into an analog-to-digital convertor, which can be very useful. The digital out will have the same signal that is sent to the MIX outs. So if you have all of your separate outs plugged in (and therefore have no signal coming out of the MIX outs), you will not have any signal coming out of the digital outs.

## Editing Samples

Most of the functions within the Sample Editor follow a general pattern. There are two ways to enter the Sample Editor. If you start from Master mode and press the **Sample** soft button, then select a sample and press EDIT, you can hear the isolated sample. If you want to hear the sample in the context of a program, start by entering the Sample Editor through the Program Editor: start by selecting a program in Program mode—usually the program containing the sample you want to edit. Press the EDIT button to enter the Program Editor. Press the **KEYMAP** soft button to view the KEYMAP page. The KeyMap parameter is selected (highlighted) when the page appears. Press the EDIT button to enter the Keymap Editor. The KeyRange parameter is selected when the page appears. The notes within the currently selected key range are the only ones that will be affected by your edits. You can hold the Enter button and trigger notes to select different key ranges.

If you want to select a different sample, use the cursor buttons to select the Sample parameter. Use the Alpha Wheel to select a sample. Press the EDIT button once more, and you'll enter the Sample Editor. (Pressing the EDIT button while in the Keymap Editor will enter the Sample Editor regardless of which parameter is selected.) The effects of the current program will be applied to the sample.

The TRIM page appears when you enter the Sample Editor. A representative TRIM page is shown below.



There are three basic sample editing pages—TRIM, LOOP, and MISC (Miscellaneous). The soft buttons for these pages are visible when you enter the Sample Editor. The **DSP** soft button is visible as well if you're editing a RAM sample; pressing it will take you to the DSP function page, where you can select a DSP function with the Alpha Wheel or Plus/Minus buttons.

The **DSP** soft button does not appear if you're viewing a ROM sample. Instead you'll see a **Link** soft button.

The **<more>** soft buttons will take you to the soft buttons for the other functions. You can trigger notes at any time while you're editing, to hear your changes as you make them.

## The Function Soft Buttons in the Sample Editor

In addition to the **MISC**, **TRIM**, and **LOOP** soft buttons, which select Sample Editor pages, there are several function soft buttons. As with other K2500 editors, the function soft buttons are labeled with upper and lower case letters, to distinguish them from the page selection soft buttons, which are labeled in all capital letters. The **<more>** soft buttons give you access to the other soft buttons that are available.

### Zoom- and Zoom+

These buttons are active only when you're viewing the TRIM and LOOP pages. They increase or decrease the resolution of the waveform display, enabling you to see a larger or smaller segment of the waveform of the currently selected sample. The top line of the display indicates the zoom position in terms of a fraction—for example, 1/256—which indicates the number of individual sample elements represented by each display pixel. A value of 1/256 means that each pixel represents 256 individual sample elements. The maximum zoom setting of 1 shows you a very small segment of the sample. The minimum setting of 1/16384 shows you the largest possible segment of the sample. Each press of a **Zoom** soft button increases or decreases the zoom by a factor of 4.

As a convenience, the Program and Setup mode buttons also serve as zoom buttons while in the Sample Editor. You can press the two left soft buttons together to toggle between the default zoom setting and your current zoom setting.

### Gain- and Gain+

Also active only for the TRIM and LOOP pages, these buttons increase or decrease the magnification of the currently displayed sample waveform, enabling you to see the waveform in greater or lesser detail. At the left of the display, you'll see the magnification setting, which is expressed in dB units. You can adjust the magnification from -72 dB (maximum magnification) to 0 dB. This doesn't affect the actual amplitude of the sample, only the magnification of its display.

As a convenience, the MIDI and Master mode buttons also serve as gain adjustment buttons while in the Sample Editor.

The simplest way to think of the **Zoom** and **Gain** buttons is to remember that the **Zoom** buttons control the left/right magnification of the waveform, while the **Gain** buttons control the up/down magnification. Neither button has any affect on the sound of the sample. You'll often use the **Zoom** and **Gain** soft buttons together to focus in on a particular sample segment, then magnify it to see it in close detail.

For example, you might want to zoom out to view an entire sample waveform, to decide which segment you want to edit. You could then zoom in to focus on a particular segment. Once you've zoomed in, you may want to boost the Gain to enable you to set a new Start (S) point with greater precision, or ensure that you get a smooth loop transition.

### Abort

Use the **Abort** soft button to cancel a sample dump before it's complete. You'll be prompted to verify whether you really want to cancel the dump.

### Split

The **Split** soft button enables you to create two mono samples from a single stereo sample, or to split up a multi-root block of samples. This button is available only when the currently selected sample is a stereo sample. When you press this button, the K2500 will prompt you: Split this sample? When you press the **Yes** soft button, you'll be prompted to enter the ID for the first sample. Select an ID with the Alpha Wheel or Plus/Minus buttons, then press the **OK** soft button. The split samples will automatically be assigned IDs, starting with the ID you select.

Splitting stereo samples enables you to use the separate sides individually, or to phase the samples by assigning each side to a separate keymap, then delaying one of the layers slightly.

#### Units

With the **Units** soft button you can change the units used to display the locations of the current sample's Start, Alt, Loop and End points. The default setting displays these points in seconds, that is, the number of seconds from the physical start of the sample. Pressing the **Units** soft button will change the units to samples, that is the number of individual sample elements from the physical start of the sample. Press it again to return to a view of the sample in seconds.

As a convenience, the Quick Access mode button also serves as a units button while in the Sample Editor.

#### Link

The **Link** soft button lets you fix the interval between the Start, Alt, Loop and end points of the current sample, so it remains constant when you move one or more of the points. This is done by selecting the desired parameter with the cursor buttons, then pressing the **Link** soft button. The colon (:) following the parameter's name will change to an arrow (>) to indicate that it is linked. You can link any or all of the four sample points. When sample points are linked, moving one of them will move the linked points correspondingly. For example, suppose the current sample's Start (S) point is 0.0 seconds, and its Alt (A) point is 0.5 seconds. The interval between the sample's Start and Alt points is exactly half a second. If you select the Start parameter, then press the **Link** soft button, the Start point will be linked. This won't have any effect until you link at least one more point. If you select the Alt parameter and press the **Link** soft button, the Start and Alt points will be linked. Now if you move the Start point to 1.0 seconds, the Alt point will automatically move to 1.5 seconds, preserving the half-second interval between Start and Alt. To remove the link on any of the points, select the point again, and press the **Link** soft button again. The arrow will change to a colon, indicating that the link has been removed.

As a convenience, the Song mode button also serves as a link button while in the Sample Editor.

#### Name, Save, Delete, and Dump

These soft buttons are similar to the **Name**, **Save**, **Delete**, and **Dump** soft buttons in the other editors, initiating the corresponding dialogs to name, save, delete, or dump the currently selected sample.

When you press the **Save** soft button, and choose to save the sample to a new ID, the K2500 will ask you if you want to copy the sample data. If you answer Yes, the K2500 will make a separate copy of the sample. If you answer No, the K2500 will simply mark the location of the original sample data and share the sample between the original and the edited sample. This can save a great deal of memory space. If you delete a sample that's partially or completely shared with another, the K2500 deletes only the portions that are unused by the shared sample, always optimizing its memory for maximum storage capacity.

Note that if you use the Utility function to view the objects currently stored in the K2500, you'll see each object listed separately, including shared samples. The shared samples will each indicate their size, even though they're referring to the same memory location. This might lead you to believe that you're using more memory for samples than you actually are. If you use the Utility function to calculate your total sample memory usage, remember not to include any shared samples in the total.

#### The Page Buttons

The soft buttons labeled in all capital letters select the various pages in the Sample Editor: **MISC**, **TRIM**, and **LOOP** (as well as **DSP** when you're editing RAM samples).

## Miscellaneous (MISC)

On the MISC page, you'll set several parameters that affect the behavior of the current sample. These parameters affect the entire sample. The diagram of the MISC page shows a ROM sample, and since the DSP functions cannot be applied to ROM samples, the **DSP** soft button is not available. In its place is the **Link** soft button, which enables you to maintain equal times between various points in the sample—Start and Alt, for example.

The default values shown in this diagram reflect the settings for the Default program 199.

```

EditRomSample:MISC
RootKeyNum   :G#1      LoopSwitch:On
PitchAdjust  :-12ct    Playback   :Normal
VolumeAdjust :0.0dB    AltSense   :Norm
AltVolAdjust :-0.5dB   IgnoreRelease:Off
DecayRate    :3dB/s    SampleSize=101Kb
ReleaseRate  :106dB/s  SampleRate=19200Hz
<more>  MISC  TRIM  LOOP  Link  more>
  
```

PARAMETER	RANGE OF VALUES
Root Key Number	C -1 to G 9
Pitch Adjust	Variable (depends on sample rate)
Volume Adjust	-64.0 to 63.5 dB
Alternative Volume Adjust	-64.0 to 63.5 dB
Decay Rate	0 to 5000 dB per second
Release Rate	0 to 5000 dB per second
Loop Switch	Off, On
Playback Mode	Normal, Reverse, Bidirectional
Alternative Sample Sense	Normal, Reverse
Ignore Release	Off, On

## Root Key Number

The root key number represents the key at which the sample will play back without transposition (at the same pitch as the pitch of the original sample). When you're creating your own samples, the key you strike as the root key will be the key you see as the value for this parameter.

## Pitch Adjust

Use this parameter to change the pitch of the sample relative to the key from which it's played. Setting a value of 100 cts, for example, will cause the sample to play back one semitone higher than normal. This parameter is handy for fine tuning samples to each other if they're slightly out of tune.

## Volume Adjust

Uniformly boost or cut the amplitude of the entire sample. Compare this to the DSP Volume Adjust parameter, which lets you boost or cut the amplitude of a specified segment of a RAM sample.

#### **Alternative Start Volume Adjust (AltVolAdjust)**

This parameter sets the amplitude of the sample when the alternative start is used. See Chapter 6 for a discussion of AltSwitch.

#### **Decay Rate**

Use this parameter to specify how long the sample will take to decay (fade) to zero amplitude when a note is sustained, either by holding the key (or other note trigger), or with the sustain pedal. The higher the value, the faster the sample will decay. The sample starts decaying as soon as the note is triggered, and decays with a linear curve. This decay affects each sample individually, and begins as soon as the note starts. It operates independently of the decay in the AMPENV, which affects every note in the layer.

#### **Release Rate**

The release rate determines how long the sample will take to decay to zero amplitude when the note trigger is released. the higher the value, the faster the release rate. This release affects each sample individually, and begins as soon as the note is released. It operates independently of the release in the AMPENV, which affects every note in the layer.

To create an extended sample loop that will play data after the sample's loop on key-up, set the Alt sample pointer after the sample end pointer, then set a relatively low value for the release rate.

#### **Loop Switch**

This parameter activates or deactivates the looping of the currently selected sample. When set to On, the sample will loop according to the settings on the LOOP page. When set to Off, the sample will play through to its End point and stop.

#### **Playback Mode (Playback)**

This parameter lets you modify the direction in which the sample is played. Set it to a value of Reverse if you want the sample to play from its End point to its Start (S) point. Choose a value of Bidirectional to cause the sample to play from Start to End, then reverse direction and play again from End to Loop and back, repeating until the note trigger is released (this works only when the Loop Switch parameter is set to On).

#### **Alternative Sample Sense (AltSense)**

This provides a convenient way to activate the alternative start of a sample. When set to Normal, the alternative start will be used when the Alt Switch control is On (this is set on the KEYMAP page), or when the control source assigned to it is above its midpoint. When set to Reverse, the alternative start will be used when the Alt Switch control is Off, or when the control source assigned to it is below its midpoint.

#### **Ignore Release (IgnRelease)**

When set to a value of Off, the sample will release normally when the note trigger is released. When set to On, the note will not release, even when the note trigger is released. This setting should be used only with samples that normally decay to silence; non-decaying samples will play forever at this setting. This parameter is equivalent to the IgnRelease parameter on the LAYER page, but affects only the currently selected sample.

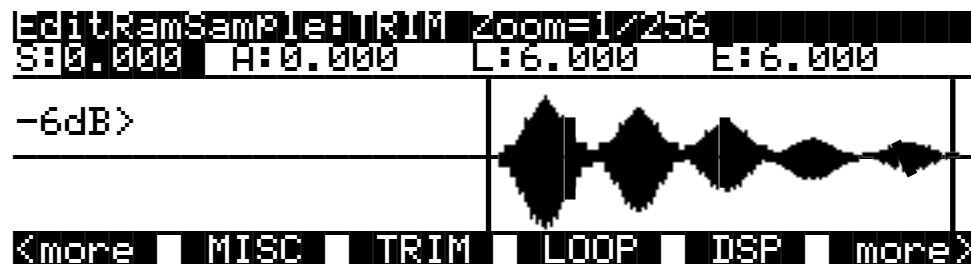
## Sample Size and Sample Rate

These are not editing parameters. They're put on this page to give you a convenient place to check the size in Kilobytes of the current sample, and the rate at which it was sampled.

## TRIM

The TRIM page lets you set the Start, Alt, Loop, and End points of the current sample. The top line tells you whether you're editing a ROM or RAM sample, and indicates the zoom setting. At the left of the display is the Gain (display magnification) setting. This Gain setting doesn't affect the amplitude of the sample, just the view in the display.

In the diagram below, the sample points are expressed in individual sample elements. Pressing the **Units** soft button (or the Quick Access mode button) will display them in seconds.



The four parameters on this page are Start (S), Alternative Start (A), Loop (L), and End (E). Selecting these parameters and adjusting their values enables you to modify how the sample plays back when notes are triggered.

There are four vertical lines that indicate the settings of the four parameters. You'll see all four lines only if the values for each of the four parameters are different; otherwise, the lines will overlap. Selecting one of the parameters will move the line corresponding to that parameter to the center of the display, where it will flash to indicate the parameter's position. The line corresponding to the currently selected parameter will flash. Turning the Alpha Wheel will move the sample waveform relative to the line. The line remains in the center of the display, and the waveform shifts to indicate the new position of the point. You can also use the alphanumeric buttonpad to enter new values directly. Press the ENTER button to register the values you enter.

The Start (S) point determines the beginning of the current sample. You can truncate the beginning of the sample by increasing the value of the Start (S) parameter. You might do this to remove silence at the beginning of a sample, or to remove some or all of the attack. You can't decrease the Start point of RAM samples below zero, but if you want to add silence at the beginning of a RAM sample to create a delay, you can use the Insert Zero DSP function to add as much silence as you like. The start points of ROM samples can be set lower than zero (you can set negative numbers for the Start (S) and Alt (A) parameters). Doing this will cause portions of other samples to be played, which can create interesting effects.

Note that for RAM samples, you won't see any waveforms displayed to the left of the Start point, but you will for ROM samples.

The Alt (A) parameter lets you set a second, optional start or end point for the current sample. The Alt will be used when the Alt Switch parameter on the KEYMAP page is set to On, or when it's set to a specific control source and that control source is generating a value of more than +.5 (for example, if you assign MWheel as the control source for the Alt Switch parameter, the Alt

will be used when the Mod Wheel—or whatever control source you have set to send MWheel—is above its halfway point). The Alt can be set before, after, or at the same point as the Start or end.

If you set the Alt after the End, you can extend the play of looped samples. Normally, looped samples will play through to the End, then will loop back to the Loop point, and continue looping like this until the note is released, when they go into their normal release. If the Alt is set after the End, looped samples will loop in the same way while notes are sustained. As soon as you release the notes, however, the samples will play through to the Alt point before going into release.

The Loop (L) parameter sets the beginning of the looped portion of the current sample. Although you can adjust this parameter while you're on the TRIM page, you'll normally want to adjust it while viewing the LOOP page, so you can see the loop transition points, enabling you to create as smooth a loop as possible. The Loop can be set at any point before the End, including before the Start and Alt. If you try to move it after the End, the End will be moved with it.

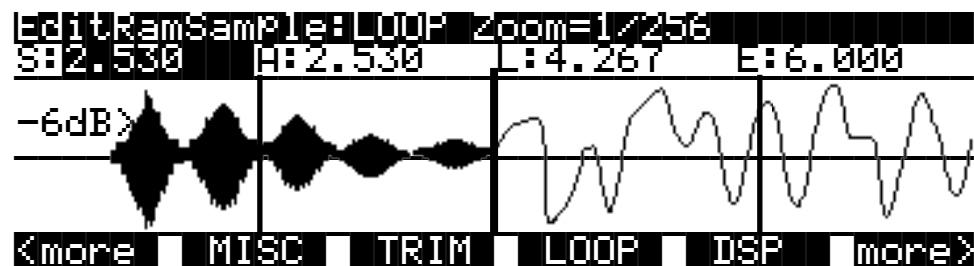
The End (E) parameter sets the point at which the current sample will stop playback. Typically you'll use this parameter to trim unwanted silence off the end of a sample, although you can use it to shorten a sample as much as you want.

If you want to truncate a sample to save memory, there are two points to keep in mind. First, if the Alt parameter is set before the Start, you won't save any memory by truncating the Start. Likewise, you won't save memory by truncating the End if the Alt is set after the End. You won't save memory by truncating a sample until you save the sample and exit the Sample Editor.

You can also use the Truncate DSP function to automatically truncate your samples at a specified noise floor. As with the TRIM page, you'll recover memory after saving the sample and exiting the Sample Editor.

## LOOP

The LOOP page features the same four parameters as the TRIM page, but the waveform display is quite different. The best way to understand what you see on the LOOP page is to switch back and forth between the TRIM and LOOP pages and study the waveform displays.



On the TRIM page you see the entire waveform—or as much of the waveform as your current zoom setting allows. When you move to the LOOP page, you'll notice that the page is split into two sections, left and right, divided by a vertical bar in the center. This bar is thicker than the vertical lines representing the Start, Alt, Loop, and End points, and does not move when you adjust any of these points.



To the left of the dividing bar you see the same segment of the current sample that you see on the TRIM page. The four vertical lines representing the Start (S), Alt (A), Loop (L), and End (E) points are visible. (Remember, you'll see all four vertical lines only if the values for the Start, Alt, Loop, and End parameters are different.) To the right of the dividing bar you see the entire loop segment of the sample.

In the center of the loop segment is a dotted vertical bar that represents the loop transition point—that is, the point at which the sample reaches its End point and loops back to the Loop point. You can visualize the loop segment by starting at the vertical transition point; this is the beginning of the loop, as defined by the setting for the Loop parameter. The waveform progresses to the right, representing the initial portion of the loop segment. The waveform “disappears” off the far right side of the display, and “reappears” at the thick dividing bar at the center of the display. The waveform again progresses to the right, representing the final portion of the loop segment. It reaches the dotted vertical transition line, representing the End point of the sample, where it jumps once again to the loop point and repeats the cycle.

If you select the Loop (L) parameter and change its value, you'll see the segment of the waveform to the right of the transition point shift its position. If you select the End parameter and change its value, you'll see the segment of the waveform to the left of the transition point shift its position.

When you're setting a loop segment for a sample, you'll want to adjust both the Loop and End parameters so the two ends of the waveform meet (or come as close as possible) at the transition point. You'll notice an audible click in your sample loop if the ends of the waveform do not meet at the transition point. The closer you can get the two ends of the waveform, the better the sound quality of your loop will be. With a bit of experimentation, you'll develop the ability to create smooth loop transitions.

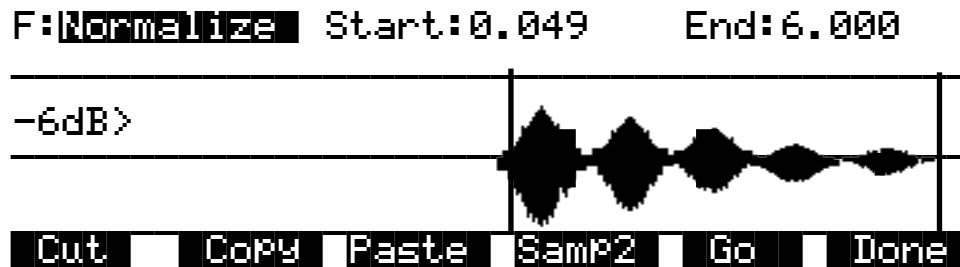
You'll also want to try to set the loop point at a zero-crossing—a point where the sample waveform is neither positive or negative. Pressing the Plus/Minus buttons together will search (from left to right) for the sample's next zero-crossing. You can usually press these buttons several times for any given sample without noticeably affecting the sound of the sample. If you press the Minus button, you'll reverse the direction of the search, and the next time you press the Plus/Minus buttons together, the K2500 will search for the next zero-crossing to the left. Press the Plus button again to search toward the right.

If you adjust the display Gain and Zoom of the sample while on the LOOP page, you'll notice that the Gain affects the waveform on both sides of the loop point, while the Zoom affects only the left side of the page. You can't zoom in on the loop transition point in the right half of the display.

You can also use the crossfade loop (XfadeLoop) DSP function to get a smooth transition between loop points. As with the TRIM page, you'll recover memory after saving the sample and exiting the Sample Editor.

## DSP (RAM samples only)

Select the DSP page with the **DSP** soft button. This gives you access to a long list of non-real time DSP functions, with which you can modify your RAM samples. The first time you select a DSP function, you'll see the Normalize function, shown below. Afterward, the most recently selected DSP function will appear when you select the DSP page.



All of the DSP functions operate on a segment of the current sample that you select before executing the function. In most cases, you'll use the Start and End parameters to define the start position and end position of the segment you want to modify. There are a few exceptions to this rule, which will be explained as applicable.

Please keep in mind that the Start and End parameters on the DSP pages are not the same as the sample Start (S) and End (E) parameters that you set on the TRIM and LOOP pages. When you're working on one of the sample DSP functions, Start and End position refer to the range of the sample that you want to process. Adjusting these parameters does not affect the overall start and end of the sample. It affects only the portion of the sample that you want to process. When you audition the sample by triggering a note, you'll hear only the range of the sample within the Start and End parameters on the current DSP page. To hear the entire sample, press the **Done** soft button to return to the Sample Editor page. Similarly, moving the S and E parameters on the TRIM or LOOP pages have no effect on the Start and End parameters on the DSP pages.

Use the Plus/Minus buttons or the Alpha Wheel to select the starting and ending positions of the selected sample segment (the Start and End parameters). You can audition the selected sample segment by triggering any note within the current key range.

The actual processing of the sample begins when you press the **Go** button. The K2500 will display a row of scrolling dots while it's processing. When it's finished, it will prompt you : "Keep these changes?" Trigger a note to hear the result. Press the **No** soft button to return to the page for the currently selected DSP function. This will undo any changes you've made. Press the **Yes** soft button to make the change. The Save dialog will appear (except for the Truncate function). You can save to a different ID than the one displayed, if you want to preserve the original sample. Pressing the **Cancel** soft button will return you to the current DSP page without saving the sample. (the sample reverts to its original condition.)

Six of the DSP functions ( Mix, Mix Beat, Mix Echo, Insert, Replicate, and Beat Volume Adjust) involve selecting a second sample segment (Sample 2) to be processed with the currently selected sample. In these cases, use the **Samp2** soft button. When you press it, another page appears, enabling you to select a second RAM sample using the Alpha Wheel or the alphanumeric buttonpad. You can audition Sample 2 by triggering a note.

Once you've selected Sample 2, use the Start and End parameters to define the start and end positions of the segment of Sample 2 that you want to process. While you're on the Samp2 page, you'll hear the currently selected Samp2 when you trigger a note. When these positions are defined, press the **OK** soft button to return to the DSP page where you can continue your editing.

Once you've selected a sample on the Samp2 page, it remains selected until you return to the Samp2 page and select another sample. You can always use the Samp2 page to audition a second sample.

You can use the **Copy** soft button on the Samp2 page to copy the selected segment to a buffer, then paste the segment into Sample 1 when you return to the DSP page.

## The Soft Buttons on the DSP Page

### Cut

This button will cut (remove) the currently selected sample segment from the currently selected sample, and store it in a buffer. This is equivalent to cutting a section out of a piece of audio tape and splicing the remaining ends together. The cut segment is then available for pasting elsewhere in the sample. The segment you cut will remain in memory until you replace it with another cut or copy command, or until the K2500 is shut off. If you accidentally cut a sample segment, you can restore it by immediately pressing the **Paste** soft button.

Compare this function to the Delete DSP function, which removes the selected sample segment, but doesn't store it in a buffer. If you delete a sample segment, it's *gone*.

### Copy

Use this button to store the selected sample segment in a buffer without altering the current sample. The copied segment will remain in memory until you replace it with another cut or copy command, or until the K2500 is shut off.

### Paste

This button has an effect only after you've cut or copied a sample segment using the Cut or **Copy** soft buttons. The **Paste** soft button inserts the contents of the Cut/Copy buffer after the start position of the currently selected sample segment. This is like splicing a section of audio tape into another section; it extends the length of the sample. You can undo a paste by pressing the **Cut** soft button immediately after pasting.

When you cut or copy a sample, it's stored in a buffer. Whatever is stored in the buffer will remain available for repeated paste commands until replaced by another cut or copy command, or until the K2500 is shut off or reset (soft or hard). The buffer is also cleared if you execute a master delete, and select "Delete everything."

### Samp2

The **Samp2** button, as described earlier in the DSP page section, enables you to select a segment of a second RAM sample to be processed with the currently selected sample.

### Go

Press the **Go** soft button when you want to execute the currently selected DSP function on the currently selected sample. When processing is completed, the K2500 will prompt you to keep the changes. You can audition your changes and decide whether to keep them, then press Yes or No.

### Done

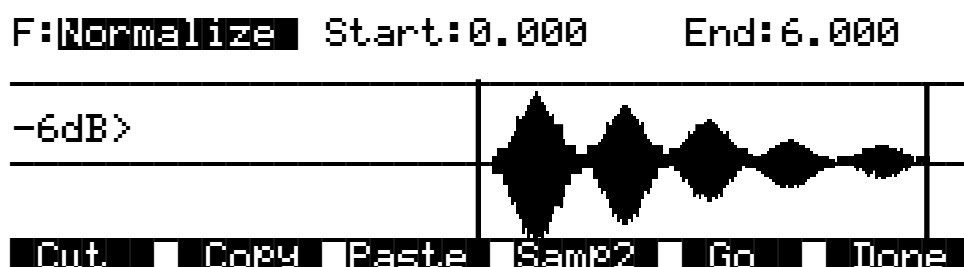
Press the **Done** soft button to return to the previously selected Sample Editor page when you're finished with the DSP functions.

## DSP Functions

Once you've entered the Sample Editor, press the **DSP** soft button to gain access to the DSP functions. The DSP function parameter will be highlighted, allowing you to scroll through the list of functions with the Alpha Wheel or Plus/Minus buttons.

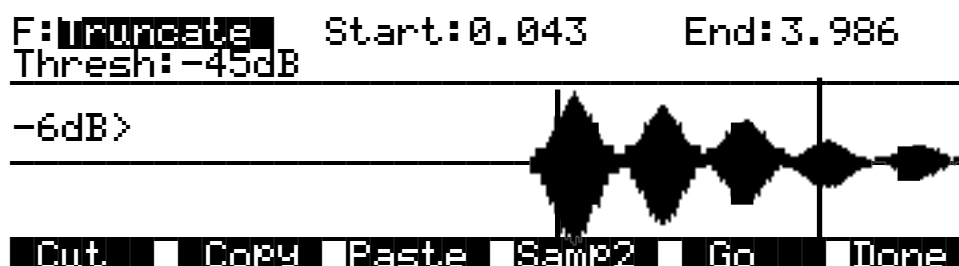
Remember that the DSP functions operate only on RAM samples. You'll notice that if you're editing a ROM sample, there is no **DSP** soft button; instead, there's an extra **Link** soft button for your convenience.

### 1 Normalize



With the Normalize function, you can rescale the amplitude of the selected sample segment to optimize its level relative to other samples. The Normalize function will uniformly boost the amplitude of the current sample range as high as possible without clipping, stopping just before the loudest element of the sample would be clipped. You might want to use the Volume Adjust function to boost the current segment manually, but the Normalize function does it automatically, and prevents you from boosting the amplitude too much.

### 2 Truncate



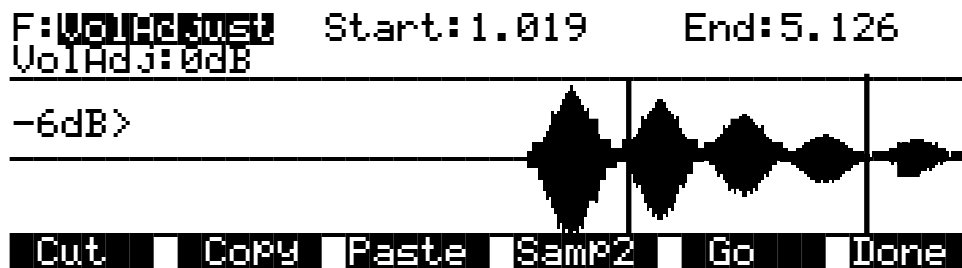
The Truncate function will automatically reset the Start (S) and End (E) points of the sample. This can be quicker than trimming the sample manually on the TRIM page.

Use the Start and End parameters to select a specific sample range. Set the Thresh parameter from -96 to 0dB to set the noise floor. When you press the **Go** soft button, the K2500 will search inward from the start and end points you set, until it finds the first sample that exceeds the noise floor. Everything outside the range will be left out. You'll want to experiment with different thresholds to find the noise floor that suits each sample you truncate. If the new End point is inside the current loop point, the loop will be disabled.

When you press the **Go** soft button, you'll be prompted "Keep this change?" Unlike the other DSP functions, answering Yes does not bring up the Save dialog. It registers the new Start and

End positions as the Start (S) and End (E) points (as set on the TRIM or LOOP page). In fact, you can go to either of these pages to readjust the S and E points if you want to include more of the sample. To save the new S and E points, press the **Done** soft button (if necessary) to return to the Sample mode page, then press either the **Save** soft button, or EXIT.

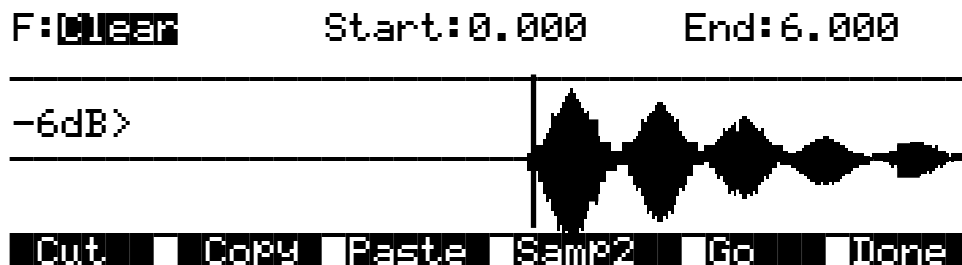
### 3 Volume Adjust



Use this function for a uniform cut or boost in the amplitude of the selected sample segment. This function will clip samples if you adjust the volume too high. This won't hurt the K2500, and you may find it useful in some applications. In any case, you'll need to choose your start and end points carefully, if you want to avoid abrupt changes in volume.

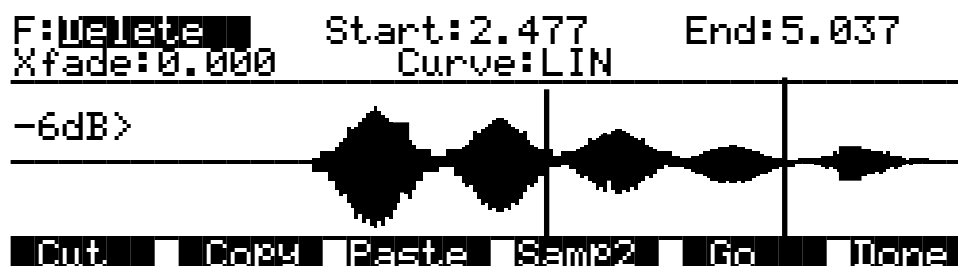
When you've selected the range to be adjusted, select the VolAdjust parameter with the cursor buttons, and use the Alpha Wheel to adjust the volume of the selected range. You can cut/boost the volume from -96 to 96 dB.

### 4 Clear



The result of this function is like erasing a section of recording tape. Use it to create sections of silence without changing the overall length of the sample. If you want to completely remove a segment and shorten the sample, you can do it with the Delete function.

## 5 Delete



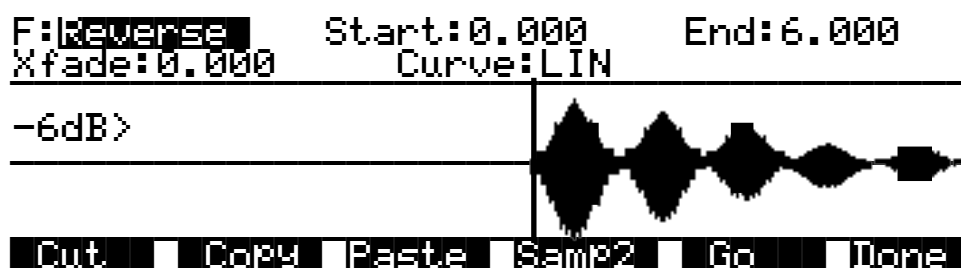
Unlike the Clear function, this will erase the samples within the selected range and shorten the sample, like cutting a section out of a tape and splicing the ends. If you want to silence a segment of the sample without shortening it, use the Clear function.

The Crossfade (Xfade) parameter enables you to smooth the transition from the deleted sample segment to the remaining sample segments, and can create overdub effects. The value of the Xfade parameter defines the amount of time the sample will take to fade to silence at the beginning of the deleted segment, and to ramp back up to the volume of the remaining portion of the sample.

The Curve parameter lets you choose from a variety of crossfade curves that affect the nature of the crossfade. A value of LIN gives a straight linear curve that fades uniformly from start to finish. A value of EXP sets an exponential curve that starts gradually and steepens toward the finish. A value of COS sets a Cosine curve. A value of EQL applies an equal crossfade that is the same at both ends of the deleted segment. A value of MIX assigns a curve that will start the crossfade before the start point of the deleted segment, and will end after the end point of the deleted segment. This differs from the other curves, which apply crossfade only to the deleted segment.

Crossfading a sample will shorten it by half the time you specify. The greatest length of crossfade you can get is the entire length of the shorter crossfaded segment (even though you can enter greater amounts with the Alpha Wheel or alphanumeric buttonpad). A crossfade of this length would crossfade the two samples over the entire length of the shorter segment.

## 6 Reverse



With this function you can reverse the order of the individual samples between the start and end positions you set. The Xfade parameter lets you apply a crossfade to the start and end of the reversed segment. The Curve parameter lets you select a crossfade curve. The available

values are LIN, EXP, COS, EQL, and MIX. These curves are described at the end of the section on DSP functions.

Like the crossfade parameter in the Delete function, this crossfade will also shorten the sample. The maximum crossfade length is half the length of the reversed segment.

## 7 Invert



Use this function to invert the waveform of the selected sample range. This reverses the phase of the sample. You won't hear any difference unless it is played in reference with the original sample.

The results will vary depending on the type of sample. For an interesting phase effect, make a copy of a sample, then invert the copy, then assign it to the keymap of a second layer in a program that uses the original sample in the keymap of the first layer (or set them as the Left and Right samples in a stereo keymap). Add a delay to one of the layers or keymaps to create the phase effect.

## 8 Insert Zero



This function will insert a period of silence of any length into the selected sample range. This function is equivalent to splicing a section of blank tape into an existing segment of recorded tape. It's useful for creating a delay on the start of a sample, or creating a silent spot in the middle or at the end of the sample. Adjust the Start parameter to determine the point at which the period of silence will begin. Adjust the Length parameter to specify how long the silence will be. The End parameter has no effect for this function.

## 9 Mix



With this function you select a segment from Sample 2, and merge it with the selected segment from Sample 1, beginning at the point you set with the Start parameter. This is equivalent to mixing two audio signals through a mixing board. If the Sample 2 segment is longer than the segment from the Start (S) to the End (E) of Sample 1, the resulting sample will be longer than the original Sample 1.

Use the In and Out parameters to specify the length of time it takes Sample 2 to reach full amplitude and to fade to silence. The Curve parameter selects the curve of the fades. The available values are LIN, EXP, COS, EQL, and MIX. These curves are described at the end of the section on DSP functions.

The Volume Adjust parameter will cut or boost the amplitude of Sample 2 from -96 to 96 dB before merging.

If the sample rate of Sample 2 is different from that of Sample 1, the K2500 will alert you that the sample rates differ. If you mix samples with different rates, you'll hear a pitch shift in the mixed sample. If you don't want this pitch shift, use the Resample function to match the sample rates of the two samples before mixing.

### **Mixing Samples, Step-by-step**

First, use the Start parameter of Sample 1 to set the point at which the mix will begin. Then press the **Samp2** soft button, and a page will appear enabling you to view and audition the list of RAM samples. Select a sample with the Sample parameter, then use the cursor keys to select the Start parameter. Turn the Alpha Wheel to set the beginning of the segment to be mixed in. Repeat this process for the end position of the sample. You can audition the sample as you're setting the range. Once the start and end points are set, press OK to return to the DSP page. Then press Go to initiate the mix.

## 10 Insert





Use this function to insert the selected segment from Sample 2 into Sample 1. This is like splicing a section of tape into an existing tape. This differs from the Mix function, which merges the two samples into one.

Use the Crossfade parameter to control the crossfades at the start and end of the inserted sample. The Curve parameter selects the curve of the crossfade. The available values are LIN, EXP, COS, EQL, and MIX. These curves are described at the end of the section on DSP functions.

The Volume Adjust parameter will cut or boost the amplitude of Sample 2 from -96 to 96 dB before merging.

If the sample rate of Sample 2 is different from that of Sample 1, the K2500 will alert you that the sample rates differ. If you insert a sample with a different rate, you'll hear a pitch shift in the inserted sample. If you don't want this pitch shift, use the Resample function to match the sample rates of the two samples before inserting.

## 11 Volume Ramp



This function lets you apply a ramp to the volume of the selected sample range. The Start Level and End Level parameters let you set the amount of cut (negative value) or boost (positive value) at the start and end points of the segment. The Curve parameter determines the shape of the ramp that scales the amplitude of the sample between the start and end amplitudes. The available values are LIN, EXP, COS, EQL, and MIX. These curves are described at the end of the section on DSP functions.

The VolRamp function affects only those samples within the start and end points. The sample will clip if you apply large amounts of volume ramp. If you want to ramp the volume of a sample segment up or down, then keep the volume at that level, use the Crescendo function.

## 12 Crescendo/Decrescendo (Crescendo)



Similar to Volume Ramp, this function applies a curve that scales the amplitude of the selected sample segment. Unlike Volume Ramp, however, you simply select a start and end point, and a single level. The amount of cut or boost starts at 0dB at the start point of the ramp, and reaches the level you specify when it reaches the end point of the ramp. The samples after the end point are scaled to the amplitude level of the end point.

When you set a negative end level (decrescendo), the sample's volume is cut by the amount you specify. If you set a positive end level (crescendo), the sample segment is first cut by the amount you specify, then is boosted back to its original level. This enables you to add large crescendos without clipping the sample. You may want to adjust the gain of the layer using the sample to match its level with other sounds.

Use the Curve parameter to select the shape of the curve within the selected segment. The available values are LIN, EXP, COS, EQL, and MIX. These curves are described at the end of the section on DSP functions.

### 13 Resample



Use this function to change the sample rate of the samples in the selected segment. This is convenient for converting samples to new rates matching those of other samples to be mixed with or inserted into. It's also useful for saving memory and altering the timbre of a sample.

If you include the entire sample in the segment, the new rate will be applied to the entire sample, and will be saved with the sample. If you select a shorter segment, only that segment will be modified, and it will sound pitch shifted relative to the remainder of the sample. To resample so that you'll hear a higher pitch for the selected segment, select a lower sample rate. For a lower pitch at the selected segment, choose a higher sample rate. This is because the K2500 applies the same playback rate to the entire sample, and doesn't compensate for the differing sample rates of the sample segments.

If the loop points of a looped sample are included in the segment to be converted, the K2500 will ask you if you want to adjust the rate slightly to optimize the loop. Press the **No** soft button if you don't want the rate adjusted.

You can use the Quick parameter to select from two resampling routines. Use Quick 0 to get an idea of the sound, then use Quick 1 for your final take.

## 14 Time Warp



With this function you can change the length of the selected sample segment without affecting the pitch. This function applies sophisticated routines that lengthen or shorten the selected sample segment to play it back over a different time period, modifying the playback rate so the pitch remains unchanged.

The Start and End parameters define the segment to be processed. Use the New Length (NewLen) parameter to specify how long you want the resulting sample to be. While the function is in progress, the display will indicate the percentage of individual sample segments that have been processed.

The Quick parameter lets you select one of three warping routines. Use Quick 0 to audition your new sample, then use Quick 1 or 2 for your final take. Quick 2 takes more time to process, but gives you a better final result.

This function is extremely useful for fine adjustments in the length of a sample. You can also apply greater amounts of warp for a wide range of effects. Experimentation will give you an idea of the amount of alteration you want to apply.

## 15 Pitch Shift

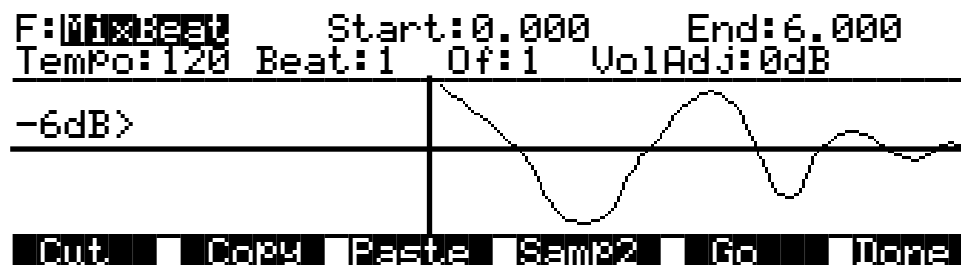


The PitchShift function is the counterpart of the TimeWarp function; it shifts the pitch of the selected sample segment without changing the playback time—very useful for tuning samples when the playback time is crucial. Use the Start and End parameters to define the segment to be shifted. The Shift parameter determines the amount of pitch shifting, up to  $\pm 30000$  cents.

Like the crossfade parameter in the Delete function, this crossfade will also shorten the sample. The maximum crossfade length is half the length of the reversed segment.

The Quick parameter lets you select one of three shift routines. Use Quick 0 to audition your sample, then use Quick 1 or 2 for the final take. Quick 2 takes longer to process, but gives you better results.

## 16 Mix Beat



With this function you can mix the selected range of Sample 2 into Sample 1 at regular intervals. The most natural application of this function is to add percussion samples, but it can be used to mix any sample at regular programmable intervals (beats).

The Tempo parameter sets the interval between repetitions in beats per minute. The Of parameter establishes the number of beats per measure, and the Beat parameter sets which beat gets mixed. For example, if you set the Tempo parameter to 120, you'll have two beats per second. If you set the Beat parameter to 1, and the Of parameter to 1, Sample 2 will be mixed in twice per second, for the duration of the selected range of Sample 1. The first mix of Sample 2 occurs at the Start (S) point of Sample 1. If you change the Of parameter to 4, Sample 2 will be mixed into Sample 1 on the downbeat of every measure of 4.

The length of the ranges you set for Sample 1 and Sample 2 affects the results of the mix. In the above example, if Sample 1 is two seconds in length, and the mix of Sample 2 is on Beat 1 of 1, you'll mix four segments of Sample 2, at half-second intervals. On beat 1 of 4, you'd hear just one mix of Sample 2, right at the top.

To set up a Mix Beat, first select the desired segment of Sample 1 using the Start and End parameters. Then press the **Samp2** soft button to select the sample to be mixed in, and the selected range of that sample. You can use the Incr parameter to shift the starting position of Sample 2. If you set a positive number, the mix range of Sample 2 will start later each time it's mixed. A negative value will mix sample ranges that start progressively earlier.

To optimize processing time, keep the range of Sample 2 shorter than the interval between mixed-in segments. In the example above, if the mix were on beat 1 of 1, you'd want to keep Sample 2 at a range of a half second or less. If it were on beat 1 of 4, the range of Sample 2 could be as much as two seconds.

Next, press the **OK** soft button to return to the Mix Beat page, then use the Tempo parameter to select the rate at which Sample 2 will be mixed in. You can choose a tempo from 1 to 9999 beats per minute. Try setting the tempo at 9000 or more, and mix a very small range of Sample 2 into Sample 1 to create a wide variety of periodic waveforms.

Now set the values of the Beat and Of parameters to determine how the mixed sample will repeat. The Beat parameter determines the beat (s) on which Sample 2 will be mixed—from 1 to 9999. The Of parameter determines the measure length—also from 1 to 9999.

Finally, use the VolAdjust parameter to set the volume of the mixed sample segment—from -96 to 96 dB.

If the Of parameter is set to a value of 0, the Sample 2 segment will be mixed in on every beat, regardless of the setting for the Beat parameter. If the Beat parameter is set to a negative value, the segment of Sample 2 that's mixed in will be moved forward in time by the length of one beat each time it's mixed in; that is, you'll hear a later portion of the sample. Another way to accomplish this is to use the Increment (Incr) parameter on the Samp2 page. Set it to a positive value to use a later portion of Sample 2 with each repetition, or a negative value to use an earlier portion. The Beat parameter must be set to a value of 0 or higher for this to work.

For example, suppose you've chosen a six-second sample as Sample 1, and you use the entire sample as the selected segment. You also select a half-second segment of Sample 2 to be mixed in. If you choose a Tempo value of 120 beats per minute (2 beats per second), there will be 12 beats within your six-second Sample 1 segment. If you set the Beat parameter to 1 and the Of parameter to 4, then Sample 2 will play on the first, fifth, and ninth beat of your six-second Sample 1 segment. The result will look something like the diagram below.



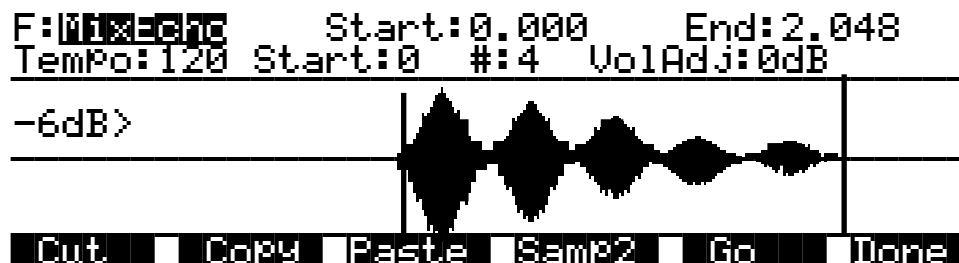
If you change Beat to 3, Sample 2 will play on beats 3, 7, and 11. If you set Beat to 1 and Of to 6, Sample 2 will play on beats 1 and 7. If you set Beat to -1, Sample 2 will still play on beats 1 and 7, but the sound of the sample will change each time, as the start point of the Sample 2 segment moves forward (later) by one beat each time it is mixed in.

## 17 Replicate



With the exception of two differences, the replicate function is similar to the Mix Beat function. The primary difference is that the replicate function uses the selected Sample 2 segment to overwrite (replace) the selected Sample 1 segment, instead of merging the two segments like the Mix Beat function. This makes it run faster. The other difference is that there is a crossfade parameter instead of a Volume Adjust parameter. The crossfade parameter lets you smooth the transition points from Sample 1 to Sample 2.

## 18 Mix Echo



This function operates much like Mix Beat, but instead of the Beat and Of parameters, you have Start and “#” parameters. The Start parameter sets the beat at which the selected Sample 2 segment begins being mixed with the selected Sample 1 segment. Sample 2 is repeated on every beat after this starting point. The “#” parameter determines how many times the Sample 2 segment is repeated.

The Volume Adjust parameter will affect the volume of each repetition of Sample 2. It sets the relative level of the last beat, and boosts or cuts each repetition by a proportionate amount from the starting level. This is a linear adjustment, measured in decibels, evenly spaced over the total number of mixed-in beats.

If you set the “#” parameter to a positive value, the adjustment in volume begins on the second mixed-in beat (the first beat plays at the initial level). If you set a negative value, the first mix of Sample 2 will be cut or boosted by the Vol adjust amount.

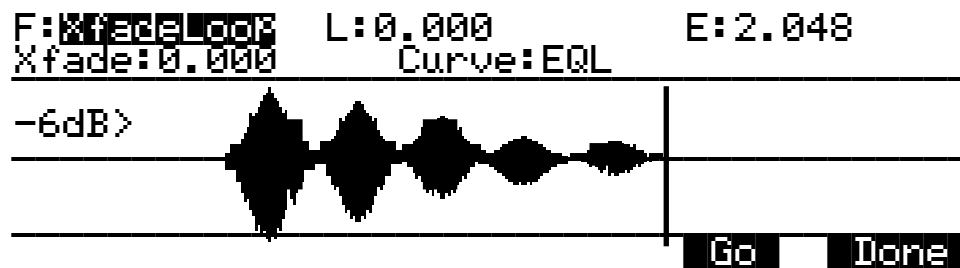
## 19 Beat Volume Adjust



This function works much like Mix Beat. The difference is that you’re not mixing in any samples, you’re just volume adjusting Sample 1 at regular intervals. Each beat is adjusted in volume by the amount specified for the VolAdj parameter. This is useful for mixing part of a sample within itself.

Set the Start and End positions of Sample 2 to define the length of time over which the volume will be adjusted. For example, if you had a six-second sample, and you set the Tempo to 120, the Beat parameter to 1, and the Of parameter to 1, you’d have twelve beats defined. If you then set the range of Sample 2 at a half-second, and set the VolAdj parameter to -3 dB, then you’ll insert 12 volume adjustments into Sample 1, each lasting a half second, and bringing the level down 3 dB.

## 20 Crossfade Loop (XfadeLoop)

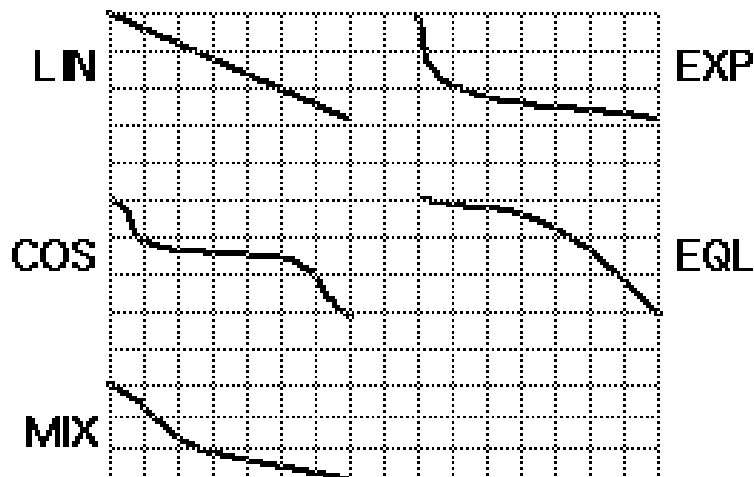


The Crossfade Loop function lets you create smoother loops by crossfading the beginning segment of the loop with a segment of equal length at the end of the loop. These segments can be defined by the Loop and End parameters as set on the TRIM or LOOP page for the current sample, or with the Loop and End points on the XfadeLoop page. Changing the Loop and End parameters on the XfadeLoop page will change them on the TRIM and LOOP pages, and vice versa. Using this function is equivalent to setting the loop on the LOOP page, but with the added feature of a crossfade at the loop transition point.

The Xfade parameter determines the length of the crossfade, while the Curve parameter sets the shape of the crossfade curve. The available values are LIN, EXP, COS, EQL, and MIX. These curves are described below.

### Crossfade and Volume Adjust Curves

There are five curves that can be applied to a number of DSP functions: LIN, EXP, COS, EQL, and MIX. The LIN curve is a straight linear curve, which will create an even cut or boost. The EXP curve is exponential, that is, gradual at one end and steep at the other. The COS curve is a segment of a cosine curve, which is relatively flat at both ends and steeper in the middle. The EQL curve steepens at an even rate, approximating an equal power fade curve. The MIX curve is a gradual curve that approximates manually dropping or raising the faders on a mix board. The diagram below shows each of the curves as it would be applied to a cut in amplitude.



## Reading Samples

The K2500 will load numerous samples from Akai, Roland, and Ensoniq EPS and EPS-16 Plus SCSI drives and floppies (including ASR-10 “Ensoniq” format), using Version 2’s enhanced Disk mode operations. The displays you see will vary depending on the samples you’re loading, but several features are the same. We’ll describe the similarities first, then elaborate on the differences.

First, enter Disk mode by pressing the Disk mode button. Select the disk to be loaded from, as described in Chapter 13. The K2500 automatically recognizes the type of disk when you select it.

Press the **Load** soft button, and you’ll see a page prompting you to select something to load (we’ll call them objects, since different manufacturers give them different names). The top line of the display will tell you the number of objects available of the currently selected type, as well as the index number of the currently selected object. You can select any object in the list by typing its index number on the alphanumeric buttonpad and pressing Enter. The next step is to use the soft buttons to select the type of object to be loaded.

Once you’ve selected the type of object to load, press the **OK** soft button, and the bank dialog will appear, enabling you to select the bank into which the object(s) will be loaded. When you’ve selected a bank, press the **OK** soft button, and the loading process will begin. At the center of the display you’ll see the object currently being loaded. The top line of the display will fill with asterisks to indicate the status of the current object. The bottom line will tell you the total number of kilobytes to be loaded.

The K2500 will create layers as necessary when you load objects. These layers have the same settings as Layer 1 of Program 199.

When the load is complete, the Disk mode page will reappear. You can now proceed with another load, or go to any other mode. If you exit Disk mode, the K2500 will remember the file that you selected most recently. When you return to Disk mode, this file will be selected.

Once you’ve loaded a sample or program file, you can save it as a Kurzweil object. You’ll find it it can be loaded and backed up much faster as a Kurzweil object than in its original format.

### Akai

The first page to appear is the page for loading files. The soft buttons name the operations: **HDrive**, **Volume**, and **File** on the left, and **OK** and **Cancel** on the right.

The hierarchy of objects is shown by the three soft buttons on the left. The display prompts you: “File to load:” The **HDrive** button selects the partition on the currently selected disk. The **Volume** button selects volumes within the currently selected partition. The **File** button selects an individual sample file from within a volume. The **OK** button, toward the right, executes the displayed function: partition selection, or loading a volume or file. The **Cancel** button returns you to the Disk mode page.

When you press the **HDrive** button, the center of the display’s top line shows the currently selected volume in the currently selected partition. The prompt at the center of the display will read: “HD Partition.” The list of available partitions will appear following the prompt. They’re usually named A through F. Use the cursor buttons or numeric entry to highlight a different partition. Pressing the **OK** soft button will select the highlighted partition.

Pressing the **Volume** button will change the prompt to “Volume to load:” The list of available volumes in the current partition will appear. The center of the top line will show the current partition. The Layer buttons will scroll through the list of available partitions. Use the cursor buttons or numeric entry to select a different volume. Pressing the **OK** button will load the entire highlighted volume, unless the volume is larger than your available sample RAM, in which case, the K2500 will load as many files as will fit.

The Bank dialog will appear, enabling you to select the bank that will receive the volume. Press **OK** again, and you’ll be prompted to press either the **Progs** soft button, which will load program information in addition to the samples, or the **Samps** soft button, which will load only



the sample information. Programs are identified by the suffix “.p,” and are stored in program RAM. Samples have the suffix “.s,” and are stored in sample RAM. You can press Cancel to return to the Disk mode page without loading the volume.

If you load sample objects, you’ll see the following prompt: “Create preview program/keymap?” If you answer **Yes**, the K2500 will load the samples into a program that it creates based on Layer 1 of Program 199. Loading program objects will load multi-layer samples and keymaps, and sometimes velocity switches. The K2500 will create layers as necessary when you load program objects. These layers have the same settings as Layer 1 of Program 199. In some cases, the K2500 will also create stereo keymaps to preserve the separation of stereo samples.

If you press the **File** button, the prompt will change to “File to load:” You can view the list of files with the cursor buttons, or use numeric entry. The top line of the display will show the currently selected volume. Select different volumes with the Layer buttons. The size of the currently selected file, in Kilobytes, is shown just above the soft buttons on the left. Press the **OK** button to load the highlighted file.

Press **OK**, and the Bank dialog will appear. Press **OK** again, and the file will be loaded into the highlighted bank.

Sometimes you may find an Akai floppy disk that your K2500 can’t read. In this case, check the format of the disk. Akai allows you to format a double sided-double density (DD) disk for high density (HD). This is not standard in the computer industry, since DD disks are not verified for use as HD disks. If this is the case, you will need to load the disk into an Akai, then save it out to a properly formatted disk.

## Roland

For Roland disks, the hierarchy is a bit different; the objects that can be loaded are called Volumes, Performances, Patches, and Samples. The page that was selected last time a SCSI load was executed will appear when you initiate the load operation. Following the prompt is the list of available objects, with the size of the object in Kilobytes displayed as well. The top line of the display will show the number of available objects of the selected type.

Use the soft buttons to highlight the object to be loaded. The layer buttons will take you through the current object list in increments of 100. Press **OK** to execute the load. The Bank dialog will appear. Press **OK** again, and the object will be loaded. The display will update you on the progress of the load.

## EPS

For EPS disks, the hierarchy consists of files and directories. Directories can be nested several layers deep. When you press the **Load** soft button, you’ll be prompted to select a file or directory to load from the list of available files and directories. The currently highlighted object will be either a file or a directory. If it’s a file, its name and size will be shown following the prompt. If it’s a directory, its name appears, followed by “(dir)” to indicate its type. The Layer buttons will take you to the first and last files of the currently selected directory. Note that if you’re loading from floppy disks, you won’t see any of the displays or soft buttons relating to directories, since the directory architecture is not supported for floppies.

When a file is highlighted and you press **OK**, the Bank dialog appears; press **OK** again to load the file. When a directory is highlighted and you press **OK**, you enter that directory, and the list of files and subdirectories in that directory appears, each file followed by its size, and each subdirectory, if any, followed by “(dir).” The top item in every list you select is always the parent directory of the files below it. Select the top item in a list to go up one directory level.

Pressing the **Exit** soft button will take you one level back up the hierarchy. Pressing it repeatedly will take you to the root directory—the directory at the top of the hierarchy. The quickest way to the root directory is to press the **Root** soft button. The top line of the display shows you the name of the currently selected directory (or subdirectory). Pressing the **All** soft button will load all files in the current directory (but not any subdirectories). The Bank dialog will appear, and when you press **OK**, you’ll be prompted to press the **Progs** button to load program information in addition to the samples, or the **Samps** button to load only the samples.

## The Keymap Editor

The Keymap Editor lets you customize the K2500's factory preset keymaps and save them to RAM. You can also build your own keymaps from scratch.

Keymaps are an integral part of every layer of a program. Each keymap contains a set of parameters determining which sample(s) the K2500 will play when you trigger a note. Each layer has at least one keymap, but it can have two keymaps when you're working with stereo samples. Each of these stereo keymaps uses two of the 48 available voices.

Each keymap consists of a set of key (note) ranges—C 4 to G 4, for example. The entire span of each keymap is from C 0 to G 10. Each range has a sample root assigned within the range. Each sample root is a distinct ROM or RAM sample. Within each key range, the sample root is transposed up and down to play on each of the range's notes. You can view each range by changing the value of the Key Range parameter on the Keymap Editor page. You can mix samples of different timbres within a single keymap, and even tune individual keys to any pitch by defining key ranges to single notes and assigning samples to each of those notes.

When you trigger a note, the K2500 identifies the key range where the Note On event occurred. It also checks the attack velocity value of the note. It then addresses its memory, and retrieves the sample root that's assigned to that key range and attack velocity value. If the note that's triggered is not the note where the sample root is assigned, the sample is transposed to play at the correct pitch. The K2500 then generates the digital signal that represents the sound of the note. At this point the keymap's job is done, and the signal proceeds through the layer's algorithm and on to the audio outputs.

You can assign as many key ranges to a keymap as you like, even creating a separate range for each note. This would allow you to tune each key independently, to create microtonal tunings. For keymaps that use a single timbre, like the Grand Piano, there's a key range for each sample root stored in memory. For acoustic instrumental sounds, the more key ranges you have for a keymap, the more realistic the sound will be, since there will be less pitch shifting of the sample root within the key range.

Of course, you can assign sample roots with different timbres within the same keymap. Many of the drum kit keymaps in ROM, for example, have about 20 key ranges, with several different timbres assigned as the sample roots. You can also create a keymap with a single key range that spans from C 0 to G 10, if you want to stretch a single sample root from C 0 to G 10. Keep in mind, however, that samples can be transposed upward only to a limit of 96 KHz for the playback rate. For 48K samples, that's an octave of upward transposition. Samples can be transposed downward without limit.

Think of a keymap as if it were a single piece of string, divided into different sections that adjoin one another. Sections can not overlap. If you have one range that goes from C4 to F4 and another that goes from F#4 to C5, then if you change the first range to be C4 to G4, the second one will change to be G#4 to C5.

Also, you can't have "nothing" assigned to a key range. Even if it is Silence (#168), there will always be a sample assigned to every range in the keymap. This is something to watch out for when creating drum programs. For example, let's say you are creating a program with 20 layers. Each layer has its own keymap, which has just one sample assigned to part of the keyboard with the rest of the key range assigned to Silence. Make sure that you limit the note range of each layer using the LoKey and HiKey parameters on the Layer page in the Program Editor. If each layer covers the entire range, then each note you played would trigger 20 voices (one for each layer). You would only hear one drum per note because all the other layers are triggering "Silence". Because of the voice stealing algorithms in the K2500, the voices would almost immediately become available again, since they have no amplitude. But for one brief instant, the voice would be triggered, which could cause other voices to be cut off.

You can also create multi-velocity keymaps—that is, keymaps that will play different timbres depending on the attack velocities of your Note On events. The program Dyno E Piano, for example, uses a keymap with two velocity ranges. Each key range in a multi-velocity keymap contains two or three distinct sample roots that the K2500 chooses between, according to the attack velocity of the note. When in the Keymap Editor, you can select the different velocity ranges within a keymap using the CHAN/BANK buttons. To create your own multi-velocity keymaps, start in Program mode by selecting a program that contains a multi-velocity keymap. Most of these programs are identified by the word “Dual” in their names. Then select the KEYMAP page, and press EDIT to enter the Keymap Editor. From there you can change the sample root assigned to each velocity level in each key range. The description of the Velocity Crossover parameter on page 15-40 has more information.

The Keymap Editor is nested within the Program Editor. The first step in using the Keymap Editor is to select the keymap you want to edit. This is done on the KEYMAP page in the Program Editor, using the Keymap parameter. Once you’ve done this, just press the EDIT button, and you’ll enter the Keymap Editor. If you want to edit a different keymap, return to the KEYMAP page in the Program Editor and select the desired keymap. If you want to build a keymap from scratch, start with the keymap 168 Silence (see “Building a Keymap” on page 15-41). This keymap template contains one key range from C 0 to G 10, and is a convenient starting point for adding key ranges and assigning sample roots. The Keymap Editor page looks like this:

```

EditKeymap      <>VelocityRange:PPP-fff
MasterXPose    :MSI
Key Range      :C 0-A 1   Lo:C 0   Hi:A 1
Sample         :1 Grand Piano-G#1
Coarse Tune    :0ST
Fine Tune      :0ct
VolumeAdjust   :0dB      VelCrossover:None
Name          Save      Delete  Dump    NewRng Assign

```

PARAMETER	RANGE OF VALUES
MASTER TRANSPOSE	-128 ST -127 ST
KEY RANGE	Variable
LOW KEY	C 0 to G 10
HIGH KEY	C 0 to G 10
SAMPLE	Sample Root list
COARSE TUNE	-120 to 60 semitones
FINE TUNE	-49 to 50 cents
VOLUME ADJUST	± 48 dB
VELOCITY CROSSOVER	None, ppp to fff

The top line of this page tells you which velocity range you’re currently looking at. If the current keymap is a multi-velocity keymap, the CHAN/BANK buttons let you select between the velocity ranges. The velocity range is set with the Velocity Crossover parameter(s), described on page 15-40.

## The Soft Buttons in the Keymap Editor

The first four soft buttons execute the basic library functions, enabling you to name, save, or delete the current keymap, or dump it via a MIDI SysEx message.

### New Range (NewRng)

The **NewRng** button lets you define a range to edit, whether it's to assign a different sample, or to adjust the pitch or volume. Just press **NewRng**, then trigger the note you want as the low note, then the high note. The K2500 will prompt you for each note. When you trigger the high note, you'll return to the Keymap Editor page, and the edit range you defined will be selected. The next change you make will affect only that edit range.

There's more than one way to use this function. If you set an edit range that's completely within an existing key range, you can modify the edit range without affecting the rest of the key range or the adjacent key ranges. If you set an edit range that overlaps part or all of another key range, the sample assigned to the lower key range will be applied to the entire edit range. This is an easy way to define a new key range that replaces one or more existing key ranges.

### Assign

The **Assign** soft button lets you select a sample, then specify the key range to which it's assigned. This enables you to insert a new key range within the current keymap. When you press the **Assign** soft button, a dialog appears that prompts you to select a keymap from the Keymap list. Scroll through the list, then press the **OK** soft button. You'll then be prompted to define the new key range by triggering the notes you want to be the lowest and highest notes of the range. (Press the **Cancel** soft button if you change your mind.) When you trigger the low and high notes, the new key range is inserted. If the new key range partially overlaps an adjacent key range, the existing key range will be adjusted to accommodate the new range. If the new key range completely overlaps an existing key range, the original key range will be replaced.

## The Keymap Editor Parameters

### Master Transpose

This parameter does not really pertain to the keymap itself. Instead it is identical to the Transpose parameter found on the MIDI Transmit page. If you change the value here, the same value will be reflected on the MIDI Transmit page, and vice versa. It transposes the entire instrument globally. The reason it is placed on this page is that it will allow you to assign samples across the entire keyboard easily, when you are using a keyboard which has fewer than 88 notes.

As with the Transpose parameter on the MIDI Transmit page, if you have the rack, this parameter will not transpose without using the Local Keyboard Channel parameter (refer to page 10-6 for more on the Local Keyboard Channel).

The transposition amounts from this page and the Master page are cumulative. For example, if you set this parameter to a value of +1ST and then set the Transpose parameter on the Master page to -1ST, the two will cancel each other out, and the K2500 will actually be back at 0ST.

### Key Range

This parameter shows you which key range you're currently viewing or editing. Changing the value of this parameter selects a different key range, and shows you the sample assignment of that key range, as well as other parameters related to that key range. Use this parameter to move from one key range to another within the keymap, and assign different samples without adding new key ranges, as well as editing the transposition, tuning, volume and velocity crossover (if any) of the sample root assigned to the key range.

## Low Key (Lo), High Key (Hi)

With these parameters you can use any of the data entry methods to change the low and high notes of the current range. These parameters let you extend or shorten the width of a key range. You can extend a key range to the full capacity of the K2500 (C 0 to G 10).

The setting for the low key cannot be higher than the setting for the high key. Similarly, the setting for the high key cannot be lower than the setting for the low key.

## Sample

This is where you assign a sample root to the current key range. For RAM samples, each sample will have its own ID number. For ROM samples, it's a little different. Because of the large number of ROM samples, we did not want to use up too many of the available IDs. So we have organized them as follows. Each sample's name consists of three parts—for example, "1 Grand Piano G#1." The first numeral indicates what we call the sample block, which represents groupings of similar sample roots in memory. Next comes the name of the sample, which describes the sample's timbre. The final part of the sample's name refers to the pitch at which it was originally sampled. For many timbres, multiple samples are made at various pitches. As you scroll through the Sample list, you'll see only the pitch of the sample change until you reach the next sample block.

Highlight the Sample parameter, hold the ENTER key, then play a note on your controller to see the complete name of the sample played by that note. For example, when you play middle C on a piano program you'll see a name such as "1 Grand Piano C4". Move down the keyboard an octave and the sample will be "1 Grand Piano G#2".

## Coarse Tune

Coarse Tune allows you to transpose a sample for a given range. This is extremely useful when you have set the Root key of the sample for one note but want to assign the sample to a different part of the keyboard and still be able to play it without transposition. For example, if you originally set the Root key at C4 but want the sample assigned to C3, you would set Coarse Tune to 12ST, transposing it up one octave. Now the original pitch will play at C3, one octave down. If you examine the drum and percussion kit keymaps in ROM, you will see that we have done this. Most of our ROM drum samples have the Root key set at C4.

## Fine Tune

This gives you further pitch control. Once the sample's pitch is close to the desired level, use the Fine tune to sharpen or flatten it as much as a half-semitone.

## Volume Adjust

Here you can adjust the volume of the notes in the current key range. This enables you to make each key range play at the same volume even if the samples in the various ranges were recorded at different volumes.

#### Velocity Crossover (VelCrossover)

This parameter applies only when the keymap assigned to the currently selected program is a multi-velocity keymap. The name of the keymap usually indicates whether it's of the multi-velocity variety (2 *Dual Elec Piano*, for example). Multi-velocity keymaps have a predetermined number of velocity levels, each of which can be assigned a different sample.

The K2500 supports keymaps with one, two, or three velocity levels. You can't add velocity levels to existing keymaps; if you want to create your own multi-velocity keymaps, select an existing multi-velocity keymap in the Program Editor before entering the Keymap Editor. Then you can select the different velocity levels with the CHAN/BANK buttons, and assign samples to the different levels. The currently selected velocity range is shown in the top line of the display. Keymaps 2, 10, 54, and 178 are dual-velocity keymaps. Keymaps 164–166 are triple-velocity keymaps.

When the current keymap is a single-velocity keymap, the VelCrossover parameter does not appear on the Keymap Editor page. When the current keymap is dual-velocity, the value for the Crossover parameter will be one of the eight dynamic markings from ppp to fff. The K2500 translates each of your Note Ons into one of these dynamic values, using the settings for the VelTouch or VelocMap parameters. When this translated value exceeds the setting for the VelCrossover parameter, the K2500 plays the sample assigned to the upper velocity range.

When the currently selected keymap has three velocity ranges, the VelCrossover parameter becomes two parameters: LowCrossover and HiCrossover. The K2500 plays the sample assigned to one of these ranges depending on the translated value of each note's attack velocity.

## Building a Keymap

If you used the Keymap Editor to enter the sampling page, then just press Exit from the Sampling page and you are ready to begin creating a keymap. If you entered the Sampling page from Master Mode, do the following. Start in Program mode, and select Program 199, the Default program. Press the EDIT button, and you'll enter the Program Editor. Press the **KEYMAP** soft button, and the KEYMAP page will appear. The Keymap parameter will be automatically selected. Press 1, 6, 8, ENTER on the alphanumeric pad to assign the keymap "Silence." This isn't absolutely necessary, but it makes it easier to recognize the key ranges that have samples assigned to them when you start assigning samples. You can actually choose any program you want to start with, but by choosing these, you are starting with a "blank slate".

With the Keymap parameter still selected, press the EDIT button, and you'll enter the Keymap Editor. The Key Range parameter will be automatically selected, and you see its values: C 0 to G 10 (the entire MIDI keyboard range). The Sample parameter will have a value of 168 Silence C 4.

Now you're ready to start assigning samples to key ranges within the keymap. We'll assume that you've loaded samples with roots at C 1, C 2, C 3, etc. and that you plan to assign a root to each octave. To begin, press the **Assign** soft button. The display will prompt you to select a sample. Use the Alpha Wheel to scroll to one of your samples, or type its ID on the alphanumeric pad and press ENTER. When you've found the sample you want to use, press the **OK** soft button. The display will say "Strike low key..." Trigger A 0 (MIDI note number 21, the lowest A on a standard 88-note keyboard) from your MIDI controller. The display will change to say "Strike High Key..." Now trigger F 1 (MIDI note number 29) from your MIDI controller. The display will return to the Keymap Editor page. The Key Range parameter will show A 0–F 1, and the Sample parameter will show the sample you selected when you started the range assignment.

One more time...Press the **Assign** soft button. Select another sample root at the prompt, and press the **OK** soft button. Now trigger F# 1 for the Low Key prompt, and F 2 for the High Key prompt. At this point you've defined two key ranges, the first from A 0 to F 1, and the second from F# 1 to F 2. You can repeat the process as many times as you want, creating a new key range each time.

Once you have your samples assigned, you may need to transpose them so that they play back at the correct pitch within the range you have chosen. To do this, highlight the Keyrange parameter, scroll to the range you need, then highlight the Coarse Tune parameter. Adjust Coarse Tune to bring the sample to the proper pitch within that keyrange. Then scroll back up to the Keyrange parameter, select the next range, and continue as needed.

Here's a fairly important point that may or may not affect your keymap construction. Suppose you want to build a keymap that uses the same sample in several adjacent key ranges, and you plan to add a bit of detuning to the samples in each range. You might think that you could build the keymap first, then go into the Sample Editor and tweak the samples when the keymap is finished. Yes, but...

Suppose you used the technique we described above to assign a vocal sample whose root was C 4 to a key range from A 3 to E 4. Then you assigned the same sample to a key range from F 4 to B 4. You might be surprised to find that when you finished the F 4–B 4 key range and the Keymap Editor page reappeared, the current key range would not be F 4 to B 4, but A 3 to B 4! This is because the K2500 automatically merges adjacent key ranges that are identical (this is done to save memory). Therefore, some parameter must be different in each adjacent key range you create if you want to build keymaps using the technique we just described. So if you want to use the same samples in adjacent key ranges with, for example, minor pitch or volume modification, you should make those changes to the current sample on the Keymap Editor page before assigning the next range.

## Using the Analog Inputs to Trigger Samples

The analog sampling inputs double as one- or two-channel trigger inputs. This allows audio signals from external sources (such as microphones and tape recorders) to trigger internal samples. The following steps explain how to use the trigger inputs.

1. **Connect cables from the outputs of your external audio source to the analog sampling inputs on the K2500.**

For details on which cables to use with the analog sampling inputs, read “Setting Up For Sampling” at the beginning of this chapter.

2. **Choose the internal samples to be triggered.**

Enter Program mode, and select the program containing the sample(s) you want to trigger.

3. **Set the Song mode Click parameters to match the program you chose.**

Enter the Song mode **MISC** page, and set **Click Channel** and **Click Program** to the MIDI channel and program number of the program you just chose. Change **Click Key** to define the keynumber played by the sequencer’s metronome; you always trigger samples assigned to the *first two* keys above the Click Key.

The default Click parameters are Click Channel 16, Click Program “198 Click”, and Click Key C4.

4. **Enter Sample mode, set Src to “External,” and set Mode to “Trigger.”**

5. **Set the trigger threshold.**

Highlight the Sample mode **Thresh** parameter, and adjust the input sensitivity so that your input signal causes smooth triggering. Threshold values approaching -90db let you trigger the sample using a lower input level, while values approaching -6db require a greater input level.

Now analog input signals will trigger samples in the Click Program. The sampler’s analog inputs are stereo, so you can trigger samples on two different keys. The left channel will trigger samples that are one key above the Click Key; the right channel will trigger samples that are two keys above the Click Key.

One great use for the trigger inputs is to replace the kick and snare from a multi-track recording with samples in your K2500. Assuming the kick and snare are recorded on separate tracks, and the Click Key is C4, the recorded drums will trigger samples assigned to C#4 and D4 in the Click Program.

Keep in mind that the Click Key number is separate from the trigger key numbers. This is because you may want to use the Click Program as the metronome sound, and trigger samples other than the Click. To do this, go to Program mode, select the Click Program, and press EDIT. Then press the **KEYMAP** button and press EDIT again. Now assign your samples to the two keys above the Click Key defined on the Song mode MISC page. Before you leave the Program Editor, be sure the KeyTrk parameter on the KEYMAP page is set to 100ct/key.

When you save the edited Click Program, remember to replace it in the same location. If you save to a new location, set the new program number in the Click Program parameter on the Song mode MISC page.



## Chapter 16

# FUNS

The name “FUNs,” although it sounds like a pun, is simply the best abbreviation for “Function.” You’ll definitely have a good time with them, however, if you’re looking for a variety of ways to control your sounds.

We’ve discussed various control sources throughout this manual, from the physical controls like the Mod Wheel to the software control sources like LFOs and attack velocity. You can assign them to affect your sounds in all sorts of ways.

The FUNs take the control sources one level further. By setting up a FUN as a control source, you can mix the signals of two control sources, and perform one of 50 functions on the combined signals. The result of that function becomes the new control source value. Because the FUNs can radically change the combined input values, the FUNs can have a profound effect on your sounds.

You may find that experimenting with the various FUN equations gives you a better idea of their effects than reading the explanations. Although there’s a great deal of mathematics behind the FUNs, the most important consideration is how they affect your sounds. The more you play around with them, the better you’ll understand how powerful they are.

## The Mechanics of Control Sources

We’ll return for a minute to the notion that the K2500 is an integrated system consisting of a MIDI-driven sound engine and a MIDI-driven effects processor. The sound engine responds to MIDI messages received at the MIDI In port and from the front panel, as does the effects processor.

The K2500’s control sources use their own internal signal format for interpreting control messages and communicating them to the sound engine. Every control source sent from your MIDI controller to the K2500’s sound engine is translated to a value in the range from -1 to +1. This consistency enables the sound engine to process control source signals very efficiently. Conversely, the K2500’s internal control source signals are translated to MIDI values before being sent to the MIDI Out port.

A control signal value of 0 represents minimum effect; it’s equivalent to the control source being turned off or disconnected. A control signal value of +1 represents the maximum positive effect of a control source, while a value of -1 represents the maximum negative effect of a control source.

### Unipolar and Bipolar Control Sources

There are two kinds of control source signals: unipolar and bipolar. A unipolar signal has a value between 0 and +1. A bipolar signal has a value between -1 and +1.

A switch pedal is unipolar; its control signal value will never go below 0. Since it’s a switch control, it has only two possible values: 0, which corresponds to off or minimum, and +1, which corresponds to on or maximum. When you depress your MIDI controller’s sustain pedal, for example, it sends a control signal value of +1 to the K2500’s sound engine.

Continuous controls can be unipolar or bipolar. Consider your MIDI controller’s Mod and Pitch Wheels as examples. Normally, the Mod Wheel affects the K2500 as a unipolar control

source; it sends a control signal value that's interpreted as 0 when it's fully down, and values interpreted between 0 and +1 as you push it up. When fully up, it sends a value that's interpreted as +1. It can be used as a bipolar control source by assigning a value of Bi-Mwl to any control source parameter.

The Pitch Wheel is normally bipolar; it sends a control signal value that's interpreted as 0 when it's centered, values interpreted between 0 and -1 as it's pulled downward, and values interpreted between 0 and +1 as it's pushed upward. It can be used as a unipolar control source by assigning a value of AbsPwl to any control source parameter.

The FUNs can act as unipolar or bipolar control sources; it depends on the values of the input signals and the nature of the function you choose. Depending on the function you choose to process the input signals, the output signal value can exceed +1 or -1. Normally the signal merely pins at +1 or -1; that is, it won't go any higher or lower. In some cases, however, the output signal value is wrapped around instead of pinning; we'll mention these cases as we get to them. You can assume that the output signal values of the functions listed below will pin at -1 or +1, unless specified otherwise.

## Programming the FUNs

Start by entering the Program Editor, then use the soft buttons to select the FUN page. Setting up a FUN as a control source is a two-step process: assigning a FUN as the value for one or more control source parameters in the Program Editor, then programming the FUN on the FUN page, by assigning control sources to two inputs—**a** and **b**, and choosing a function (equation) that will process the combined signals from **input a** and **input b**.

```

EditProg:FUN          <>Layer:1/1

FUN1:  Input a: Input b: Function:
FUN1:  OFF      OFF      a+b
FUN2:  OFF      OFF      a-b
FUN3:  OFF      OFF      (a+b)/2
FUN4:  OFF      OFF      a/2+b
<more> LFO      ASR      FUN      UTRIG  <more>

```

There are four FUNs; you can combine and process four different pairs of control source signals. FUNs 1 and 3 are always local, that is, they affect each note in their respective layers independently. FUNs 2 and 4 are local by default, but they can be made global by setting a value of On for the Globals parameter on the COMMON page in the Program Editor. A global FUN affects all notes in its layer equally and simultaneously.

The best way to understand the use of the FUNs is to set up a simple test model, then plug in the different equations and listen to their effects. We'll walk you through the programming of a FUN and assigning it to control pitch. Then you can scroll through the list of equations at your leisure.

Start in Program mode and select program 199. Press EDIT to enter the Program Editor. Select the KEYMAP page, and change the keymap to 152 Dull Sawtooth. Then select the PITCH page, and assign a value of FUN1 for the Src1 parameter (a shortcut is to press 1, 1, 2, ENTER on the alphanumeric pad). Select the Depth parameter and change the value to 1200 cents. Next, select the FUN page, and select the **Input a** parameter for FUN1. Assign a value of MWheel (the quickest way is to hold the K2500's ENTER button and move your MIDI controller's Mod Wheel). Next, select the **Input b** parameter for FUN1, and assign a value of Data. This assumes your MIDI controller either has a data slider, or a programmable control that you set to send Data messages (MIDI 06). If you don't have a data slider or a programmable control, you can set the value of **Input b** to AbsPwl, and use your Pitch Wheel to control **Input b**. If you do this,

you'll need to go to the LAYER page and set the PBMode parameter to a value of Off to keep Pitch Wheel messages from interfering with the test model.

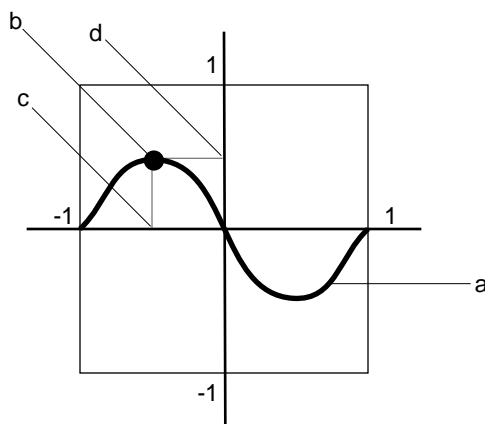
Now select the Function parameter, and scroll through the list of equations. Move your MIDI controller's Mod Wheel and Data slider as you play, and listen to their effects. Actually listening to the various effects while reading the explanations below will help your understanding. In the model we've set up here, **inputs a** and **b** are both unipolar. The effect of each equation will differ depending on the type of controls you assign to the inputs. There are four possible combinations: both inputs unipolar; both inputs bipolar; **input a** unipolar with **input b** bipolar; **input a** bipolar with **input b** unipolar.

## The FUN Equations

In this section we'll describe how each of the FUN equations works. In some cases, a small graph will accompany the explanation. Here's how to interpret the graphs.

Each graph shows a curve illustrating the effect of the equations on the input signals. The horizontal axis represents the possible values of the input to the FUN (the combined control signals of **inputs a** and **b**). The vertical axis represents the possible values of the FUN's output signal. The four elements in the diagram below show you how to read these graphs:

- a the curve representing the effect of the FUN's equation on every possible input value
- b one point on that curve, representing a single input value and the corresponding output value generated by the FUN's equation
- c the input value represented by the point
- d the output value represented by the point



For any point on the equation's curve, you can determine the input value by tracing a line from the point to the horizontal axis. Similarly, you can determine the output value by tracing a line from the point to the vertical axis. For the point shown in the example above, the combined values of the control signals of **inputs a** and **b** equal about -.5, which translates to an output value of +.5. An input value of -1 gives an output value of 0, as do input values of 0 and +1. An input value of +.5 gives an output value of -.5.

## The List of Equations

The first six equations are weighted sums and differences—that is, the signal values of **inputs a** and **b** are added to or subtracted from each other, and are divided in turn by various amounts to alter their effects relative to each other. These equations give you several different types of mixers for combining the signals of the two inputs.

### **a + b**

The values of **inputs a** and **b** are added, creating a simple mixer. For example, you might have LFO1 assigned for the Src2 parameter on a layer's PITCH page, and a FUN assigned for the DptCtl parameter. On the FUN page, if you set **input a** to a value of MWheel, and **input b** to a value of MPress, then this equation will let you modulate the depth of the LFO's pitch modulation with your MIDI controller's Mod Wheel or with mono pressure. You could set a fixed initial depth with the Mod Wheel and alter it further with mono pressure. In this case the output signal would pin at +1 or -1 fairly quickly.

### **a - b**

This operates similarly to the previous equation, but the value of **input b** is subtracted from the value of **input a**. This equation will reverse the normal effect of the control source assigned to **input b**. For example, if **input a** is off, and **input b** is assigned to a unipolar control source like MWheel, then the Mod Wheel will generate a control signal of -1 when fully down, and 0 when fully up.

### **(a + b) / 2**

The values of **inputs a** and **b** are added, and the sum is divided by 2. This gives you the same kind of control as the previous two equations, but the output signal will reach +1 or -1 half as often as with the equation  $a + b$ .

### **a / 2 + b**

The value of **input a** is divided by 2, and the result is added to the value of **input b**. **Input a** has half the effect of **input b**.

### **a / 4 + b / 2**

The value of **input a** is divided by 4, and the value of **input b** is divided by 2. The two results are added to give the output value. **Input a** has half the effect of **input b**, and the total result has half the effect of the previous equation.

### **(a + 2b) / 3**

The value of **input b** is multiplied by 2, and the result is added to the value of **input a**. This sum is then divided by 3. **Input a** has half the effect of **input b**, and the total result has somewhat more effect than the previous equation, but less effect than  $a + b$ .

### **a \* b**

The values of **inputs a** and **b** are multiplied. If you like using Src2 and DptCtl, this equation can be used to create a similar type of control source (it's equivalent to the Src2/DptCtl pair with the MinDpt parameter set to 0).

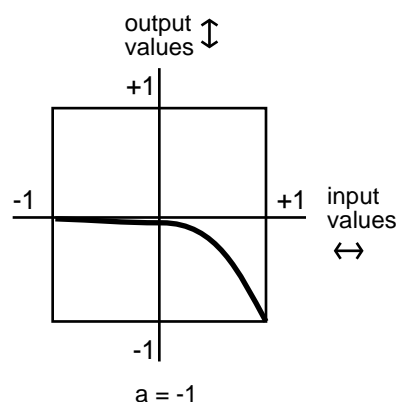
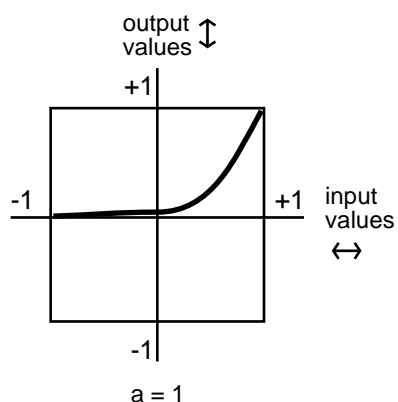
### **-a \* b**

The value of **input a** is multiplied by -1, then multiplied by the value of **input b**. This will reverse the normal effect of the control source assigned to **input a**. This equation also produces an effect like that of Src2 and DptCtl with the MinDpt parameter set to 0.

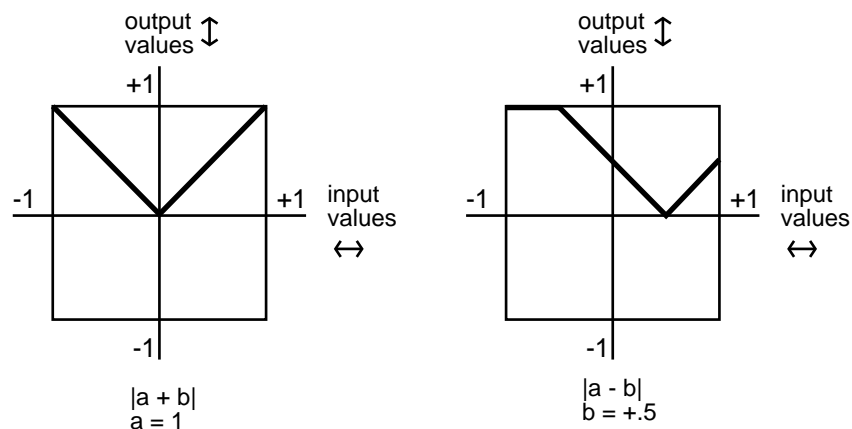
**$a * 10^b$** 

The actual equation is:  $a \times (10^{(2 \times b)} \div 100)$ . This is an exponential curve. 10 is raised to the  $(2 \times b)$  power, then divided by 100. This result is then multiplied by **a**. Another way to express this is as follows: a change of 1 in the value of input **b** results in a hundredfold change in the output value. Here are a few possible output values:

Input values	Output values
<b>a</b> = +1, <b>b</b> = +1	+1
<b>a</b> = +1, <b>b</b> = 0	.01
<b>a</b> = +1, <b>b</b> = -1	.0001
<b>a</b> = 0, <b>b</b> = +1	0
<b>a</b> = 0, <b>b</b> = 0	0
<b>a</b> = 0, <b>b</b> = -1	0
<b>a</b> = -1, <b>b</b> = +1	-1
<b>a</b> = -1, <b>b</b> = 0	-.01
<b>a</b> = -1, <b>b</b> = -1	-.0001

 **$| a + b |$** 

The values of **inputs a** and **b** are added, and the absolute value of the sum is taken. If the sum is negative, it is multiplied by -1. This makes the FUN a unipolar control source. (See the illustration on the following page.)



### $|a - b|$

The value of **input b** is subtracted from the value of **input a**, and the absolute value is taken. If the difference is negative, it is multiplied by -1. This also makes the FUN unipolar.

### $\min(a, b)$

The values of **inputs a** and **b** are compared, and the smaller value becomes the output value. This can be used to limit the value range of a control source. If, for example the value of the control source assigned to **input b** is left at +.5, then when the value of the control source assigned to **input a** is between -1 and +.5, its value will be used. As soon as its value exceeds +.5, the value of **input b** is used.

### $\max(a, b)$

This is the opposite of the previous equation. The values of **inputs a** and **b** are compared, and the larger value becomes the output value.

### Quantize b to a

This turns the control source assigned to **input b** into a stepped control source. Instead of smooth transitions from minimum to maximum, it will jump from minimum to maximum in some number of equal steps. The number of steps is determined by the value of **input a**. The normal realtime application of this is to set a stationary value for input a to set the number of steps in the effect. Then use the control source assigned to **input b** as a realtime control to induce the stepped effect. Changing the value of **input a** in realtime will produce an extraneous (but possibly useful) effect.

Range of values for <b>input a</b>		Total number of steps as <b>input b</b> moves from min to max	
from	to	(when <b>input b</b> is bipolar)	(when <b>input b</b> is unipolar)
0	.0625	1 (no effect)	1
.0625	.125	2	1*
.125	.1875	3	2
.1875	.25	4	2*
.25	.3125	5	3
.3125	.375	6	3*
.375	.4375	7	4
.4375	.5	8	4*
.5	.5625	9	5

.5625	.625	10	5*
.625	.6875	11	6
.6875	.75	12	6*
.75	.8125	13	7
.8125	.875	14	7*
.875	.9375	15	8
.9375	1	16	8*

As an example, consider the FUN we set up at the beginning of the previous section: the Mod Wheel was assigned as **input a**, and the data slider as **input b**. The FUN was assigned as Src1 on the PITCH page, and the depth of Src1 was set to 1200 cents. If you push the Mod Wheel all the way up, the value of **input a** will be +1. This will set the number of steps at 8, since the data slider sends a unipolar control signal. With your MIDI controller's data slider at minimum, play and sustain a note. Then move the data slider slowly up. The pitch of the note will jump up an octave in 8 steps as you move the data slider all the way up.

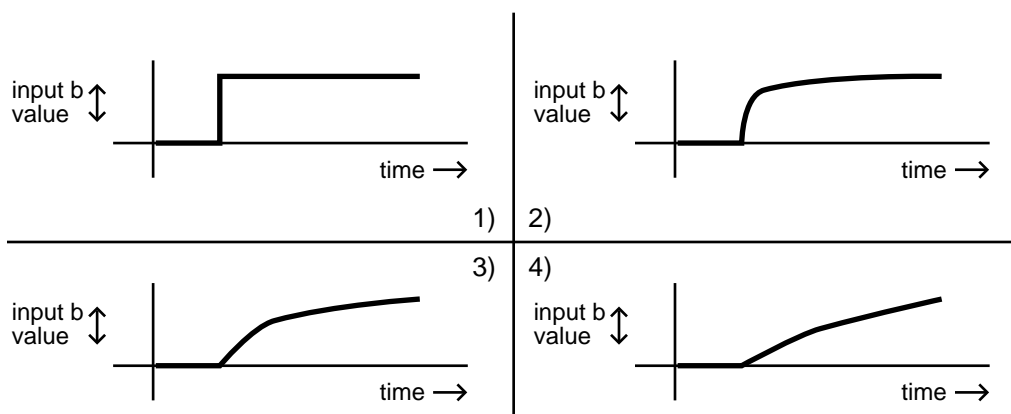
If the value of **input a** is negative, it's multiplied by -1, so its value always falls within the ranges above. When **input b** is bipolar and the resulting number of steps is an odd number, the steps are centered around a value of 0—that is, the center step is equivalent to no effect from **input b**. When the number of steps is even, a value of zero is not included in the steps. This is also true for the values marked by an asterisk when input b is unipolar.

### lowpass (f = a, b)

This equation might be called a lag equation. Its effect is to introduce a delay in the K2500's response to changes in the value of **input b**. It works by filtering (reducing) higher values of **input b**. The value of **input a** determines the degree to which the values of **input b** are filtered. Low values for **input a** will induce a long lag when the value of **input b** changes. High values will shorten the lag. When **input b** remains constant at a high level, low values of **input a** will cause the FUN to sweep up slowly from 0 to the value of **input b**. Higher values for **input a** will cause the FUN to sweep more rapidly.

The four graphs below show the effect of different values for **input a** on the change of **input b**. In each graph, the value of **input b** jumps from 0 to +1. In graph 1, the value of **input a** is +1. Each successive graph represents the same change in the value of **input b**, at successively lower values for **input a**.

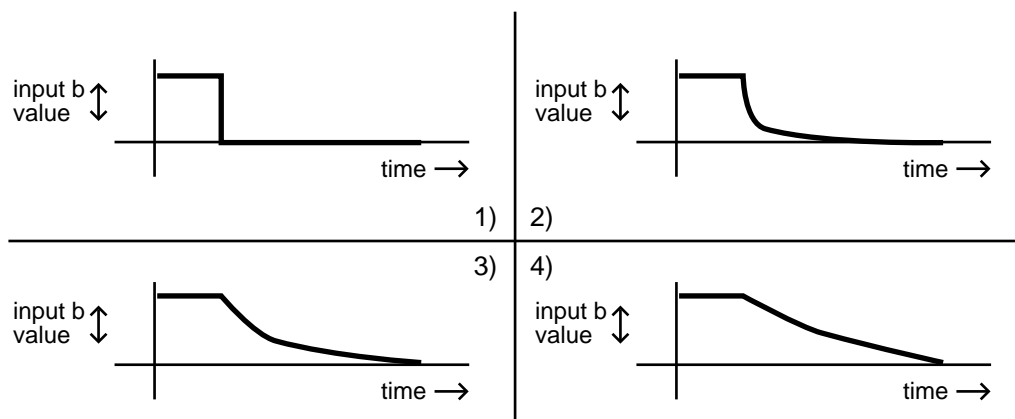
This equation works as intended only when the value of **input a** is 0 or positive. Negative values for **input a** will result in a much less predictable response than positive values. You might like the effect, but it won't be anything like what we've just described.



### hipass ( $f = a, b$ )

With this equation the low values of **input b** are filtered according to the value of **input a**. This causes somewhat different results compared with the lowpass equation above. At low values for **input a**, low values for **input b** will have little effect, while high values for **input b** will cause the FUN to quickly reach full effect then slowly sweep down to its starting level. At high values for **input a**, a rapid change in the value of **input b** will have little effect. At low values for **input a**, rapid changes in the value of **input b** will cause the FUN to respond quickly to the change, then slowly fade back to minimum effect. Listening to the effects at different values for each input will give you the best understanding.

The four graphs below show the effect of different values for **input a** on the change of **input b**. In each graph, the value of **input b** drops from +1 to 0. In graph 1, the value of **input a** is +1. Each successive graph represents the same change in the value of **input b**, at successively lower values for **input a**.



### $b / (1 - a)$

This is another weighted difference equation similar to the first six. The value of **input a** is subtracted from 1. The value of **input b** is then divided by the difference. You'll get considerably different results for different input values of **a** and **b**.

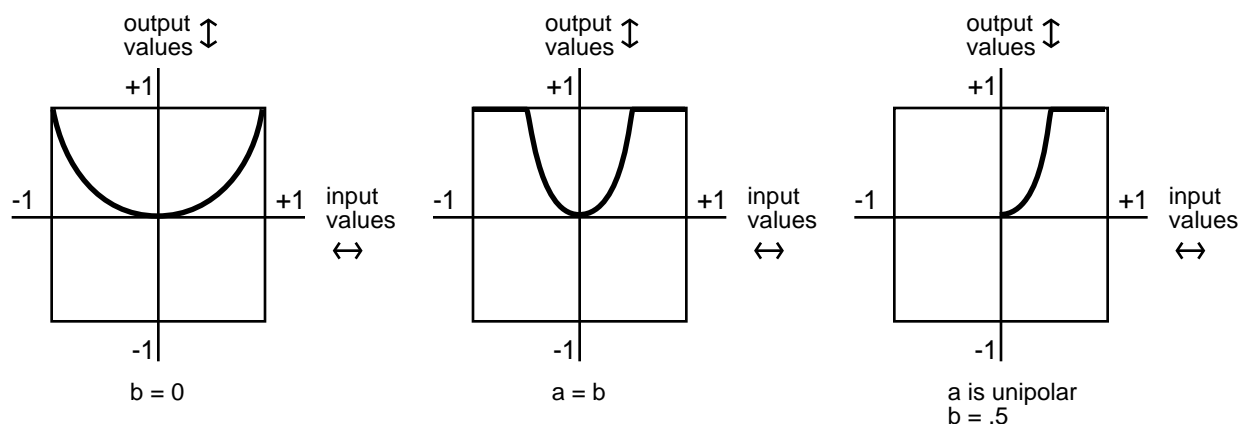
### $a(b-y)$

Think of this equation as reading "y is replaced by the result of the function  $a(b-y)$ ." The value of y indicates the value of the FUN's output signal. Every 20 milliseconds, the K2500 takes the current value of y, runs the equation, calculates a new value of y, and inserts the new value into the equation. Consequently the value of y will change every twenty milliseconds. Here's an example. When you play a note, the K2500 starts running the FUN. The first value for y is always 0. We'll assume the value of **input a** is +.5, and the value of **input b** is +1. The first time the K2500 evaluates the FUN, the result of the equation is  $.5 \times (+1 - 0)$ , or .5. So the FUN's output value after the first evaluation is .5. This becomes the new value for y, and when the K2500 does its next evaluation of the FUN, the equation becomes  $.5 \times (+1 - .5)$ , or .25. The resulting output value is .25, which becomes the new value for y. For the next evaluation, the equation is  $.5 \times (+1 - .25)$ , or .375.

### $(a + b)^2$

The values of **inputs a** and **b** are added, and the result is squared (multiplied by itself). This will change the linear curve of a unipolar control signal into a curve that's lower at its midpoint (by a factor of 2). Bipolar control signals will generate curves that are high at both ends, and 0 in the middle.

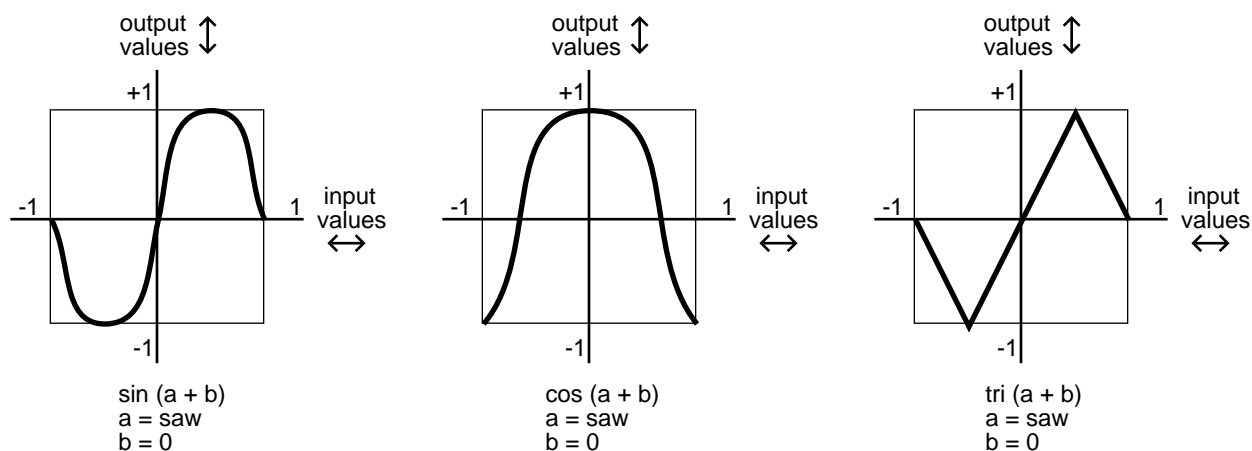


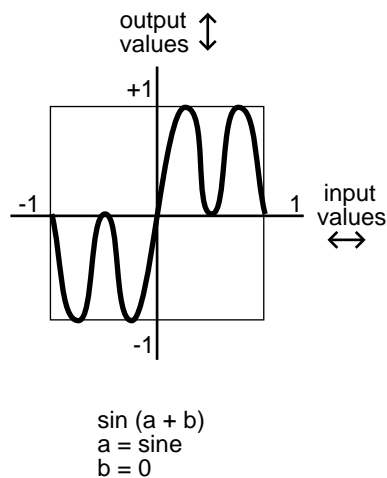


### $\sin(a + b)$ , $\cos(a + b)$ , $\text{tri}(a + b)$

These equations are intended to be used with inputs that are sawtooth waves—for example, **input a** might be LFO1 with its shape set as a sawtooth. Each equation will map a sawtooth-shaped input into a sine-, cosine-, or triangle-shaped output. Other input waveform shapes will result in outputs with more complex waveform shapes.

Other ways to get sawtooth shapes as inputs to these FUNs are to use other FUNs as the inputs, with their equations set as any of the ramp equations described later in this section (see the note at the end of this chapter about the evaluation order of the FUNs). You could also use LFOph1 or LFOph2 as inputs. The first three graphs below show the result of these functions when **input a** is a rising sawtooth wave, and the value of **input b** is 0. The fourth shows the result of the  $\sin(f=a + b)$  equation when the value of **input b** is 0 and **input a** is a sine wave.



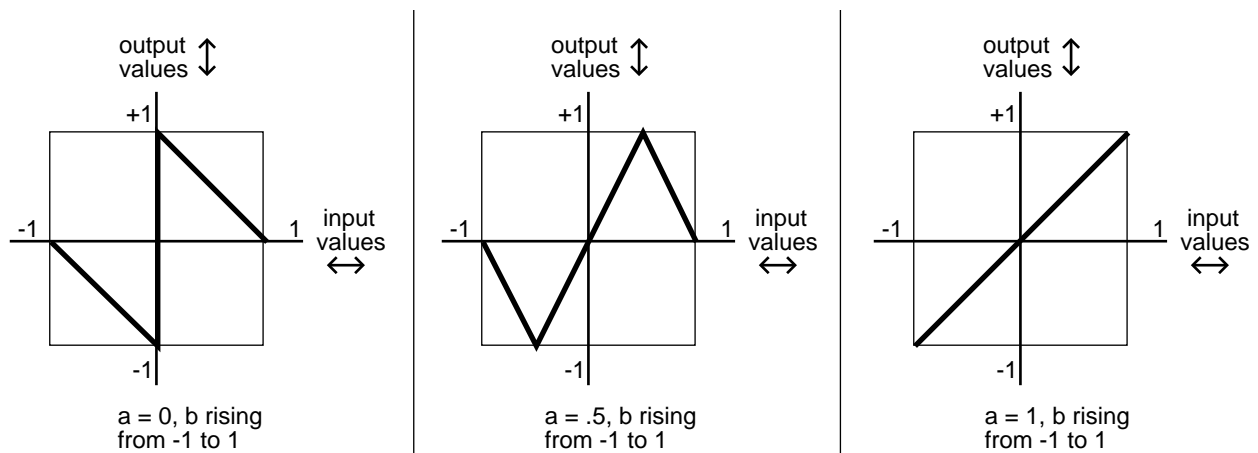


## Warp Equations

The next five equations all behave similarly, and are intended to be used as follows: the value of **input a** is the controlling value, and normally remains constant, although it doesn't have to. The value of **input b** is expected to change over time; **input b** might be an LFO, for example. The value for **input a** affects how the FUN calculates its output value while the value of **input b** changes.

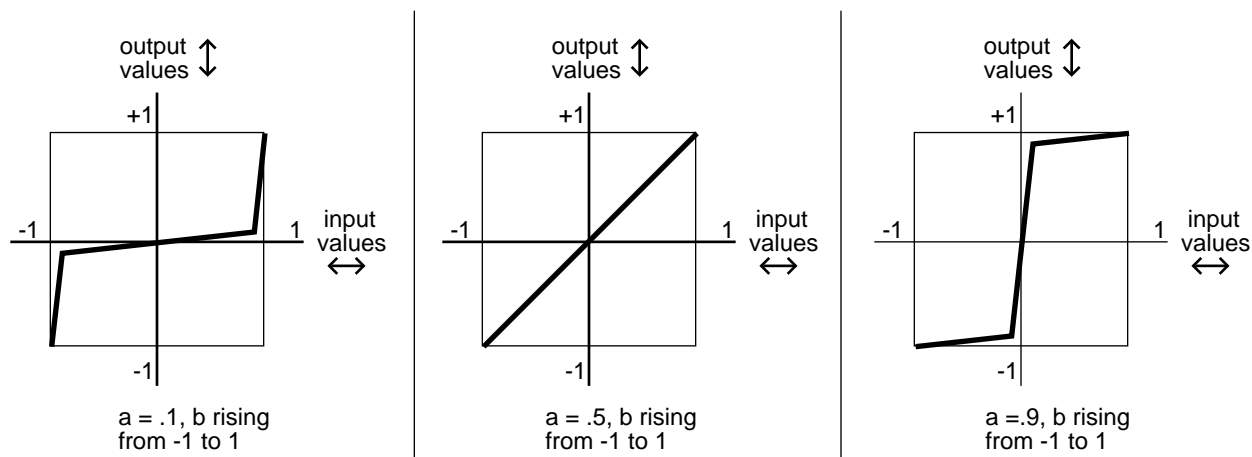
### warp1(a, b)

We call this the Vari-slope™ equation. The value of input a controls the mapping of values for **input b**. If **input b** is a sawtooth wave, different values for **input a** will change it into a triangle wave. If **input b** is a more complicated waveform, the output waveform is also more complicated.



### warp2(a, b)

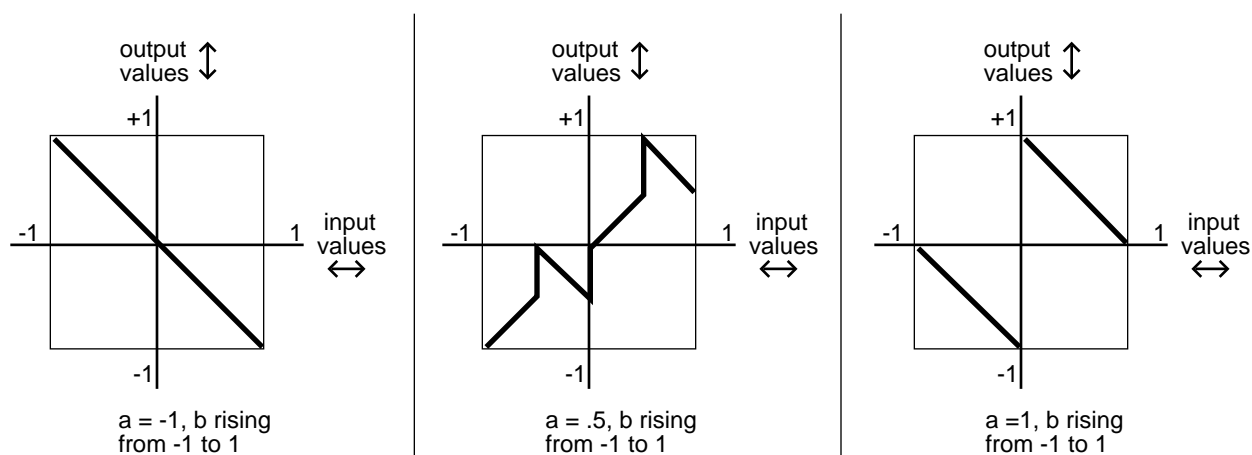
We call this equation Slant-square.<sup>TM</sup> Again, the value of **input a** controls the mapping of values for **input b**. If **input b** is a sawtooth wave, different values for **input a** will turn it into a number of variations on square waves.



### warp3(a, b)

We call this one the Variable Inverter.<sup>TM</sup> It looks at the binary numbers that represent the values of **inputs a** and **b**, compares the corresponding bits in each number, and performs an XOR operation on them (we'll explain that below). The resulting number is converted into the output value. This can produce some erratic results, but if variety is what you're after, this equation will give it to you. You'll get your best results when **input b** is an LFO with a slow rate.

The XOR operation is a subprogram that applies a truth table to each of the digits in the binary numbers that represent the values of **inputs a** and **b**. Each of these numbers is a string of 16 digits (bits); each bit is either a 0 or a 1. The subprogram looks at the first bit of each number. If they're both 0s, the resulting value is 1. If one is a 0 and the other is a 1, the result is 1. If they're both 1s, the result is 0. This process is repeated for the remaining 15 bits of each number, and a new 16-bit number is generated. This number represents the output value of the FUN.

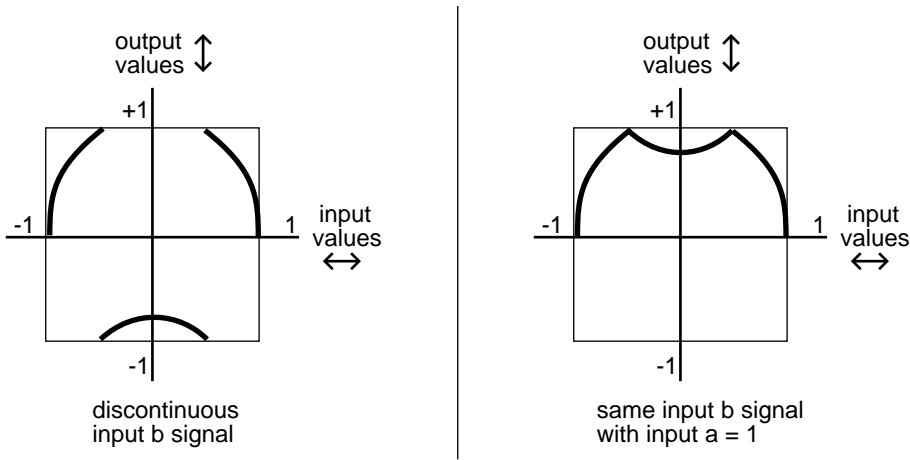


warp4(a, b)

This equation, the Period Inverter,<sup>TM</sup> is based on repeated evaluations of the value of **input b**. The K2500 compares each new value of **input b** with the value from the previous evaluation. If the absolute value (always a positive number) of the difference between the two is greater than the value of **input a**, the current value of **input b** is multiplied by -1.

The primary feature of this equation is that it will take a discontinuous signal and make it continuous. If, for example, FUN1 uses an equation like  $a(y + b)$ , its output can wrap around from +1 to -1, or vice versa. You could set FUN1 as **input b** for FUN2, set **input a** of FUN2 to ON (+1), and FUN2 would remove the discontinuity from the signal. The first graph below shows a hypothetical output signal with such a discontinuity, and the second shows how FUN2 in this case would make the signal continuous without drastically changing its shape.

If, on the other hand you *want* the signal to become discontinuous, you can use the warp4(a, b) equation in a single FUN, with **input a** set to OFF (0), and the signal would be multiplied by -1 with each evaluation of **input b**.



warp8(a, b)

This relatively simple equation is  $a \times b \times 8$ . If the result is beyond the range of -1 to +1, it wraps around from +1 to -1 (or vice versa), until it's within the allowable range. The table below shows some examples of how this works.

$a \times b \times 8 =$	final output value
-7.4	.4
-4.2	-.2
-1.8	.8
-.6	-.6
.4	.4
1.2	-.2
2.6	.6
5.4	-.4

a AND b

The values of **inputs a** and **b** are interpreted as logical quantities—they're considered TRUE if they're greater than +.5, and FALSE otherwise. This turns the FUN into an on/off switch. In the

model we set up in the previous section, FUN1 was set to control Src1 on the PITCH page, and Src1's depth was set to 1200 cents. With this equation, both **input a** (the Mod Wheel in this case) and **input b** (the data slider in this case) would have to be more than halfway up for the FUN to switch on. The pitch would jump 1200 cents as soon as both control sources moved above their halfway points. As soon as one of them moved below its halfway point, the pitch would jump back to its original level.

This equation can be used to trigger ASRs, or as a layer enable control, or for any control source that toggles on and off. If you set one of the inputs to an LFO, the FUN would switch on and off every time the LFO's signal went above +.5 (as long as the other input was also above +.5).

### **a OR b**

This equation is very similar to a AND b. The only difference is that the FUN will switch on when the value of either **input a** or **input b** moves above +.5.

## **Sawtooth LFOs**

The next six equations case the FUN to generate a sawtooth LFO as its output signal. Each performs a different operation on the values of **inputs a** and **b**, and the resulting value is multiplied by 25. The result determines the frequency of the LFO. If the value is a positive number, the LFO has a rising sawtooth shape. If the value is negative, the LFO has a falling sawtooth shape. When the resulting values are large (above 10 or so), the output waveform is not a pure sawtooth; a bit of distortion occurs.

### **ramp(f=a + b)**

The values of **inputs a** and **b** are added, then multiplied by 25.

### **ramp(f=a - b)**

The value of **input b** is subtracted from the value of **input a**, and the difference is multiplied by 25.

### **ramp(f=(a + b) / 4)**

The values of **inputs a** and **b** are added, and the sum is divided by 4. This value is multiplied by 25.

### **ramp(f=a \* b)**

The values of **inputs a** and **b** are multiplied, and the result is multiplied by 25.

### **ramp(f=-a \* b)**

The value of **input a** is multiplied by -1, then multiplied by the value of **input b**. The result is multiplied by 25.

### **ramp(f=a \* 10^b),**

10 is raised to the power of **b**, then multiplied by the value of **input a**. The result is multiplied by 25.

## **Chaotic LFOs**

The next five equations function somewhat like the equation  $a(b-y)$  described earlier, in that they start with a value of 0 for  $y$ , evaluate the equation, and use the result as the new value of  $y$  for the next evaluation. Although they all can function as LFOs (they can have a repeating cycle of output values), they can become chaotic depending on the input values.

**$a(y + b)$** 

The values of  $y$  and  $b$  are added, then multiplied by the value of  $a$ .

 **$ay + b$** 

The values of  $a$  and  $y$  are multiplied, then added to the value of  $b$ .

 **$(a + 1)y + b$** 

1 is added to the value of  $a$ . The sum is multiplied by the value of  $y$ . The result is added to the value of  $b$ .

 **$y + a(y + b)$** 

The values of  $y$  and  $b$  are added. The sum is multiplied by the value of  $a$ . The result is added to the value of  $y$ .

 **$a |y| + b$** 

The absolute value of  $y$  is taken (if it's a negative value, it's multiplied by -1). The absolute value of  $y$  is multiplied by the value of  $a$ . The result is added to the value of  $b$ .

**Sample  $b$  On  $a$** 

This is a sample and hold function. The values of **inputs**  $a$  and  $b$  are interpreted as logical quantities, as described for the equations  $a$  AND  $b$ ,  $a$  OR  $b$ . When the value of **input**  $a$  changes from FALSE to TRUE (goes above +.5), the value of **input**  $b$  at that moment is sampled (recorded), and becomes the FUN's output value. This value remains constant until **input**  $a$  makes another transition from FALSE to TRUE.

**Sample  $b$  On  $\sim a$** 

This works like the previous equation, but the value of **input**  $b$  is sampled whenever the value of **input**  $a$  makes a transition from TRUE to FALSE.

**Track  $b$  While  $a$** 

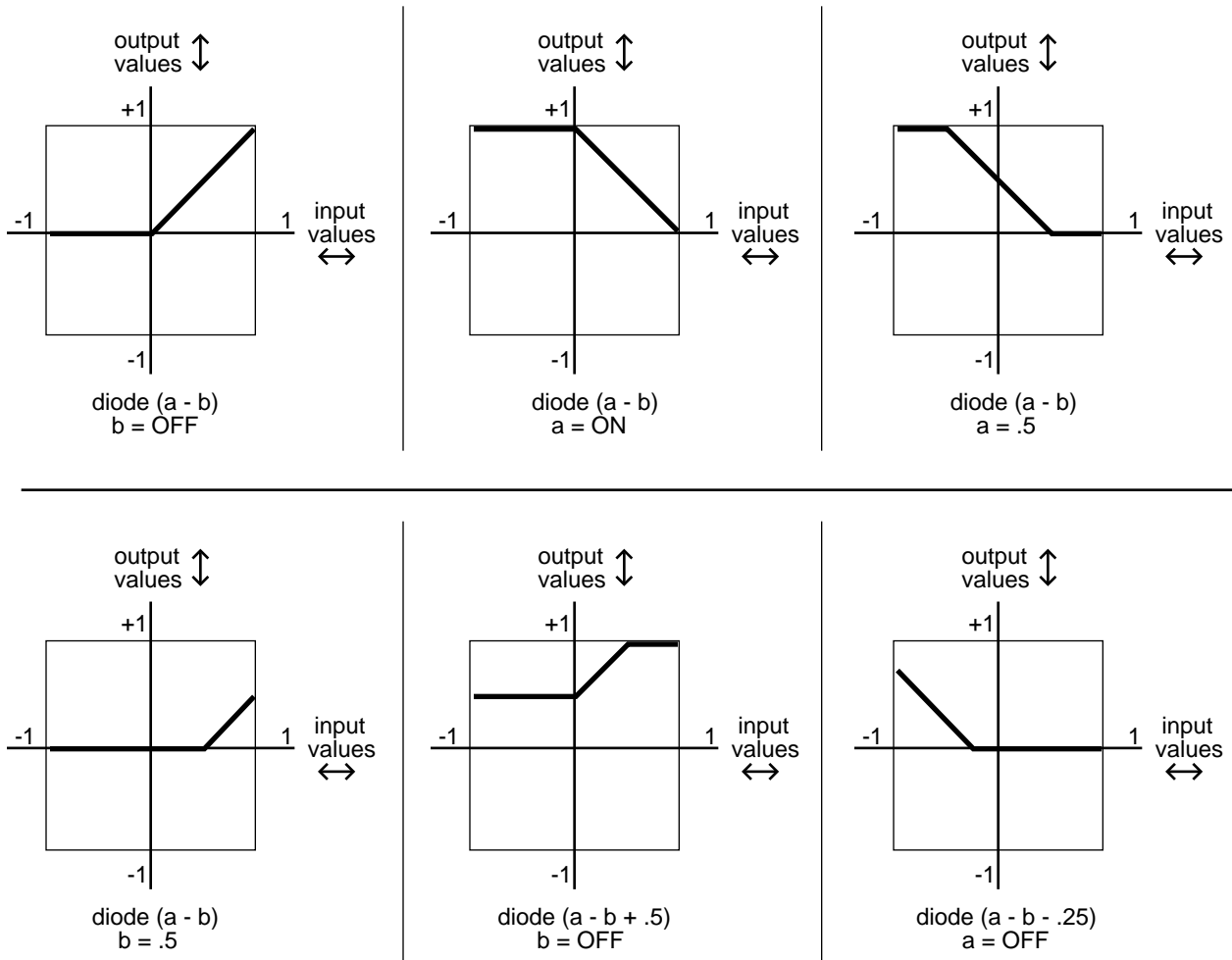
This equation also interprets the values of **inputs**  $a$  and  $b$  as logical quantities. While the value of **input**  $a$  is TRUE, the value of **input**  $b$  is used as the FUN's output value. The output value changes exactly as the value of **input**  $b$  changes. When the value of **input**  $a$  goes FALSE, the FUN's output value freezes and remains constant until the value of **input**  $a$  becomes TRUE again. The FUN's output value then continues to track the value of **input**  $b$ .

**Track  $b$  While  $\sim a$** 

This is the opposite of the previous equation. The FUN's output value tracks the value of **input**  $b$  as long as the value of **input**  $a$  is FALSE.

## Diode Equations

The remaining equations perform a diode-like function; only positive input values are significant. If the result of the equation is negative, the FUN's output value is 0. You can use these equations to limit bipolar control signals to unipolar values. Normally you'll use these by setting input  $a$  or  $b$  to ON or OFF, and assigning some control source to the other input. These will enable you to produce a variety of output curves.



Diode (a - b) simply subtracts the value of **input b** from the value of **input a**. If the difference is less than 0, the output value is 0.

Diode (a - b + .5) adds a constant of +.5 to the difference of (a - b), then maps all negative values to 0. The curve is the same shape as Diode (a - b), but shifted upward 1/4 of the range between -1 and +1.

Diode (a - b - .5) is the same curve as diode (a - b), shifted downward 1/4 of the range.

Diode (a - b + .25) shifts the curve up 1/8 of the range, and diode (a - b - .25) shifts the curve down 1/8 of the range.

## The Order of Evaluation for FUNs

The K2500 is a computer, and processes information at very high speeds. Every 20 milliseconds, it checks the condition of every active parameter, evaluates any changes, and processes the new information. This is done according to a rigid set of priorities that determines the sequence in which the parameters are evaluated.

The status of each control source is evaluated in turn, according to the sequence in which they appear in the Control Source list. In the case of the FUNs, they are evaluated in the following order: FUN1, FUN2, FUN3, FUN4 (although there are a few control sources that get evaluated between FUN2 and FUN3).

This sequence of evaluation becomes significant if you assign a FUN as the input for another FUN. You should always assign FUNs as inputs for higher-numbered FUNs. For example, you might assign FUN1 as input a for FUN2. Since the K2500 needs to know the value of FUN1's output before it can evaluate FUN2, assigning the FUNs in this way will ensure that the K2500 can evaluate both FUNs as quickly as possible.

If you were to assign FUN2 as an input for FUN1, then when the time came for the K2500 to evaluate FUN1, it wouldn't know the current value for FUN2's output, so it would evaluate FUN1 according to the previous value of FUN2. There would be a delay of one evaluation cycle before a change in FUN2 would be reflected in FUN1. This might have an adverse affect on the start of notes as you play. You won't harm anything, but you might not hear what you expect to hear.



## Chapter 17

# Other Editors

In addition to the editors accessible from their respective modes, there are three editors that enable you to modify other performance parameters of the K2500. They are the Intonation Table Editor, the Velocity Map Editor, and the Pressure Map Editor.

### The Intonation Table Editor

Intonation tables define the interval between the notes in each octave. The default intonation table is "1 Equal," which sets precisely equal intervals between notes—the standard for modern western music. If you're interested in playing other styles of music, you can use any of the K2500's other factory intonation tables, or create you own with the Intonation Table Editor.

The intonation table is selected in Master mode with the Intonation parameter. Here's the list of available factory intonation tables (check the *Reference Guide* for brief descriptions):

- |    |               |
|----|---------------|
| 1  | Equal         |
| 2  | Classic Just  |
| 3  | Just b/7th    |
| 4  | Harmonic      |
| 5  | Just Harmonic |
| 6  | Werkmeister   |
| 7  | 1/5th Comma   |
| 8  | 1/4th Comma   |
| 9  | Indian Raga   |
| 10 | Arabic        |
| 11 | 1Bali/Java    |
| 12 | 2Bali/Java    |
| 13 | 3Bali/Java    |
| 14 | Tibetan       |
| 15 | CarlosAlpha   |
| 16 | Pyth/aug4     |
| 17 | Pyth/dim5     |

As you scroll through the options for the Intonation parameter, you'll notice an eighteenth entry with a name like "Obj *vn.n*". This isn't really an intonation table; it's simply a convenient place for the K2500 to keep a record of the current version of ROM objects. If you ever need to find out the version of ROM objects loaded into your unit, this is the place to check. Just remember, if you're using a non-standard intonation table, that you'll need to reset the Intonation parameter after you check the ROM object version list at "table" 18.

It's important to emphasize that the detuning values for each parameter affect the intervals in each octave independently. That is, notes that are an octave apart will remain perfectly tuned regardless of the detuning between notes within the octave. Consequently, editing intonation tables will not enable you to create tunings with more (or fewer) than 12 notes per octave. If you want to do this, use the Keymap Editor, assign each key to its own key range, then tune the samples in each key range.

When you're ready to edit an intonation table, select Master mode, then select the Intonation parameter. Use any data entry method to select the intonation table you want to edit. Check the

value of the intonation key parameter (IntonaKey). This sets the tonic, or reference note for the parameters on the Intonation Table Editor page. We'll explain this below.

Enter the Intonation Table Editor by pressing the Edit button. The Intonation Table Editor page displays a graphic representation of a C octave, with the low C always representing the tonic. The values for these parameters indicate the amount of detuning applied to each note relative to perfectly equal intonation. If you look at the values for the intonation table 1 Equal (as illustrated below), you'll see that the value of each parameter is 0. None of the notes is detuned, so the intervals between each note are exactly the same. Take a minute to look at the values for some of the other intonation tables, and listen to the effect on the tuning.

EditIntonationTable (cents)									
0		0		0		0		0	
0	0	0	0	0	0	0	0	0	0
Name		Save		Delete		Dump			

The top line of the display gives you the usual reminder of your location. The bottom line labels the soft buttons for the Intonation Table Editor, which let you perform the usual librarian functions of naming, saving, and deleting the currently selected intonation table, or dumping it via SysEx.

Assume for now that the IntonaKey parameter is set to a value of C (which is the default). This means that each C is the reference point for defining the intervals of all the other notes.

Let's say you want to create an intonation table that flats the 5th in each octave. Select the value on the G note displayed by the Intonation Table Editor, and set its value to a negative number—for example, -12 cents. This will reduce the interval between the tonic (C, in this case) and the 5th (G, in this case) by 12/100ths of a semitone. All the Gs will remain exactly an octave apart from each other, but they'll be 12 cents flat from their normal pitches.

Press the Compare (Disk mode) button to hear the difference between the edited table and the original. Press the **Save** soft button to begin the save dialog, where you can rename and save your edited intonation table.

The changes we've discussed here are based on a value of C for the IntonaKey parameter. If you were to select Master mode and change the value of the IntonaKey parameter, you'd change the tonic note that the intonation table uses to define the intervals of the other notes. Let's say you change IntonaKey from C to G. The tonic changes to G, and the intervals in each octave are defined relative to G. The minor 2nd is now G#, the major 2nd is A, and so on. In the example we gave above, all the Ds would now be 12 cents flat.

A good reference source for descriptions of various alternative intonations is a book by Scott R. Wilkinson entitled *Tuning In: Microtonality in Electronic Music*, available from the Hal Leonard Publishing Corporation.

## The Velocity Map Editor

Velocity maps affect the way the K2500 interprets attack velocity values. Velocity maps are assigned as the values for two different parameters: `VelocMap` on the `XMIT` page in MIDI mode, and `VelocityMap` on the `RECV` page in MIDI mode. (They also apply to the `VelTouch` parameter in Master mode, but since the `VelTouch` parameter doesn't affect the K2500, you shouldn't change its default value.) There's only one list of velocity maps, and all three of these parameters use this list. Even so, certain maps are intended for use with the `VelTouch` parameter in the keyboard version of the K2500, and others for use with the MIDI mode parameters. You can think of these as internal maps and MIDI maps. With the K2500, you should use only the MIDI maps, and unless you have a problem with the K2500's velocity response, or that of MIDI devices connected to the K2500's MIDI Out port, you should leave both the Receive Velocity map and the Transmit Velocity map set to their default values.

Every attack velocity value generated by your MIDI controller is mapped through the velocity map assigned on the MIDI mode `RECV` page before being passed to the sound engine. If you're using the Local Keyboard Channel feature, attack velocity values received at the K2500's MIDI in port go first through the Receive Velocity map, then to the sound engine, then through the Transmit Velocity map (as selected with the `VelocMap` parameter on the `XMIT` page in MIDI mode), and to the MIDI Out port.

The `VelocMap` parameter on the `XMIT` page in MIDI mode affects only the velocity values the K2500 sends to its MIDI Out port. Normally you'll leave its value set to 1 Linear, especially when you're recording sequences with a personal computer or hardware sequencer. You might want to adjust this parameter, however, when you need to change the response of MIDI slaves that you're driving from the K2500. If, for example, you're overdriving a DX7 that you have slaved to the K2500, try the map 6 Hard2.

The `VelocityMap` parameter on the `RECV` page in MIDI mode affects how the K2500 responds to attack velocity values received at its MIDI In port. Normally you'll leave this set to Linear, as with the `VelocMap` parameter on the `XMIT` page. You might want to adjust this parameter, for example, to boost or cut the volume response of the K2500 during playback in Song mode, or when playing a sequence from an external sequencer.

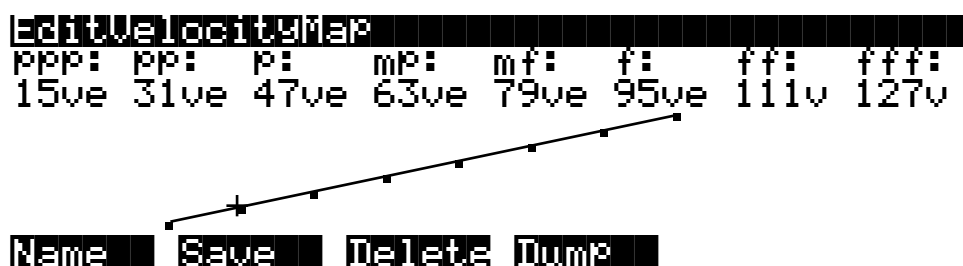
Here's a list of the velocity maps stored in ROM (the factory presets). Keep in mind, however, that you should use only the linear MIDI maps; the internal maps are meant for the K2500.

- 1 Linear (the default on the `XMIT` and `RECV` pages in MIDI mode): Velocity values are passed through this MIDI map unchanged.
- 2 Light1
- 3 Light2.
- 4 Light3.
- 5 Hard1.
- 6 Hard2.
- 7 Hard3.

The attack velocity value of every note received by the K2500's sound engine is routed through the map assigned to the `VelocityMap` parameter on the `RECV` page in MIDI mode). The velocity maps convert each attack velocity value into a new value, depending on the curve of the map. By editing velocity maps, you can change the attack velocity value that the maps calculate.

## Using the Velocity Map Editor

To get to the Velocity Map Editor, select any of the parameters we've mentioned above—`VelTouch` (Master mode) `VelocMap` (`XMIT` page, MIDI mode), or `VelocityMap` (`RECV` page, MIDI mode). Use any data entry method to select the map you want to edit, then press Edit.



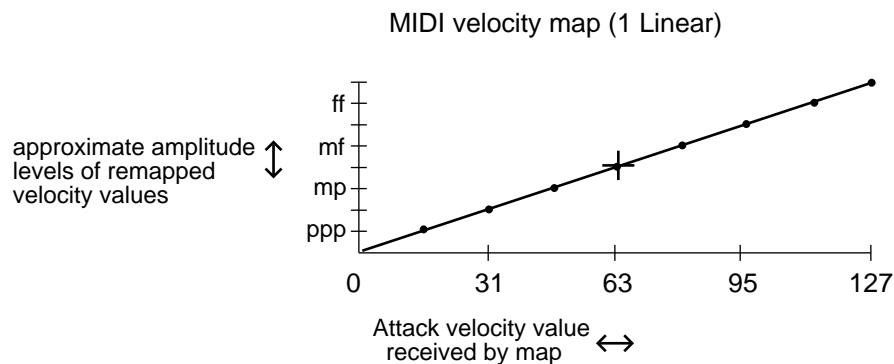
The top line of the display gives you the usual reminder of your location. The bottom line labels the soft buttons, which execute the normal librarian functions of naming, saving, or deleting velocity maps, or dumping them via SysEx. The values you see in this diagram are the settings for the velocity map 1 Linear.

The eight parameters on the Velocity Map Editor page are expressed in terms of the eight dynamic levels of standard musical notation. As far as the velocity maps are concerned, the dynamic levels represented by the parameter names are simply reference points—for example, when using the map 1 Linear, an attack velocity value of 63 will result in a remapped velocity value at the mp level of the K2500's dynamic range.

The values for these parameters are expressed in "vels," which correspond to the standard MIDI attack velocity values of 0—127. The values you set for each parameter determine the velocity value that must be generated to achieve the dynamic level for that parameter. For example, if you're using the map 1 Linear as the value for the VelocityMap parameter, then the K2500 will play notes at full volume only when the attack velocity values for those notes are 127. Notes with attack velocities from 0 to 15 will be played at the lowest volumes. They wouldn't all play at the *same* volume; their volumes would be graduated, but would be at the low end of the scale.

The values for these parameters have another function as well. They determine the velocity thresholds for the velocity triggers (VTRIG), as well as the velocity crossovers for multi-velocity keymaps. Once again using the example of the map 1 Linear, if you have the VelCrossover parameter in the Keymap Editor set to a value of mp, then attack velocity values higher than 63 will trigger the high-velocity range of a dual-velocity keymap, while velocity values of 63 or lower will trigger the low-velocity range.

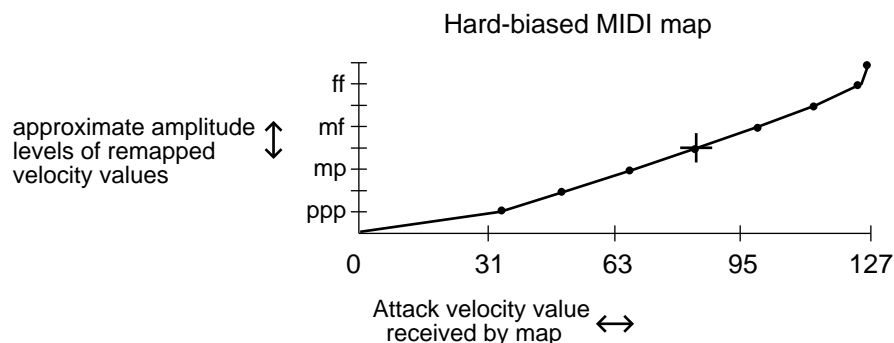
The curve that represents the velocity map is shown in the center of the page. Think of this curve as a plot on a graph, as shown in the diagram below. The horizontal axis gives the attack velocity values received by the velocity map. The vertical axis gives the approximate dynamic level at which the K2500 will play the note (in response to the remapped velocity value).



When you select a parameter, the small crosshairs move to indicate the position on the curve of the dynamic level represented by that parameter. Each of the eight levels is marked on the curve by a small dot. The crosshairs jump to one of the dots when you select the corresponding parameter. The dots always have the same vertical location, but move to the left or right depending on the velocity value assigned to them. The farther to the right that one of these dots is located, the higher the attack velocity value required to play a note at that level. As the curve suggests, velocity values between those set for the eight dynamic levels will be remapped to a new velocity represented by the height of the curve.

As you change the value of the currently selected parameter, the shape of the curve changes to reflect the new value. As an example, let's assume that you're driving a MIDI slave synth from the K2500's MIDI Out port, and you get full-amplitude sounds from the slave, even when you play softly on your MIDI controller. To fix this, you'd edit the map to give it a harder bias (or use one of the preset MIDI maps with a hard bias). That is, you want to have to strike your controller's keys harder to get full amplitudes.

The diagram below shows a hard-biased linear map similar to the Hard3 map. Although the even progression from low to high remapped values remains, the entire curve is shifted so that a given input velocity value will remap to a lower value.



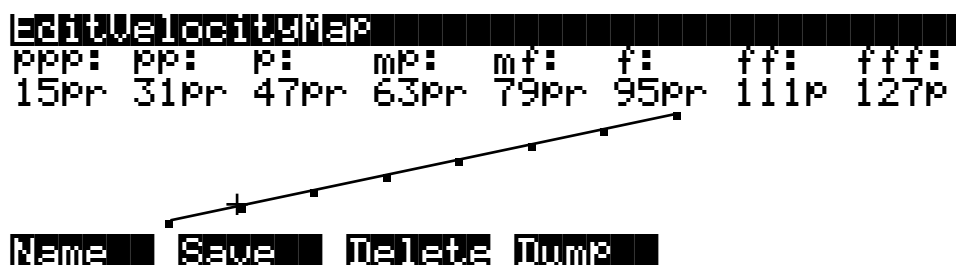
## The Pressure Map Editor

Pressure maps function in much the same way as velocity maps, but they affect the K2500's response to mono pressure messages received from your MIDI controller. Like the velocity maps, pressure maps convert pressure values of 0—127 to new values that are sent to the sound engine. Also like the velocity maps, pressure messages received at the K2500's MIDI In port from your MIDI controller pass through the Receive Pressure map, then to the sound engine. If you're using the Local Keyboard Channel feature, the pressure messages pass first through the Receive Pressure map, then through the Transmit Pressure map, then to the MIDI Out port.

Pressure maps are assigned as the values for the PressTouch parameter in Master mode, the PressMap parameter on the XMIT page in MIDI mode, and the PressureMap parameter on the RECV page in MIDI mode. There's one list of pressure maps, which is used to select the pressure map for all three parameters. As with the velocity maps, you should use only the MIDI map (ID 1) for the MIDI mode parameters, leave the value of the PressTouch parameter at its default value. Here's the list of all seven maps. Keep in mind, however, that you should use only the linear MIDI maps; the internal maps are meant for the K2500.

- 1 Linear: The default internal and MIDI pressure map.
- 2 Easy.
- 3 Easier.
- 4 Easiest.
- 5 Hard.
- 6 Harder.
- 7 Hardest.

To edit a pressure map, select one of the three pressure map parameters, and use any data entry method to set its value to the pressure map you want to edit. Edit the map 1 Linear if you want to use your edited map with the MIDI pressure map parameters. Press EDIT, and the Pressure Map Editor page will appear.



The top line of the display gives you the usual reminder of your location. The bottom line labels the soft buttons, which execute the normal librarian functions of naming, saving, or deleting the current pressure map, or dumping it via SysEx.

The eight parameters are expressed in terms of the dynamic levels of standard musical notation. They don't correspond to actual volume levels, they're just a reference point (ppp corresponds to minimum pressure effect, and fff corresponds to maximum). Their values are expressed as "prs" units, and range from 0—127, corresponding to the values of mono or poly pressure messages. The values you set for these parameters determine the pressure message values that must be generated to achieve the dynamic levels expressed by the parameters. Values in between those indicated by the parameters will be remapped as indicated by the curve.

## Chapter 18

# Audio Outputs

### Audio Configurations

The K2500 features ten audio jacks, enabling you to route the K2500's sounds through its own internal effects processor, through an external effects device, or through combinations of the two. Of course the K2500's effects processor can be bypassed as well.

On the rear panel are a pair of stereo mix jacks (MIX L and R), and eight additional jacks—A (L and R), B (L and R), C (L and R), and D (L and R). These can be configured as four stereo pairs or as eight separate outputs. Your audio configuration is determined by two factors. The “hardware” side has to do with which jacks you use and what kind of cables you connect to them. The “software” side depends on the settings you make for parameters on the OUTPUT page in the Program Editor, and on the Master mode page.

The simplest configuration would involve connecting the left and right MIX outputs directly to a mixing board or keyboard amplifier. If you're using only one input to your sound system from the K2500, use the left MIX output for a mono signal. We'll assume for now that you're using only the MIX outputs. The use of the separate outputs is discussed later in this chapter.

Note: we recommend that you make the cable connection to the K2500 (or any instrument) after you've made your other audio connections, since this reduces the chance of creating static electricity that can cause an audible “pop” (and, in extreme cases, cause equipment damage).

### Using the MIX Outputs

If you plan to use the K2500's global effects processor to add dimension to your sounds, you'll need to use the MIX outputs. The audio output to the separate outputs is always dry—that is, the audio signal is sent to the separate outputs without passing through the global effects processor. The MIX outputs and the headphone jack are the only jacks where the signal will appear with effects.

The signal is routed using the OUTPUT page in the Program Editor. While in Program mode, press the EDIT button, then press one of the <more> soft buttons until you see “OUTPUT” on the display's bottom line. Press the corresponding soft button, and the OUTPUT page will appear. Set the value of the Pair parameter to A(FX) if you want the current layer's sound to be routed through the effects processor before appearing at the MIX outputs.

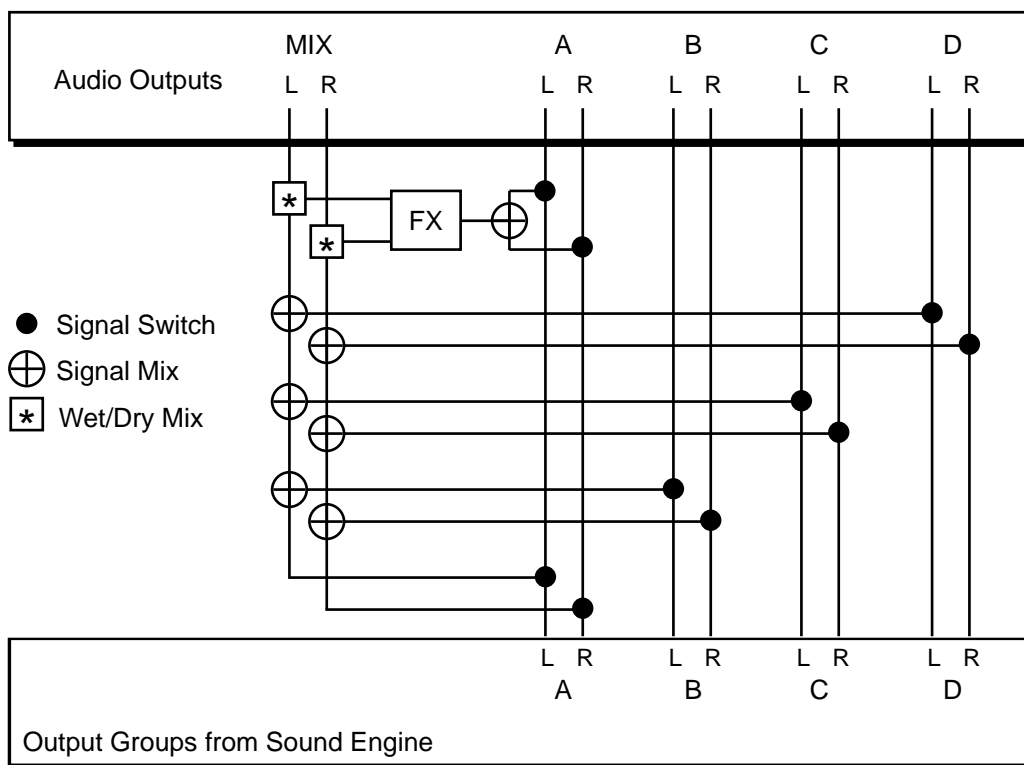
A value of A(FX) means two things: first, the layer's sound will appear with effects at the MIX outputs, and second, that the layer's sound will appear *without* effects at the Group A outputs if standard audio cables are inserted into the Group A output jacks. Set the value of the Pair parameter to B(DRY), C(DRY), or D(DRY) if you want the layer's sound to appear without effects at the MIX outputs. It will also appear at the Group B, C, or D outputs if standard audio cables are inserted into the respective output jacks. Connecting audio cables to the separate outputs will remove part or all of the signal from the MIX outputs. Be sure to set the output pair for every layer in the program; each layer can be routed independently (use the CHAN/Bank buttons to scroll through the layers). You'll also want to set the pan position for each layer while you're on the OUTPUT page. Depending on the program you're working with, you'll have several different options for panning. See “The OUTPUT Page” in Chapter 6 for an explanation of these options. Also see “PANNER” in Chapter 14 to learn how to gain even more control over a layer's pan position.

## Using the Separate Outputs

The K2500's eight separate outputs are arranged in four pairs, A, B, C, and D. These are mono output jacks, and you'll normally plug a standard audio cable into one or more of them. Depending on the OUTPUT page settings you use for a given program's layers, they can function as four stereo output pairs, or as eight separate outputs.

The diagram below illustrates the wiring of the K2500's audio system. From the sound engine, the audio signal can be routed to any of the four output groups. If the signal is routed to Output Group B, C, or D it will appear at the MIX outputs if no audio cables are plugged into those outputs. Plugging cables into any of the separate outputs will divert the signal from the MIX outputs to that output.

If the signal is routed to Output Group A, and audio cables are plugged into the A outputs, the signal will pass directly to the A outputs. If no cables are plugged into the A outputs, the signal will pass from the sound engine to the global effects processor. The left and right sides are mixed before passing to the effects processor. The effects processor splits the signal into left and right sides again after the effects are applied. Effects will be applied to both sides of the signal unless the Out A->FX parameter is set to L Only, in which case only the left side will have effects applied. In either case, the signal will pass to the MIX outputs. Remember that the effects processor isn't stereo; effects can't be panned the way dry sounds can.



If a layer's pan position is centered, then the outputs to which it's assigned will carry an equal mix of the layer's sound in its left and right outputs. This makes the audio outputs a stereo pair with respect to that layer. If the layer is panned all the way to the left or right, the separate outputs function as individual monophonic outputs with respect to that layer. Of course, you could have a multi-layer program with all its layers assigned to the same output group, and pan each layer so that some of them appear at both outputs, while others appear only at the left or right output. If you're using your K2500 for recording and multi-timbral sequencing, you'll want to use the separate outputs to increase your mixdown options, and for adding outboard effects to your sounds. The separate outputs give you a great deal of added flexibility.



## Hardware Audio Switching

It's important that you understand the effect on the MIX outputs when you use the separate outputs. When you connect a standard audio cable to any of the separate outputs, *every* sound (or part of a sound) that's assigned to the output you connect will be removed from the MIX outputs. For example, imagine that you plug a cable into the left output of Group A. The left side of every program layer assigned to Output Group A will disappear from the left side of the MIX outputs, and will appear only at the left Group A output. If a layer is panned all the way left, it will disappear from the MIX outputs entirely. If it's centered, the left side will disappear, but the right side will remain at the right MIX output. A layer that's panned hard right will not be affected in this case. If you use all eight separate outputs, the MIX outputs will not carry any signal.

The pan settings you make on each program's OUTPUT page apply to the separate outputs just as they do for the MIX outputs. Keep in mind, however, that the signal at the separate outputs will always be dry; only the MIX outputs make use of the global effects processor.

Once you have the output groups assigned on the OUTPUT page, you can go to Master mode to specify how the signal will appear at the MIX outputs. On the Master mode page are three parameters that affect the routing of all signals that are sent to the MIX outputs. The settings for these parameters will affect each layer in every program.

The parameters Out A->Mix and Out B->Mix determine the panning of the A and B Group's audio signal at the mixed outs. If they're set to Stereo, then whatever panning is applied to each program (as set on the OUTPUT page in the Program Editor) will show up at the MIX outputs. Output groups C and D are permanently set to Stereo.

If either of these parameters is set to Mono, then programs routed to the corresponding audio output group will send all of their sound at equal levels to both outputs, regardless of any panning that might be applied to them on the OUTPUT page. This is a quick and convenient way to convert the MIX outputs into two mono outputs.

The parameter Out A->FX is intended for special cases when you want an additional separate output that doesn't go through the K2500's effects processor. A value of L Only bypasses the effects processor for all sounds assigned to the right side of Output Group A. There's an example in Chapter 11 that describes how to use this parameter.

## Using Stereo Insert Cables

The preceding description of the separate outputs is only part of the picture. You can make the separate outputs even more useful by using stereo insert cables. A stereo insert cable has a single stereo plug on one end; the other end is split into two mono cables fitted with mono plugs. By plugging the stereo end of a stereo insert cable into one or more of the K2500's separate outputs, you can create all sorts of audio configurations with outboard gear. The tip of the stereo end carries the signal to the send side of the split end, and the ring of the stereo end receives the signal from the return side.

### Effects Return

Plug the stereo end of a stereo insert cable into one of the K2500's separate outputs, then plug the send (tip) side of the split end into an input on an outboard effects box. If you connect the effects box's output to the return (ring) side of the split end, you've created a loop that will send the K2500's output into the effects box, and return it wet to the K2500. Since the loop returns the signal to the same output jack on the K2500, the K2500's MIX outputs act as if there were no cable at all plugged into the separate output, and the signal will appear at one or both of the MIX outputs. If you're using one of the Group A outputs, you can apply the effect of both the outboard box and the K2500's global effects processor to the corresponding side of the K2500's MIX outputs. If you don't want to double the effects in this way, you can go to Effects mode, set the FX mode parameter to Master, and set the Wet/Dry Mix parameter to 0%. You can also set the Pan parameter of programs routed to Output Group A fully right, then go to Master mode

and set the Out A->Mix parameter to mono, and the Out A->FX parameter to L Only. This will send programs routed to Output Group A to the MIX outputs without going through the K2500's effects processor, but will still send the signal to both the left and right MIX outputs.

### INPUT TO K2500

Suppose you want to bring the outputs from a microphone and another synth into the K2500's MIX outputs so you can apply K2500 effects to them. You have two options for setting this up. (You may need to run the mic output through a pre-amp before it gets to the K2500.)

The most versatile way to do this requires an external mixing board. Use standard audio cables to connect the outputs from the other synth and the mic to two of the input channels on the mixing board (for this example, we'll use Channel 1 for the other synth—we'll assume it's a mono unit—and Channel 2 for the mic). Then plug the stereo end of a stereo insert cable into the A Left output of the K2500. Connect the send (tip) side of this cable to Input channel 3 on your mixing board. Connect the return (ring) side of this cable to the left side of a subgroup output on the mixing board. Similarly, connect the stereo end of a second stereo insert cable to the A Right output of the K2500. Connect the send side of this cable to Input channel 4 on the mixing board, and the return side to the right side of the same output subgroup on the mixing board. Pan Channel 3 fully left, and Channel 4 fully right. Pan Channels 1 and 2 as desired.

Here's what happens. Every K2500 sound routed to Output group A will be initially removed from the K2500's MIX outputs, and will pass instead to Channels 3 and 4 on the mixing board. These signals, along with the signals from Channels 1 and 2 on the mixing board (the outputs from the other synth and the mic), will be routed to the mixing board's output subgroup, and will pass to the A outputs of the K2500, which now act as inputs. Since there's a closed circuit between the K2500 and the mixing board, the K2500 handles the input at its A outputs as if it were its own output, and passes it along to the effects processor, then to the MIX outputs. The signals from the other synth and the mic, as well as the K2500's Output group A sounds, will have effects applied to them before they appear at the MIX outputs. (The dry portions of all these sounds will be panned according to your pan settings for Channels 1–4 on the mixing board and the pan settings for the K2500 sounds routed to Output Group A, while the wet portions of all the sounds will appear equally at the MIX outputs.)

A second, simpler method is to pan all the K2500 sounds routed to Output Group A to the left, then connect the output from the other synth or the mic into the K2500's A Right output using a standard audio cable. The K2500 sounds routed to Output Group A will appear at the MIX Left output, and the signal from the other synth or the mic will appear at the MIX Right output. This restricts your panning options somewhat, but is easier to set up.

## Output Groups and MIDI Channels

You can also set a program's output group on the basis of the MIDI channel it's assigned to. This is helpful for mixing purposes, in both studio and live situations. When you're using the MIX outputs, it's an easy way to apply or remove effects without editing the program.

To set the output group on the basis of MIDI channel assignments, select MIDI mode by pressing the MIDI mode button. Then press the CHANLS soft button. Use the CHAN/BANK buttons to select the channel you want to edit. The OutPair parameter controls the output group assignment for all programs assigned to the channel. If you choose a value of "Prog" (the default), then any program assigned to that MIDI channel will be routed according to its own output settings on the OUTPUT page in the Program Editor. Otherwise, programs on that channel will be routed according to the settings for the Output Group parameter on the CHANLS page.

Say for example that you set the OutPair parameter for Channel 1 to a value of A(FX). Now when you select MIDI Channel 1 as the current MIDI channel in Program mode, *any* program you select will appear at the A outputs, or at the MIX outputs (with effects) if no cables are connected to the A outputs. Set the OutPair parameter to B(DRY), and any program you assign to Channel 1 will appear at the B outputs, or dry at the MIX outputs if no cables are connected to the B outputs.

## Chapter 19

# Programming Examples

The other chapters in this manual have described the K2500's features in detail. This tutorial chapter will take you step-by step through several programming operations.

Each of the following examples will begin from the same starting point: the default program with ID 199. This program is included specifically for the purpose of giving you a programming template. Most of its parameters have been set at values that don't affect the sound of the program.

You may want to adjust some of the parameters of Program 199, to create your own customized programming template. Even if you don't, it's a good idea to begin with Program 199 when you're building a new sound, so you'll know exactly what you have from the start.

Keep in mind that none of these examples provides you with a usable program. Instead, the examples are designed to give you tools and concepts which you can apply to your own sounds. Once you become familiar with the programming basics in this chapter, analyze a few of the factory presets by moving through the Program Editor, and observing how those presets were designed. This may help you learn more techniques for creating new sounds.

While in the Program Editor, there are several editing shortcuts you can use. To call up a control source, enter its number on the numeric keypad, or hold ENTER and strike a key on the keyboard (see Control Sources in chapter 6 of the *Reference Guide*). When a highlighted parameter has a Control Source as its value, press EDIT, and you will go directly to that Control Source page. You can also use the Previous Pg, Mark, and Jump buttons (see Chapter 5).

## Example 1

### Trumpet with Delayed Vibrato and Velocity-triggered Fall-offs

Vibrato is a regular oscillation in pitch that adds dimension to any sound. Brass players will often "fall-off" from a note, punching it then letting the pitch roll down smoothly or in small fast steps.

To create these effects, we'll use an LFO to control the pitch, (this is the typical way to create vibrato), and delay it with an ASR. This way you'll hear the vibrato only on notes that you hold for a second or so. The stab will be done with a second ASR controlling pitch and amplitude. The stab's ASR will be triggered by a velocity trigger (VTRIG), so only those notes you play at fortissimo will stab.

Start by selecting Program 199 and pressing EDIT. The ALG page will appear. The first task is to change the keymap. Press the KEYMAP soft button to select the KEYMAP page.

```

EditProg:KEYMAP          <>Layer:1/1

KeyMap:17 Trumpet          Stereo:Off
XPose :0ST               TimbreShift :0ST
KeyTrk:100ct/key         AltSwitch   :OFF
VelTrk:0ct               PlayBackMode:Normal

<more  ALG  LAYER  KEYMAP  PITCH  more>

```

The KeyMap parameter is already selected, and as you can see, the Default program uses the Grand Piano Keymap. Use any data entry method to change the Keymap to Trumpet, which has ID 17. The KEYMAP page should look like the diagram above when you're done. Remember that you can play your MIDI controller's keyboard at any time while editing, so you can listen to each change as you make it.

Next set up the vibrato. Start by selecting the PITCH page (press the PITCH soft button). Use the cursor keys to move the cursor to the Src2 parameter. Use any data entry method to select LFO1 as its value by pressing 1, 1, 4, ENTER on the numeric keypad (or hold ENTER and strike B5 on the keyboard). This assigns LFO1 to control the pitch of the trumpet sample.

The next step is to set the depth of the vibrato. Select the MaxDpt parameter and assign a value of 10 cents (1, 0, ENTER). Since the default program is preset to have your controller's Mod Wheel control the depth of Src2, you can hear the vibrato by pushing the Mod Wheel fully up (LFO1 has non-zero default values in the default program, otherwise, you wouldn't hear the vibrato). If you're not sure you hear the vibrato, try setting the MaxDpt parameter to a larger value.

Next, select the DptCtl parameter and assign a value of ASR2 by pressing 1, 1, 1, then ENTER (or hold ENTER and strike G#5 on the keyboard). This will cause ASR2 to control the depth of the vibrato. At this point, the default values for ASR2 will cause the vibrato to fade in and out.

There are two more steps to programming the delayed vibrato: adjusting the rate of LFO1 and setting up ASR2 to control the vibrato's delay. First, highlight Src2 and press EDIT; this brings up the LFO page. The default value for LFO1's minimum rate (the MnRate parameter) is 2 seconds. Select this parameter with the cursor buttons, and set its value to .16 seconds (1, 6, ENTER). Select the MxRate parameter, and set its value to 4.40 Hz (4, 4, 0, ENTER). Select the RateCt parameter, and assign a value of ASR2 (1, 1, 1, ENTER). The vibrato will still fade in and out because of the default settings of ASR2.

The LFO page should now look like this:

```

EditProg:LFO          <>Layer:1/1

      MnRate:MxRate:RateCt:Shape:  Phase:
LF01:  0.16H  4.40H  ASR2   Sine    0deg
LF02:  2.00H  0.00H  OFF    Sine    0deg

<more  LFO  ASR  FUN  UTRIG  more>

```

Now select the ASR page to adjust the settings for ASR2. Since the cursor is highlighting ASR2 as the value for the RateCt parameter, you can select the ASR page by pressing EDIT.

To program a realistic delayed vibrato, you need to adjust the Mode, Delay, and Attack parameters. Select the Mode parameter and change its value to Hold (use the Alpha Wheel or

Plus/Minus buttons). This will prevent the vibrato from fading as it did. (This fading was caused by the ASR repeating, which was the default setting). Now select the Delay parameter and set its value to .4 seconds (4, 0, ENTER). Select the Attack parameter and change its value to .48 seconds (4, 8, ENTER). The vibrato should now begin to fade in gradually after a short delay, then remain constant at a rate of 4.40 cycles per second. The ASR page should look like the page below. We're finished with the delayed vibrato; next is the velocity stab.

```

EditProg:ASR          <>Layer:1/1
ASR1:  Trig:  Mode:  Delay:  Attack:Releas:
      ON      Hold  1.00s  1.00s  1.00s
ASR2:  ON      Hold  0.40s  0.48s  1.00s

<more  LFO  ASR  FUN  UTRIG  more>

```

We want the stab to drop the trumpet's pitch *and* amplitude, but only when notes are played fortissimo or louder. This is done by using ASR1 as a control source on both the PITCH and AMP pages, then using a velocity trigger (VTRIG) to control ASR1.

First return to the PITCH page (if you're still on the ASR page, press the more> soft button three times, and the PITCH soft button will appear). Press PITCH, then select the Src1 parameter, and set its value to ASR1 (press 1, 1, 0, ENTER or hold ENTER and strike B5). Then select the Depth parameter, and set its value to -1200 cents (+/-, 1, 2, 0, 0, ENTER). ASR1 has non-zero default values in Program 199, so you'll hear the pitch drop an octave if you strike and hold a note. The PITCH page should look like this:

```

Edit Prog:PITCH          >Layer:1/1
Coarse:0S1              Src1  :ASR1
Fine  :0ct              Depth :-1200ct
FineHz: 0.00Hz          Src2  :LFO1
KeyTrk:0ct/key          DptCt1:ASR2
VelTrk:0ct              MinDpt:0ct
                          MaxDpt:-10ct
<more  ALG  LAYER  KEYMAP PITCH  more>

```

The next step is to adjust the characteristics of the stab by programming ASR1. Select the Src1 parameter again, and press EDIT to select the ASR page.

First, select the Trig parameter, and assign a value of VTRIG1 (press 1, 0, 6, ENTER or hold ENTER and strike E5). Select the Delay parameter and set its value to .34 seconds (3, 4, ENTER). Set the Attack parameter to a value of .78 seconds. The ASR page should now look like this:

```

EditProg:ASR          <>Layer:1/1
ASR1:  Trig:  Mode:  Delay:  Attack:Releas:
      VTRIG1 Hold  0.34s  0.78s  1.00s
ASR2:  ON      Hold  0.40s  0.48s  1.00s

<more  ALG  LAYER  KEYMAP PITCH  more>

```

Notice that the soft buttons are identical to those on the PITCH page; this is because you pressed EDIT to call up the ASR page, instead of using the **more** soft buttons.

To make the stab sound realistic, we'll drop the amplitude at the same rate as the pitch. To do this, select the F4 AMP page (press the more> soft button once, then press the F4 AMP soft button). Select the Adjust parameter, and assign a value of 8 dB. This will give the trumpet a bit more punch. Next, select the Src1 parameter, and set its value to ASR1. Then select the Depth parameter and set its value to -68 dB.

Here's an important point to remember: ASR1 is being used to control both the drop in pitch for the stab, and the drop in amplitude as well. All of the control sources can be similarly assigned for as many parameters as you like.

The final step in this example is to set the velocity threshold of the stab. Right now the stab is occurring on almost every note, but we want it to happen only when playing fortissimo or louder. To do this, press the more> soft button three times, then press the VTRIG soft button to select the VTRIG page. The VTrig1 Level parameter is already selected, so just turn the Alpha Wheel until the VTrig1 Level value is ff. Now you can play softly without triggering the stab.

That's it for Example 1. If you want to save your work, the easiest way is to press EXIT. The K2500 will ask you if you want to save. You should probably press the Rename soft button, give your program a new name, then save it with a new ID. See Chapter 5 if you need help with the Save dialog.

## Example 2

### Lowpass Filter, Envelopes

This example will show you how to assign a DSP function to an algorithm block (the 4-pole lowpass filter), and adjust its control parameters. You'll also set up an envelope to control the cutoff frequency of the filter.

Start with Program 199, press EDIT, and select the KEYMAP page (This is explained in a bit more detail in the previous example, if you haven't read it). Change the value to "6 Ensemble Strings." String sounds are especially responsive to lowpass filtering, because they have a great deal of activity in the higher frequencies. Lowpass filters, depending on their cutoff frequencies, attenuate high frequencies, so a bit of adjustment can alter a string sound considerably.

Next, press the ALG soft button to select the ALG page. The center DSP function block will already be selected. Set its value to "4POLE LOWPASS W/ SEP." Notice how the sound changes. Press the more> soft button, and the three soft buttons that select the control input pages for the lowpass filter will appear. Press the F1 FRQ soft button.

The Coarse Adjust parameter will already be selected. Try a few different values for this parameter, to get a feel for its effect on the sound. Then set it to a value of G#3 208 Hz. Use the cursor buttons to select the Src1 parameter, and set its value to MPress (3, 3, ENTER). Cursor down once to the Depth parameter, and set it to a moderate value, like 3800 cents. Now try playing and applying pressure to the keys. (Obviously, this will have no effect if your MIDI controller doesn't send mono pressure.)

Now we're going to use an envelope to control the sweep of the filter. The K2500 has one envelope dedicated to shaping the output of the sound; this is called AMPENV. However, two other envelopes (ENV 2 and ENV3) can be assigned to control other parameters. You can assign AMPENV as a Control Source, but using ENV2 and ENV3 lets you assign separate envelopes to do different things.

Cursor down to the Src2 parameter, and set its value to ENV2 (1, 2, 1, ENTER). Cursor down to the DptCtl parameter and set its value to ON. Cursor down to the MaxDpt parameter, and set

its value to 5800 cents. As soon as you set a depth, you'll hear the envelope sweep the cutoff frequency. We'll adjust it further in a minute. The F1 FRQ page should now look like this:

```

Edit Prog:F1 FRQ(4P LOPASS) >Layer:1/1
Coarse:G#3 208Hz Src1 :MPress
Fine :0ct Depth :3800ct
Src2 :ENV2
KeyTrk:0ct/key DptCtl:ON
VelTrk:0ct MinDpt:0ct
Pad :0dB MaxDpt:5800ct
<more F1 FRQ F2 RES F3 SEP F4 AMP more>

```

Next, press the F2 RES soft button to select the page for adjusting the resonance of the lowpass filter. Select the Src1 parameter, and set its value to Data (6, ENTER). Select the Depth parameter and set its value to 30.0 dB. This means that the partials at the cutoff frequency will be boosted 30 dB. The F2 RES page should look like the following diagram.

If your MIDI controller has a data slider (or a control that can be programmed to send MIDI 06), the data slider (or programmable control) will now affect the resonance of the filter. Be aware though that the change in resonance is subtle; resonance affects some waveforms more noticeably than others.

```

Edit Prog:F2 RES(4P LOPASS) >Layer:1/1
Adjust: 0.0dB Src1 :Data
Depth :30.0dB
Src2 :OFF
KeyTrk:0.00dB/key DptCtl:MWheel
VelTrk:0.00dB MinDpt:0.00dB
MaxDpt:0.00dB
<more F1 FRQ F2 RES F3 SEP F4 AMP more>

```


Next press either of the <more> soft buttons until you see the ENV2 soft button. Press it to select the ENV2 page. Here you'll program Envelope 2 to control the filter's cutoff frequency.

The Att1 time parameter will already be selected; set its value to 0.50 seconds. Press the right cursor button once to select the Att2 time parameter and set its value to 1.06 seconds. Cursor down once to select the Att2 level parameter and set its value to -1%. Cursor right once, then up once, selecting the Att3 time parameter. Set its value to 3.52 seconds. The ENV2 page should look like the diagram below. Play around with the sound a bit. Hold a few notes through the full length of ENV2; the sound will continue to change for several seconds. Work your controller's Mod Wheel and data slider through several different positions to get an idea of the range of variation you can add by using the two control sources in tandem.

```

EditProg:ENV2 [1/1] <>Layer:1/1
Att1:Att2:Att3:Dec1:Rel1:Rel2:Rel3:Loop:
0.50 1.06 3.52 0s 0s 0s 0s Off
100% -1% 100% 0% 0% 0% 0% Inf

```



```

<more AMPENV ENV2 ENV3 ENVCTL more>

```

## Example 3

### Sample and Hold; Using a FUN

This example will use one of the FUNs to create a sample and hold program. As usual, start with Program 199, and press EDIT. While you're on the ALG page, select a value of "PARAMETRIC EQ" for the center DSP function block. Then select the KEYMAP page, and select the keymap 152 Dull Sawtooth.

Now press the more> soft button, then the F1 FRQ soft button. This enables you to set the frequency for the parametric EQ, which will set the depth of the modulation for the sample and hold function. The Coarse parameter will already be selected; set its value to D#4 311 Hz. Cursor down to the KeyTrk parameter, and set its value to 100 cents per key. Cursor over to the Src1 parameter, and set its value to FUN1 (1, 1, 2, ENTER). Cursor down once to the Depth parameter, and set its value to a fairly large negative number, like -4000 cents.

Next, press the F3 AMP soft button, to set the amplitude for the Parametric EQ (make sure you don't select F4 FINAL AMP). Set the value of the Adjust parameter to about 17 dB.

Now press either <more> soft button until you see the LFO soft button. Press the LFO button to select the LFO page. We'll program LFO1 as a square wave, then use it as one of the FUN's inputs.

Select the MnRate parameter for LFO1 and set it to a value of 5.00 Hz. Select the MxRate parameter and set its value to 15.00 Hz. Select the RateCt parameter and set it to a value of Data. Select the Shape parameter and set its value to Square. That's it for the LFO page.

Next press the FUN soft button to select the FUN page. Select the Input A parameter for FUN1, and set its value to LFO1 (1, 1, 4, ENTER). Select the Input B parameter and set its value to RandV1 (1, 0, 8, ENTER). Finally, select the Function parameter, and set its value to "Sample B On A." Now when you play and hold a note, you'll hear the sample and hold effect. Use your MIDI controller's data slider (or a programmable control set to MIDI 06) to control the rate of the effect.

Here's what's happening. The square wave of the LFO cycles from +1 to -1. Every time this occurs, the random signal generator (RandV1) randomly picks a value which changes the frequency of the parametric EQ. There are lots of ways to set up sample and hold effects, but the FUN is the basic element. In this case, every time the value of the FUN's "A" input (LFO1) becomes greater than +.5, the value of the "B" input (RandV1) is sampled. That value becomes the FUN's output until the next time the value of the "A" input becomes greater than +.5 (see Chapter 16 for more). In this example, the FUN's output modulates the frequency of the EQ, and causes rapid changes in timbre.

Now we'll apply the FUN to control the pitch of the sound as well. Press either of the <more> soft buttons until the PITCH soft button is visible, and press it to select the PITCH page. Select the Src1 parameter and set its value to FUN1. Select the Depth parameter and set its value to about 400 cents. Now the sound's pitch will fluctuate in sync with the EQ effect.



## Example 4

### SHAPER and PANNER

Our next example incorporates two of the DSP functions, and will give you a general overview of using the algorithms to build sounds.

Starting with Program 199, press EDIT, and while you're on the ALG page, cursor up to the Algorithm parameter, and select Algorithm 13. Select values of NONE in the F1 and F2 blocks. The value of the F3 block is already set to PANNER, as shown below.



Next press the KEYMAP soft button to select the KEYMAP page, and select 163 Sine Wave as the keymap. Play a few notes to accustom your ear to the sound of the unshaped sine wave. Then return to the ALG page, and select SHAPER for the algorithm's F1 block. You'll notice a bit of an effect right away. The SHAPER DSP adds additional high frequency partials to the waveform in unpredictable ways, resulting in large changes in timbre. For more on the SHAPER, see Chapter 14.



Press the more> soft button, then press the F1 AMT soft button. This enables you to adjust the amount of shaping applied to the sine wave. Try different values for the Adjust parameter, then set it to its minimum—0.100x. Then cursor down to the VelTrk parameter, and set a value of 0.70. Listen for the variation in the effect as you play with different attack velocities.

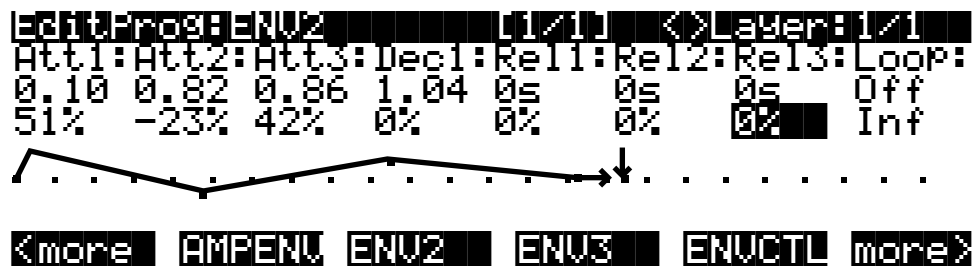
Now go back to the KEYMAP page, and select different keymaps so you can hear the SHAPER's effect on different sounds. When you're finished experimenting, set the Keymap parameter at 152 Dull Sawtooth.

Notice how notes on the high end tend to break up. If you return to the F1 AMT page you can reduce this distortion and/or *aliasing* with the KStart and KeyTrk parameters.

Select the KStart parameter and set a value of C 2 Unipolar. This limits the amount of shaping applied to notes above or below C 2, depending on the value of the KeyTrk parameter. Next set the value for the KeyTrk parameter to -0.018x / key. Since we're using a negative value, the amount of shaping will decrease with higher notes.

Next we'll program an envelope to change the SHAPER in realtime. While still on the F1 AMT page, select the Src2 parameter, and set it to a value of ENV2 (1, 2, 1, ENTER). Cursor down to the DptCtl parameter and set it to a value of AttVel (1, 0, 0, ENTER). Set the MinDpt parameter to 0.00x and the MaxDpt parameter to a value of 1.70 x. This will let you use attack velocity to control the amount of the envelope's effect on the SHAPER.

Next, use the <more> soft buttons to locate the ENV2 soft button, then press it to select the ENV2 page. Set up the parameters as shown in the diagram on the following page.



This can still be a little harsh on the high end when you play with high attack velocities. One way to smooth it out would be to go back to the ALG page, select a lowpass filter for the F2 block, and adjust its cutoff frequency to about F# 6. This is done by pressing EDIT when the F2 block is selected, then selecting the Adjust parameter and changing the value with any data entry method.



That's it for the SHAPER example; we'll continue with this program to describe the PANNER. You'll notice that in Algorithm 13, PANNER is the only value available for the F3 block. The PANNER function takes a single signal from the sound engine, and splits it into two. These are referred to as the upper and lower wires. The upper and lower wires pass independently into the final Amp block, and from there to the audio outputs.

The parameters on the control input page for the PANNER let you distribute the signal between the upper and lower wires. You can send the signal all to the lower wire (an Adjust value of -100%), all to the upper wire (100%), or anywhere in between. This in itself won't necessarily change the pan position of the current layer. It works in tandem with the Pan parameter on the layer's OUTPUT page.

When a layer uses an algorithm that contains the PANNER function, you always have two wires going through the final Amp and to the audio outputs. Consequently, on the layer's OUTPUT page, there are parameters to assign the output pair and pan position of each wire. When you have one wire panned hard left, and the other hard right, changing the parameters on the PANNER control input page will enable you to move the layer's pan position in realtime. The closer a layer's output is to the center of the stereo field, the less effective the PANNER function will be.

The first step in our PANNER example, therefore, will be to select the OUTPUT page. Select the Pan parameter for the upper wire, and set it all the way to the right. Select the Pan parameter for the lower wire and set it all the way to the left.

Now you can select the F3 POS page, and program it to move the sound around. First try changing the value of the Adjust parameter. You should hear the sound move to the left when the value is negative, and to the right when it's positive. Set it back to 0%, then select the Src1 parameter. Select LFO1 as the value, then select the Depth parameter and set it to 100%.

Now select the Src1 parameter again, and press EDIT (or use the more soft button locate the LFO button, then press it). You will now see the LFO page. Set the MnRate parameter for LFO1 to a value of 0.1 Hz. Set the MxRate parameter to a value of 2.00 Hz, and the RateCt parameter to Data. Leave the Shape parameter set to Sine. You should hear the sound shift slowly from left to right as the LFO cycles. You can adjust the speed of the shift with your MIDI controller's data slider (or a programmable control set to MIDI 06).

## Example 5

### Building a Drum Program; Using the Keymap Editor

With our next example, you'll learn how to build a drum program using the Program and Keymap Editors. To keep the example as brief as possible, we'll include only a few timbres and DSP examples. This won't make for a terribly realistic drum program, but it will give you the basic ideas you need to build your own. In this example, you'll create a four-layer program, with a different percussion timbre in each layer, each timbre having a different set of DSP functions applied.

Start with the default program 199. Press EDIT, then press the KEYMAP soft button. Select a value of 168 Silence. This gives you a keymap with a single key range from C 0 to G 10. Select the KeyTrk parameter, and change the value to 0. This will make the pitches of all the samples you assign the same on each key (you won't hear anything until you assign the samples). Next, press the <more soft button once, then press the DupLyr soft button. Layer 2 will be created. Repeat this twice, until you have four layers. If the K2500 tells you that you can't add any more layers, exit the Program Editor (don't bother to save at this point), check the current MIDI channel shown at the top right of the Program mode page, then go to Master mode and set the DrumChan parameter so that one of the eight available drum channels matches the current MIDI channel. Now you can start over, and the K2500 will allow you to add the fourth layer.

When you've created the four layers, you'll notice that the top line of the display tells you that you're looking at Layer 4 of a 4-layer program. Press the Up CHAN/BANK button to return to Layer 1. Press the LAYER soft button to select the LAYER page. Set the LoKey and HiKey parameters to C 4 and D 4 respectively. The easiest way to do this is to select the LoKey parameter, hold the ENTER button, and strike C 4 on your MIDI controller. Do the same for the HiKey parameter, striking D 4. Press the Up CHAN/BANK button to select Layer 2. Set its LoKey and HiKey to D# 4 and F 4, respectively. Repeat this for Layer 3, setting its LoKey and HiKey parameters to F# 4 and G# 4 respectively. Do the same for Layer 4, setting its LoKey and HiKey parameters to A 4 and B 4 respectively. This might be a good time to save what you've done so far.

Next, return to Layer 1 (CHAN/BANK button), press the KEYMAP soft button, select the Keymap parameter, and press EDIT to enter the Keymap Editor. Select the Sample parameter, and select a value of 20 14in Dry Tom-C 4. Press the Save soft button, and the K2500 will prompt you to save the keymap. Rename it as Tom, and save it. Press EXIT to return to the Program Editor, and select Layer 2 with the CHAN/BANK buttons. Press EDIT to return to the Keymap Editor, select the Sample parameter, and assign a value of 20 22in Dry Kick C 4. Save the keymap, renaming it Kick. Press EXIT to return to the Program Editor, and select Layer 3. Return to the Keymap Editor, select the Sample parameter, assign a value of 20 8in Dry Snare C 4, save the keymap—renaming it Snare. Return to the Program Editor, select Layer 4, select the Keymap parameter, and assign a value of 20 Vol Ft Cl Hat C 4. Save the Keymap, renaming it HiHat. You now have a 4-layer program, each layer having its own keymap with a different sample assigned to each one.

This is the basic process for creating any keymap and incorporating it into a program. In this case, we don't want the layers to overlap, and we want each layer to use a distinct keymap with its own sample assignment. In other programs, you might want to create a keymap with different timbres in a single layer, and you might want the layers to overlap.

As an example of how to quickly set up a multi-sample keymap, we'll change the sample assignment in Layer 1. Return to the Program Editor, and select Layer 1. Return to the Keymap Editor (by pressing EDIT while the Keymap parameter is highlighted on the EditProg\*KEYMAP page), then press the NewRng soft button. The K2500 will prompt you to strike a low and high key on your MIDI controller. Strike C 4 and D 4. Notice that the value of the Key Range parameter changes to reflect the new range assignment. Now select the sample

parameter, and turn the Alpha Wheel one click to the right, to select the sample 20 12in Dry Tom C 4. Save the keymap, replacing the earlier version. You can repeat this process to create as many new key ranges as you like (in this example, doing so would have no effect, since we've limited each layer to a narrow 3-key span).

If you wanted the layers to overlap, you would simply set each layer's LoKey and HiKey parameters (on the LAYER page in the Program Editor) to the same respective values. For example, you might set each LoKey parameter to C 2, and each HiKey parameter to C 7, causing all layers to play across five octaves.

Now we'll add some processing to some of the layers in our drum program example. The fact that each sound is on a different layer enables us to use a different algorithm for each layer, giving us enormous control over each sound.

Return to the Program Editor and select Layer 1. Press the PITCH soft button, and select the Coarse transpose parameter. Set its value to -3 ST. Select the KeyTrk parameter, and set its value to 400 cents per key. This will give you a different pitch on each key.

Next, select the PITCH page and select the Src2 parameter. Set its value to ENV2 (1, 2, 1, ENTER). Select the MaxDpt parameter and set its value to 300 cents. Select the DptCtl parameter, and set its value to VTRIG1 (1, 0, 6, ENTER). Press EDIT to select the VTRIG page, and set the VTrig1 Level parameter to a value of fff. Return to the PITCH page, select the Src2 parameter, and press EDIT to select the ENV2 page. Set it up as shown in the page below.

```

EditProg8:ENV2 [1/1] <>Layer:1/1
Att1:Att2:Att3:Dec1:Rel1:Rel2:Rel3:Loop:
0.02 0.16 0s 0.38 0s 0s 0s Off
28% -100 0% -76% 0% 0% 0% Inf

```

↓

```

<more> AMPENV ENV2 ENV3 ENVCTL more>

```

Next, select Layer 2 and select the F4 AMP page. Set the Adjust parameter to 8 dB to give the kick a little more presence. Select the Src1 parameter and set it to a value of AttVel (1, 0, 0, ENTER). Select the Depth parameter and set a value of 10 dB. The kick will get considerably louder as you strike the keys harder.

Select Layer 3, and press the ALG soft button. Select the center, 3-stage DSP function block and assign a value of HIFREQ STIMULATOR. Press the more> soft button once, then press the F1 FRQ soft button. Select the Coarse Adjust parameter, and assign a value of G 10 25088 Hz. Select the Src1 parameter and assign a value of MWheel. Select the Depth parameter and set it to a value of -10800 cents. Move your MIDI controller's Mod Wheel to bring out the snare's high end.

## Symbols

! AMP 14-50  
+ SHAPE MOD OSC 14-49

## Numerics

2PARAM SHAPER 14-45  
2POLE ALPASS 14-16  
2POLE LOWPASS 14-8  
32 track sequencing 12-46  
40335  
    heading 1  
        Controllers 7-19  
4POLE HIPASS W/SEP 14-13  
4POLE LOPASS W/SEP 14-10

## A

Aborting a Macro Load 13-54  
Aborting a Multiple File Load 13-21  
Aborting a sample recording 15-6  
Added waveforms 14-38  
Adding layers 6-39  
Adjust  
    Coarse 6-7  
    Fine 6-8  
Adjusting effects mix 9-2  
Adjusting sample volumes 15-39  
Aftertouch 11-2  
AIFF files 13-67  
AIFF-C files 13-67  
Akai 15-34  
Algorithm (ALG) page 6-10  
Algorithm output 6-5  
Algorithm programming 14-1  
Algorithm wiring 6-5  
All Controllers Off 10-4  
All Notes Off 10-4  
Allpass filters 14-15  
ALPASS 14-15  
Alpha Wheel 2-5, 3-4  
Alphabetic entry 3-4, 5-3  
Alphanumeric pad 3-4  
Alternative attack 6-16  
AMP  
    Adjust 6-19  
    Key tracking 6-19  
    Pad 6-20  
    Sources 1 and 2 6-20

    Velocity tracking 6-19  
Amp control 14-31  
AMP MOD 14-50  
AMP MOD OSC 14-49  
AMP U AMP L 14-32  
Amplitude envelope  
    Attack segment 6-29  
    Decay segment 6-29  
    Loop type 6-30  
    Natural 6-28  
    Number of loops 6-30  
    Release segment 6-29  
    User 6-28  
Amplitude envelope (AMPENV) page 6-28  
Append soft button 13-19  
Are you sure? dialog  
    suppressing with Confirm on Master page 11-3  
Arp Active 7-35  
Arpeggiator 7-33  
    Active 7-35  
    Beats 7-36  
    Duration 7-38  
    Glissando 7-38  
    Latch 7-35  
    Limit Option 7-37  
    Low Key (LoKey) and High Key (HiKey) 7-35  
    Note Shift 7-37  
    Order 7-36  
    Shift Limit 7-37  
    shifting notes 7-37  
    Tempo 7-36  
    Velocity 7-38  
ARRANGE Page  
    Song Mode 12-46  
Arrangement Feature  
    Edit Song Common parameters used with 12-26  
ASCII characters 5-3  
ASR page 6-36  
ASRs  
    Attack 6-36  
    Delay 6-36  
    Mode 6-36  
    Release 6-36  
    Trigger 6-36  
Assign sample 15-38  
Assigning samples to keymaps 15-39  
Attack portamento 6-26  
Attack segment 6-29  
Attack time for ASRs 6-36  
Attack velocity 11-2  
Audio cables 2-3  
Audio Configurations 18-1  
Audio output 6-21

Audio output groups and MIDI channels 18-4  
Auditioning Objects in RAM 13-25  
Auditioning Samples from a Disk File 13-17

## B

Backup 13-62  
Balance and Amp (BAL AMP) 14-33  
BAND2 14-20  
BANDPASS FILTER 14-18  
Bandpass filters 14-18  
Bank Select  
    MIDI Receive page 10-7  
Bank Status Indicator 13-18  
Basic editing 5-1  
Basic MIDI channel 10-4  
Bass tone control 14-27  
Battery Installation 2-2  
Battery replacement 2-2  
Beat Volume Adjust 15-32  
Bipolar control sources 16-1  
Bipolar keystart 14-4  
Boldface italic type 1-4  
BOOT.MAC 13-65  
Bottom line of display 3-2  
Bounce  
    Song Mode TRACK Page 12-35  
Building a Drum Program 19-9  
Building a keymap 15-41  
Bypass effects 5-6

## C

Cable 15-10  
Cables and Input Jacks 15-1  
Cancel  
    soft button 13-10  
Cancel soft button 13-19  
CD-ROM 13-1  
CHAN/BANK buttons 3-3, 5-6  
Chan/Bank buttons 5-6  
Change  
    Song Mode TRACK Page 12-39  
Changing intonation key 11-4  
Changing song playback programs 12-49  
Channel/Program (CH/PROG) Page 7-4  
    Setup Mode 7-2  
Channels on and off 10-8  
Chaotic LFOs (FUNs) 16-13  
Characters 5-3  
Choose File Name Function 13-37

Clear 15-23  
Clicking during portamento 6-26  
CLIP indicator  
    Sampling page 15-3  
Clock  
    Song mode 12-21  
Coarse adjust 6-7  
Common DSP control parameters 6-6  
COMMON Page  
    Song Mode 12-25  
COMMON page 6-25  
Compare 5-6  
Confirmations 11-3  
Connecting MIDI 2-3  
Connecting SCSI devices 13-2  
Control input pages 14-2  
Control inputs 6-5  
Control parameters 2-7  
Control Setup 6-3  
    Setup Editor 7-2  
Control sources  
    Unipolar vs Bipolar 16-1  
Control sources—Overview 16-1  
Controlling envelopes 6-32  
Controlling LFO rate 6-34  
Copy  
    disk mode 13-62  
    on Object Utilities page 11-8  
    Song Mode TRACK Page 12-34  
Copying layers 6-39  
Creating a Macro 13-41  
Creating Directories 13-11  
Creating new layers 6-39  
Crescendo/Decrescendo (Crescendo) 15-27  
Crossfade 6-22, 14-34  
    Orientation 6-22  
Crossfade and Volume Adjust Curves 15-33  
Crossfade Loop (XfadeLoop) 15-33  
Cursor buttons 3-2

## D

Data entry 3-4  
DC offset 14-45, 14-47, 14-48, 14-49, 14-51  
Decay segment 6-29  
Delayed vibrato 19-1  
Delaying ASRs 6-36  
Delete 15-24  
    on Object Utilities page 11-9  
Deleting Banks in a Startup File 13-65  
Deleting Files and Directories 13-61  
Deleting layers 6-40

- Deleting objects 5-4, 11-4
- Deleting programs 6-39
- Dependent objects
  - saving 13-26
- Dependents
  - loading 13-17
- Dialogs
  - Save 5-2
- Digital Outputs
  - using 15-11
- Diode equations (FUNs) 16-14
- Directories 13-4
  - creating 13-11
  - deleting 13-61
- Directory Selection Dialog 13-12
- Disk 5-6
- Disk button 5-6
- Disk Drive Information 13-5
- Disk format requirements 13-1
- Disk Function soft buttons 13-5
- Disk Mode 13-1
- Disk mode 4-4
- Disk Mode Functions 13-13
- Disk Utilities 13-56
- Display 3-1
  - Bottom line 3-2
  - Top line 3-2
- Display contrast 2-4, 11-3
- Distortion (DIST) 14-41
- Dither
  - Effects mode 9-5
- Double button presses 3-5
- Double notch filter 14-20
- DOUBLE NOTCH W/SEP 14-20
- Double shaper 14-45
- Double-output algorithm 6-5
- Double-output algorithms 6-23
- Drum channel 11-1
- Drum Loop
  - example of recording 12-3
- Drum programs 11-1
  - creating 19-9
- Drum Tracks
  - Song mode 12-27
- Drum trigger 15-4
- DSP control inputs 6-5
- DSP control parameters 6-6, 6-9, 14-2, 14-3
- DSP Functions 15-22
- DSP functions 14-5
  - inputs and outputs 6-5
- Dump
  - on Object Utilities page 11-10
- Dump program 6-39

- Duplicating layers 6-39
- Dynamic voice allocation 11-5

## E

- EDIT button 3-3
- Edit button 5-6
- Edit compare 5-6
- Edit Song
  - ARRANGE Page 12-46
  - COMMON Page 12-25
  - EVENT Page 12-28
  - STEP Page 12-43
  - TRACK Page 12-31
- Edit Song Pages 12-25
- Editing 5-1
- Editing Macros 13-50
- Editing Samples 15-12
- Effect changes when entering Effects mode 9-3
- EFFECT page 6-24
- Effect preset 6-24
- Effect Selection During Recording and Playback 12-50
- EffectChan
  - in Song mode 12-26
- Effects
  - selecting 9-3
  - Wet/Dry mix 6-24
- Effects button 5-6, 9-1
- Effects bypass 5-6
- Effects configurations 9-6
- Effects configurations and parameters 9-7
- Effects Editor 9-6
- Effects mix 9-2
- Effects mode 4-3, 9-1
  - parameters 9-2
  - Selecting effects 9-2
  - Soft buttons 9-1
- Effects return 18-3
- Electrical grounding 2-3
- Enable layer 6-12
- Enable MIDI channels 10-8
- Envelope control
  - Adjust 6-32
  - Key tracking 6-33
  - Source and depth 6-33
  - Velocity tracking 6-33
- Envelope control page 6-32
- Envelopes 19-4
- Envelopes 2 and 3 6-31
- EPS 15-35
- EQ
  - Bass tone control 14-27

- Mid-range parametric 14-26
- Parametric 14-22
- Steep bass tone control 14-29
- Treble tone control 14-28
- Equalization 14-22, 14-30
- Erase
  - Song Mode TRACK Page 12-34
- Erase soft button
  - Song mode 12-16
- Evaluation of FUNs 16-16
- EVENT Page
  - Song Mode 12-28
- Everything
  - loading objects as 13-20
- Everything files 13-23
- EXIT button 3-4
- Exit values 7-25
- Exporting data 13-67
- Extended program changes 10-11
- Extended sample loop 15-16
- Extensions
  - used by or accepted by the K2500 13-7
- Extra mono dry output 11-3

## F

- F1-F3 pages 6-18
- F4 AMP page 6-19
- File Backup 13-62
- File Copy 13-64
- File Formats
  - supported by K2500 13-67
- File Index 13-8
- File List Dialog 13-6
- Files
  - deleting 13-61
  - Everything 13-23
  - Loading 5-5, 13-13
  - Master 13-23
  - moving between directories 13-59
  - renaming 13-61
  - Saving 5-5
  - saving 13-22
  - Split 13-23
- Fill Mode 13-18
- Fill soft button 13-19
- Filters 14-5
  - Allpass 14-15
  - Bandpass 14-18
  - Double notch 14-20
  - Highpass 14-12
  - Lowpass 14-7

- Notch 14-17
- Twin peaks bandpass 14-21
- Filters—A review 14-6
- Find Files 13-56
- Finding objects 3-7
- Fine adjust 6-8
- Fine Hz 6-17
- Floppy disks 13-1
- Floppy disks, formatting 13-2
- Format 15-10
- Formatting floppy disks 13-2
- Formatting SCSI devices 13-4
- Four-pole highpass filter 14-13
- Four-pole lowpass filter 14-10
- Free Utility 13-58
- Freeze pedal 6-13
- Front panel MIDI 3-7
- Front panel navigation 3-1
- FUN diagram 16-3
- FUN equations 16-3
- FUN page 6-37
- Function soft buttons 6-39
- Function Soft Buttons in the Sample Editor 15-12
- FUNS
  - For sample and hold 19-6
- FUNs
  - Chaotic LFOs 16-13
  - Diode equations 16-14
  - Evaluation sequence 16-16
  - Sawtooth LFOs 16-13
  - Warp equations 16-10
- FUNS equation list 16-4, 16-15
- FX bypass 5-6
- FX channel 9-4
- FX mode 9-3

## G

- GAIN 14-34
- Gain 15-3
- Gated lowpass filter 14-12
- General synthesis concepts 14-1
- Glitches
  - with guitar or wind controller 11-5
- Global preset effect 9-5
- Globals 6-27
- Glossary entries 1-4
- Grab
  - Song Mode TRACK Page 12-41
- Grounding 2-3
- Guitar/Wind Controller Mode 11-5



## H

- Hard reset 11-5
- Hard sync functions 14-51
  - SYNC M and SYNC S 14-51
- Hardware audio switching 18-3
- High frequency stimulator 14-39
- High key 6-11
- High velocity parameter 6-11
- Highpass filters 14-12
- HIPAS2 14-13
- HIPASS 14-12
- Hold through attack 6-13
- Hold until decay 6-13

## I

- Ignore release 6-13
- Impact
  - ENVCTL page 6-33
- Importing data 13-67
- Importing layers 6-40
- Input
  - Sample mode page 15-3
- Input Quantize 12-20
- Insert 15-26
  - Song Mode TRACK Page 12-35
  - with Macros 13-53
- Insert cables 18-3
- Insert Zero 15-25
- Installation 2-2
- Intonation key 11-4
- Intonation Table Editor 17-1
- Intonation tables 11-2
- Introduction to editing 5-1
- Intuitive data entry 3-6
- Invert 15-25

## J

- Jump to page 5-6

## K

- K1000
  - bank mode 7-8
- K2000
  - bank mode 7-8
- K2500 Features 1-2

- K2500 Overview 1-2
- K2500R Only 7-2
- Key range 15-38
- Key tracking 6-8, 6-14, 6-19, 6-33
- Key/Velocity (KEY/VEL) Page 7-9
- Keyboard
  - using to trigger sampler 15-5
- Keymap 15-41
  - Key tracking 6-14
  - Stereo 6-15
  - Transpose 6-14
  - Velocity tracking 6-14
- Keymap Editor 15-36, 19-9
  - Assign sample 15-38
  - New range 15-38
- Keymap Editor Parameters 15-38
- KEYMAP page 6-14
- Keymap parameter 6-14
- Keymaps 6-2
  - multi-velocity 15-40
- Keystart
  - bipolar 14-4
  - unipolar 14-4
- Keytrack start (KStart) 14-3
- KRZ files
  - saving and loading a macro table in 13-54
- Kurzweil program changes 10-13

## L

- Large disk files 13-23
- Layer
  - Delete 6-40
  - Duplicate 6-39
  - Hold through attack 6-13
  - Hold until decay 6-13
  - Ignore release 6-13
  - Import 6-40
  - new 6-39
  - Opaque 6-13
  - Set range 6-39
- Layer delay
  - Maximum 6-12
  - Minimum 6-12
- Layer delay control 6-12
- Layer enable 6-12
- Layer enable sense 6-12
- Layers
  - moving between in multi-layer programs 3-6
  - Muting 5-6
- LCD 3-1, 11-3
- Legato play 6-26

- LF SAW 14-37
- LF SIN 14-37
- LF SQR 14-37
- LFO page 6-34
- LFOs 6-34
  - Maximum rate 6-34
  - Minimum rate 6-34
  - Phase 6-35
  - Rate control 6-34
  - Shape 6-35
- Library 13-5
- Library Disk 13-47
- Line cord 2-3
- List Utility 13-58
- Load Function Dialog 13-18
- Loading a Macro File 13-43
- Loading Dependents of Selected Objects 13-17
- Loading files 5-5, 13-13
- Loading Individual Objects 13-13
- Loading Objects from Floppy Disk Files 13-18
- Loading Older Setup Versions 7-2
- Loading Selected Entries from a Macro File 13-49
- Loading Songs From Disk 12-51
- Local control 10-2
- Local Keyboard Channel 2-5
  - with Song Mode 12-1
- Lock parameters 10-9
- LOOP 15-18
- Looping amplitude envelopes 6-30
- LOPAS2 14-9
- LOPASS 14-7
- Low frequency waveforms
  - Coarse adjust 14-37
  - Sawtooth 14-37
  - Sine 14-37
  - Square 14-37
- Low key 6-11
- Low velocity parameter 6-11
- Lowpass filter with clipping 14-47
- Lowpass filters 19-4
- LP2RES 14-10
- LPCLIP 14-47
- LPGATE 14-12

## M

- Macro Insert 13-53
- Macro Load
  - aborting 13-54
  - remote 13-55
- Macro On Indicator 13-5
- Macros 13-38

- editing 13-50
- Mark pages 5-6
- Master button 5-6
- Master files 13-23
- Master mode 4-4
- Master mode page 11-1
- Master mode—Soft buttons 11-4
- Master panning 11-3
- Master Transpose 11-1
- Maximum layer delay 6-12
- Maximum LFO rate 6-34
- Memory banks 5-5
- Memory display 11-4
- Memory Limits 12-51
- Merge soft button 13-19
- Microtonal tunings 11-2
- MIDI
  - All Notes Off 10-4
  - Audio output gain 10-9
  - Audio output groups 18-4
  - Audio output pair 10-9
  - Basic channel 10-4
  - Channel enable 10-8
  - LED 3-3
  - Pan 10-8
  - Parameter locks 10-9
  - Program change formats 10-10
  - Program changes 10-13
  - Receive mode 10-4
  - Reset channels 10-18
  - Sending from front panel 3-7
  - Song recording 12-52
  - Transmit parameters 10-1
  - Volume 10-9
- MIDI button 5-6
- MIDI channel 10-2
- MIDI channel parameters 10-8
- MIDI connections 2-3
- MIDI control 10-2
- MIDI control parameters 2-7
- MIDI Faders Page 3-8
- MIDI loop 10-2
- MIDI mode 4-4
  - Soft buttons 10-18
- MIDI program changes 10-3, 10-16
- MIDI Receive page 10-7
  - Power Mode 10-7
- MIDI Receive parameters 10-4
- MIDI Type 0 and Type 1 Files 13-68
- MIDI Type 0 or Type 1 Files 12-51
- MIDIScope 11-4
- Mid-range parametric EQ 14-26
- Minimum layer delay 6-12

- Minimum LFO rate 6-34
- MISC Page
  - Song Mode 12-18
- Miscellaneous (MISC) 15-15
- Mix 15-26
- Mix Beat 15-30
- MIX outputs 18-1
- MIX Page
  - Song Mode 12-23
- Mixdown Page 3-8
- Mixers 14-34
- Mixers with non-linear inputs 14-50
  - !AMP 14-50
  - AMP MOD 14-50
  - x AMP 14-50
  - x GAIN 14-50
- Mixing outboard effects with the K2500's 18-3
- Mode 15-4
- Mode buttons 3-1
- Mode Selection 3-1
- Mode selection 4-1
- Modes 4-1
  - Using 4-3
- Momentary
  - switch type 7-30
- Monitor (Mon) 15-5
- Mono sound systems 2-3
- Monophonic programs 6-25
- Move
  - on Object Utilities page 11-7
- Moving Files Between Directories 13-59
- MPressure 7-20
- MS-DOS File System Compatibility 13-66
- Multi-disk files 13-23
- Multiple Object Selector
  - entering selection criteria in 13-34
  - soft buttons 13-33
- Multiple Object Selector Page 13-32
- Multiple Selection of Files to Load 13-21
- Multi-velocity keymaps 15-40
- Mute
  - Layers 5-6
  - Setup zones 5-6
- Muting tracks 12-24

## N

- Name
  - on Object Utilities page 11-9
- Name Table 13-27
  - not loading 13-31
- Naming objects 5-2

- Natural amplitude envelope 6-28
- Navigation 3-1
- Nested editors 4-2
- New layer 6-39
- Noise generator 6-16
- Non-linear functions 14-38
  - 2PARAM SHAPER 14-45
  - DIST 14-41
  - HIFREQ STIMULATOR 14-39
  - LPCLIP 14-47
  - PWM 14-47
  - SHAPE2 14-45
  - SHAPER 14-43
  - WRAP 14-46
- Normalize 15-22
- Notch filters 14-17
- NOTCH2 14-18
- Note triggering 6-12
- Numeric entry 3-4

## O

- Object type and ID 5-2
- Object Utilities 11-6
  - accessing from the editor 11-10
- Objects 5-1
  - auditioning 13-25
  - Deleting 5-4, 11-4
  - loading individually 13-13
  - Naming 5-2
  - RAM 5-2, 5-4
  - Renaming 5-3
  - ROM 5-2, 5-4
  - Saving 5-2
    - saving individually 13-24
    - soft button on Utility page 11-4
- Octave buttons don't work 6-4
- OK
  - soft button 13-10
- One-pole allpass filter 14-15
- One-pole highpass filter 14-12
- One-pole lowpass filter 14-7
- Opaque layer 6-13
- Open
  - soft button 13-9
- Output
  - Assignment and control 6-21
  - Double-wire algorithms 6-23
  - Gain 6-22
  - Pan mode 6-22
  - Panning 6-22
  - Selecting output group 6-22

- Stereo keymaps 6-23
- Output gain 10-9
- OUTPUT page 6-21
- Output pair 10-9
- Overdubs
  - creating with sampling option 15-8
- Overview 1-2
- Overwrt soft button 13-19
- OvFill soft button 13-19

## P

- Pad 6-20, 14-3
- Page Buttons 15-14
- Pages 3-2
  - Algorithm (ALG) 6-10
  - Amplitude envelope (AMPENV) 6-28
  - ASR 6-36
  - COMMON 6-25
  - EFFECT 6-24
  - Effects mode 9-1
  - Envelope control 6-32
  - Envelopes 2 and 3 6-31
  - F1-F3 6-18
  - F4 AMP 6-19
  - FUN 6-37
  - Jumping to 5-6
  - KEYMAP 6-14
  - LAYER 6-11
  - LFO 6-34
  - Marked 5-6
  - Master mode 11-1
  - MIDI CHANLS 10-8
  - MIDI RECV 10-4
  - MIDI XMIT 10-1
  - OUTPUT 6-21
  - PITCH 6-7, 6-17
  - Previous 5-6
  - VTRIG 6-38
- Pan 6-22
  - Master 11-3
  - MIDI 10-8
- Pan lock 10-9
- Pan Position
  - MIX page 12-23
- Panel play feature 3-7
- Panic button 2-6, 10-18
- PANNER 14-32, 19-7
- PARA BASS 14-27
- PARA MID 14-26
- PARA TREBLE 14-28
- Parameter locks 10-9

- Parametric EQ 14-22
- Parent
  - soft button 13-9
- Parentheses around program name 6-3, 11-2
- Patch Through
  - with Song mode 12-1
- Path 13-4
- PCH 10-12
- Pedals 7-44
- Performance controls 2-7
- Phase of LFO waveforms 6-35
- Pitch
  - Fine Hz 6-17
- Pitch bend mode 6-12
- Pitch bend range 6-25
- Pitch control 14-31
- PITCH page 6-7, 6-17
- Pitch Shift 15-29
- Pitch, amplitude, panner 14-31
- Playback mode 6-16
- Playing notes from front panel 3-7
- Playing songs 12-49
- Plus/Minus buttons 3-4
- Polyphony 6-2, 11-4
- Portamento 6-26
- Portamento click 6-26
- Portamento rate 6-26
- Power cable 2-3
- Power Mode
  - MIDI Receive page 10-7
- Preset effect 6-24
- Pressure (Press) Page
  - Setup Mode 7-29
- Pressure Map Editor 17-6
- Pressure maps 10-3, 10-5
- Pressure sensitivity 11-2
- Previous page 5-6
- Program button 5-6
- Program change formats 10-10
- Program change type 10-5
- Program changes
  - Extended 10-11
  - Kurzweil 10-13
  - MIDI 10-3, 10-13, 10-16
  - Quick Access banks 10-13, 10-16
  - Quick Access mode 8-1
- Program Editor
  - Pages 6-10
  - Soft buttons 6-5
- Program lock 10-9
- Program mode 2-4, 4-3, 6-1, 6-2
  - Soft buttons 6-4
- Program mode page 6-2

- Program structure 6-2
- Program, monophonic 6-25
- Programming algorithms 14-1
- Programming FUNs 16-2
- Programs 2-4
  - Deleting 6-39
  - SysEx dumps 6-39
- Programs not playing 11-2
- Punctuation 5-3
- PWM 14-47

## Q

- QA 0-127 10-17
- Quantize
  - Input 12-20
  - Song Mode TRACK Page 12-36
- Quick Access bank program changes 10-13, 10-16
- Quick Access button 5-6
- Quick Access Editor 8-1
- Quick Access mode 2-6, 4-3, 8-1
- Quick Mute
  - Song Mode MIX Page 12-24
- Quick Mute Track Bank Indicator
  - MIX page 12-24
- Quick Scrolling to Subdirectories 13-10

## R

- RAM objects 5-2, 5-4
- Rate 15-3
- Rate control of LFOs 6-34
- Reading Samples 15-34
- Realtime effects 6-24
- Recording Multi-timbral Sequences via MIDI 12-52
- Recording songs via MIDI 12-52
- Reference Quantize
  - Song Mode TRACK Page 12-37
- Region/Criteria window
  - Song mode 12-31
- Release segment 6-29
- Release time for ASRs 6-36
- Releasing ASRs 6-36
- Relink-by-Name 13-28
- Remap
  - Song Mode TRACK Page 12-41
- Removing layers 6-40
- Renaming Files 13-61
- Renaming objects 5-3
  - with Name utility 11-9
- Repeating ASRs 6-36

- Resample 15-28
- Reset
  - hard 11-5
- Reset MIDI channels 10-18
- Reverse 15-24
- Reverse samples 6-16
- Ribbon Configuration (RIBCFG) Page
  - Setup Mode 7-39
- RIBBON Page
  - Setup Mode 7-39
- Ribbons
  - Center 7-40
  - Large Ribbon Configuration 7-39
  - Position Mode (PosMode) 7-39
  - Ribbon Configuration (RIBCFG) Page 7-39
  - Spring 7-40
- Roland 15-35
- ROM objects 5-2, 5-4
- Root
  - soft button 13-9

## S

- Safety precautions 2-2
- Sample 15-5
  - Playback loops 6-16
  - Playback mode 6-16
  - Stereo 6-15
- Sample and Hold 19-6
- Sample and hold 19-6
- Sample and Hold with FUNs 19-6
- Sample Editor 15-41
- Sample recording
  - aborting 15-6
- Sample Skipping 6-15
- Sampler
  - entering from Master Mode 11-5
  - triggering from keyboard 15-5
- Samples
  - Adjusting volume 15-39
  - Assigning to keymaps 15-39
  - triggering from analog inputs 15-42
  - triggering from audio signals 15-4
  - Tuning 15-39
- Sampling Analog Signals 15-2
- Save dialog 5-2
- Saving and naming 5-2
- Saving Dependent Objects 13-26
- Saving Files 13-22
- Saving files 5-5
- Saving Individual Objects 13-24
- Saving Master and Everything Files 13-23

- Saving objects 5-2
  - RAM 5-2, 5-4
  - ROM 5-2, 5-4
- Saving RAM objects 5-2, 5-4
- Saving ROM objects 5-2, 5-4
- SAW 14-36
- Sawtooth LFOs (FUNs) 16-13
- SCSI 1-2
- SCSI connections 2-4
- SCSI devices 13-1
  - Formatting 13-4
  - Termination 13-3
- SCSI ID 10-5
- SCSI IDs 13-4
- SCSI Musical Data Interchange 1-2
- Search function 3-7
- Select soft button 13-8
- Selecting a Song for Playback 12-49
- Selecting effects 9-3
- Selecting modes 4-1
- Selecting parameters 3-1
- Selection by Dependents 13-36
- Selection by Type/Range 13-35
- Separate outputs 18-2
- Sequencer
  - tutorial 12-1
- Setting layer ranges 6-39
- Setting Up For Sampling 15-1
- Setup button 5-6
- Setup control parameters 2-7
- Setup Editor 7-4
  - Clear Soft Button 7-42
  - Control Setup 7-2
  - Copy and Paste Soft Buttons 7-42
  - Delete Soft Button 7-41
  - Delete Zone (DelZn) Soft Button 7-42
  - Dump Soft Button 7-41
  - Duplicate Zone (DupZn) Soft Button 7-41
  - Import Zone (ImpZn) Soft Button 7-41
  - Local Program (LocalPrg) 7-4
  - Name Soft Button 7-41
  - New Zone (NewZn) Soft Button 7-41
  - Set Range (SetRng) Soft Button 7-43
  - Soft buttons 7-41
- Setup Mode
  - ARPEG Page 7-33
  - Channel/Program (CH/PROG) Page 7-2
  - Pressure (Press) Page 7-29
  - Ribbon Configuration (RIBCFG) Page 7-39
  - RIBBON Page 7-39
  - WHEEL Page 7-28
- Setup mode 4-3, 7-1
- Setups 2-5
- Aux Bend 1 7-17
- Aux Bend 2 7-18
- BEND Page 7-17
- Bend Range 7-17
- COMMON Page 7-32
- Continuous Control Pedal (CPEDAL) Page 7-26
- Continuous Controller Parameters 7-24
- Controllers 7-19
- Curve (Curv) 7-24
- Destination 7-7, 7-30
- Destination (Dest) 7-24
- Editing Hints and Suggestions 7-44
- EFFECT 7-32
- Effects Mix (FXMix) 7-33
- Entry (Ent) and Exit States 7-31
- Entry Pan, Exit Pan 7-16
- Entry Program Change (EntryProgChg) 7-8
- Entry values 7-25
- Entry Volume, Exit Volume 7-16
- Footswitch (FOOTSW) Page 7-31
- Low and high key 7-9
- Low Velocity (LoVel), HighVelocity (HiVel) 7-15
- MIDI Bank 7-5
- MIDI Bank Mode 7-7
- MIDI channel 7-4
- MIDI Control Source List 7-21
- MIDI Program (MIDIProg) 7-5
- Muting zones 5-6
- Note Maps 7-10
- Off Value 7-31
- Offset (Add) 7-24
- On Value 7-30
- Pan/Volume (PAN/VOL) Page 7-16
- physical controllers 7-19
- RIBBON Page 7-27
- Save Soft Button 7-41
- Scale 7-24
- Selecting via MIDI 10-13, 10-16
- Silent Zones 7-7
- SLIDER and SLID/2 Pages 7-25
- SONG 7-33
- Status 7-5
- Switch controllers 7-30
- SWITCH Page 7-32
- Switch Type (SwType) 7-30
- Transpose 7-9
- Transposing 7-1
- Velocity Curve 7-13
- Velocity Offset 7-11
- Velocity Scale (VelScale) 7-11
- Zone Arpeggiation (ZoneArpeg) 7-8
- SHAPE MOD OSC 14-49
- SHAPE2 14-45

- SHAPER 14-43, 19-7
- Shift
  - Song Mode TRACK Page 12-38
- Shifting notes
  - in the Arpeggiator 7-37
- Signal-to-noise ratio 2-4
- SIMMs 1-2
- SINE 14-36
- Single-output algorithm 6-5
- Sleep soft button 13-6
- Sliders 7-20
- SMDI 1-2
- Soft buttons 3-2, 6-4, 6-5
  - Disk Mode page 13-5
  - Effects mode 9-1
  - Keymap Editor 15-38
  - Master mode 11-4
  - MIDI mode 10-18
  - Setup Editor 7-41
  - Special functions 6-39
- Soft Buttons on the DSP Page 15-21
- Soft Through
  - with Song Mode 12-1
- Software Upgrades 2-10
- Soloing current track 12-25
- Song button 5-6
- Song Mode 12-1
  - MISC Page 12-18
  - MIX Page 12-23
- Song mode 4-4
- Song playback 12-49
- Songs
  - Loading 12-51
  - Memory limits 12-51
  - Recording via MIDI 12-52
  - Synchronizing 12-50
- Sostenuto pedal 6-13
- Source 1 6-9
- Source 2 6-9
- Special button functions 5-6
- Split files 13-23
- SQUARE 14-36
- Src parameter
  - Sample page 15-3
- Standard MIDI Files 13-68
- Startup 2-1, 13-5
- Startup File
  - creating 13-65
  - deleting banks in 13-65
- Stealer
  - soft button on Utility page 11-5
- Steep bass tone control 14-29
- STEP Page

- Song Mode 12-43
- Stereo insert cables 11-3, 18-3
- Stereo keymaps 6-15, 6-23
- Stereo samples 6-15
- Sustain not working 10-4
- Sustain pedal 6-13
- Sustaining ASRs 6-36
- SW+DST 14-48
- SW+SHP 14-49
- SYNC M and SYNC S 14-51
- Synchronizing songs 12-21, 12-50
- Synthesis—General concepts 14-1
- System Exclusive 10-3
- System exclusive 6-39
- System Exclusive dumps 11-10
- System Exclusive ID 10-5

## T

- Terminating SCSI devices 13-3
- The 7-33
- Thin
  - Song Mode TRACK Page 12-40
- Threshold (Thresh) 15-5
- Timbre shift 6-15
- Time 15-5
- Time Warp 15-29
- Toggle
  - switch type 7-30
- Top line of display 3-2
- Total size of all files 13-10
- Track mute 12-24
- TRACK Page
  - Song Mode 12-31
- Track Status Indicators
  - MIX page 12-23
- Transpose
  - Master 11-1
  - MIDI 10-2
  - Song Mode TRACK Page 12-39
- Transposing setups 7-1
- Treble tone control 14-28
- Triggering ASRs 6-36
- Triggering notes on startup 6-12
- Triggering samples from analog inputs 15-42
- Triggering sampler from keyboard 15-5
- Triggering samples from audio signals 15-4
- TRIM 15-17
- Truncate 15-22
- Tuning samples 15-39
- Tuning to other instruments 11-1
- Twin peaks bandpass filter 14-21

- Two-parameter shaper 14-45
- Two-pole allpass filter 14-16
- Two-pole bandpass filter 14-18
- Two-pole bandpass filter, fixed width 14-20
- Two-pole highpass filter 14-13
- Two-pole lowpass filter 14-8
- Two-pole notch filter 14-17
- Two-pole notch filter, fixed width 14-18

## U

- Unipolar control sources 16-1
- Unipolar keystart 14-4
- Upper and lower amp 14-32
- User amplitude envelope 6-28
- Using the Digital Outputs 15-11
- Using the MIX Outputs 18-1
- Using the modes 4-3
- Using the Velocity Map Editor 17-3
- Using your K2500 in a SCSI System 13-4
- Utilities 11-4

## V

- Variable Architecture Synthesis 1-2, 1-3
- VAST 1-2, 1-3
- Velocity crossover 15-40
- Velocity Map Editor 17-3
- Velocity maps 10-2, 10-5
- Velocity sensitivity 11-2
- Velocity stabs 19-1
- Velocity tracking 6-8, 6-14, 6-19, 6-33
- Velocity triggers 6-38
- Ventilation 2-2
- View soft button 11-5
- Viewing Selected Objects 13-25
- Voice allocation 11-4
- Voice channels 6-2
- Voltage levels 2-3
- Volume
  - MIDI 10-9
- Volume Adjust 15-23
- Volume Knob 2-7
- Volume Level
  - MIX page 12-23
- Volume lock 10-9
- Volume Ramp 15-27
- VTRIG page 6-38

## W

- Warp equations 16-10
- WAVE Files 13-68
- Waveform wraparound 14-46
- Waveforms 14-35
  - Sawtooth 14-36
  - Sine 14-36
  - Square 14-36
- Waveforms for LFOs 6-35
- Waveforms with non-linear functions 14-48
  - + SHAPE MOD OSC 14-49
  - AMP MOD OSC 14-49
  - SHAPE MOD OSC 14-49
  - SW+DIST 14-48
  - SW+SHP 14-49
  - x SHAPE MOD OSC 14-49
- Wet/Dry mix 6-24
- Wet/dry mix 9-2
- WHEEL Page
  - Setup Mode 7-28
- Wind controller
  - controlling K2500 with 11-5
- World-Wide Web 2-10
- WRAP 14-46

## X

- x AMP 14-50
- x GAIN 14-50
- x SHAPE MOD OSC 14-49
- XFADE 14-34
- XMIT page 10-1

## Y

- Young Chang Distributors x

## Z

- Zone button lights 7-7