

Chapter 11

Master Mode

Press the **Master** mode button to enter Master mode, which contains parameters affecting the K2600'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 K2600, the MIDI channel to be used for KB3 programs (explained below), and for several keyboard and programming adjustments. You can also enter the sampler from the Master-mode page.

```
MasterMode  Samples:8192K  Memory:121K
Tune       :0ct
Transpose  :05T
KB3Chan    :1
VelTouch   :1 Linear
PressTouch:1 Linear      Confirm  :On
Intonation:1 Equal      IntonaKey:C
Object Delete Util Sample View MAST2
```

Parameter	Range of Values	Default
Tune	± 100 cents	0
Transpose	± 60 semitones	0
KB3 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
Confirm	On, Off	On
IntonaKey	C-B	C

Tune

Adjusting the value of this parameter tunes every program in the K2600 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 getting in tune 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 K2600's notes, and not notes sent via MIDI.

Transpose

Like the Tune parameter above, Transpose affects every K2600 program, but not those notes sent to the MIDI Out port. You can adjust the MIDI transposition on the TRANSMIT page in MIDI mode.

KB3 Channel

The architecture of KB3 programs is completely different from the architecture of regular VAST programs. KB3 don't have layers containing keymaps and samples that get played with each keystroke. Instead, they have an array of oscillators that are constantly running, which requires a different kind of processing (and more of it). Consequently, a KB3 program won't run on a "normal" MIDI channel. It has to have a special channel that processes note information differently. And that's what the KB3 Channel parameter does.

There can be only one KB3 channel, although you can set it to be any of the 16 MIDI channels. It's like saying "OK, I want Channel 1 to be the channel that plays KB3 programs. All the other channels are for regular programs and other stuff."

Any program can play on the KB3 channel, but *KB3* programs can play *only* on the KB3 channel. If you're on a keyboard channel that's not the KB3 channel, and you select a KB3 program, the program's name appears in parentheses in the display, and you won't be able to trigger any notes on that channel. In this case the box at the left of the display reminds you which channel is the KB3 channel.

Velocity and Pressure Touch (VelTouch and PressTouch)

These two parameters are used by the keyboard model only, and have no function in the K2600R. If you are using a K2600R, leave these parameters 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 (which is on the MIDI-mode TRANSMIT 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 K2600 supplies you with 17 different intonation "tables" to choose from. (There are also a few extra "tables" 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 K2600'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 18, 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 14.

Determining the Version Number of Your ROM Objects (Intonation Tables 18–22)

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.mn**. This isn't really another intonation table. Rather, this is where the K2600 stores the version number of some 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. If you have more than one block of ROM objects installed, you'll see additional "tables," up to and including 22. And don't forget to return to your correct intonation table when you've checked the version numbers of your ROM objects.

List and Description of Intonation Tables

1	Equal	No detuning of any intervals. The standard for modern western music.
2	Classic Just	Tunings are defined based on the ratios of the frequencies between intervals. The original tuning of Classical European music.
3	Just 1 0 th	Similar to classic Just, but with the Dominant 7th flatted an additional 15 cents.
4	Harmonic	The perfect 4th, Tritone, and Dominant 7th are heavily flatted.
5	Just Harmonic	
6	Werkmeister	Named for its inventor, Andreas Werkmeister. It's fairly close to equal temperament, and was developed to enable transposition with less dissonance.
7	1/5th Comma	
8	1/4th Comma	
9	Indian Raga	Based on the tunings for traditional Indian music.
10	Arabic	Oriented toward the tunings of Mid-Eastern music.
11	BaliJava1	Based on the pentatonic scale of Balinese and Javanese music.
12	BaliJava2	A variation on 1Bali/Java, slightly more subtle overall.
13	BaliJava3	A more extreme variation.
14	Tibetan	Based on the Chinese pentatonic scale.
15	CarlosAlpha	Developed by Wendy Carlos, an innovator in microtonal tunings, this intonation table flats each interval increasingly, resulting in an octave with quarter-tone intervals.
16	Pyth/aug4	This is a Pythagorean tuning, based on the Greek pentatonic scale. The tritone is 12 cents sharp.
17	Pyth/dim5	This is a Pythagorean tuning, based on the Greek pentatonic scale. The tritone is 12 cents flat.
18–22	Obj vn.n	Not an intonation table; indicates version number of K2500 ROM objects.

In general, you should select a nonstandard intonation table when you're playing simple melodies (as opposed to chords) in a particular musical style. When you use intonation tables based on pentatonic scales, you'll normally play pentatonic scales to most accurately reproduce those styles. An excellent reference source for further study of alternative tunings is *Tuning In: Microtonality in Electronic Music*, by Scott R. Wilkinson.

Confirm

Confirmations are special displays that the K2600 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, but your margin of error is slimmer with confirmations turned off.

Intonation Key (IntonaKey)

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^\sharp will be a quartertone flat relative to equal intonation. If you change the intonation key to **D**, then D^\sharp will be a quartertone flat. If you use nonstandard intonations, you'll want to change the intonation key as you change the key you're playing in. If the Intonation parameter is set to **Equal**, changing IntonaKey has no effect.

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 K2600 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 K2600 from a sequencer, you could simply insert the appropriate note events anywhere in the sequence to change the intonation key.

The MAST2 Page

Press the MAST2 soft button to reach the MAST2 page:



The MAST2 page enables you to perform a hard reset of your instrument (this deletes everything in RAM, so be careful!), and to turn the vocoder feature on and off.

The Vocoder

Vocoding is a special feature that allows you to use an input signal to control another audio (slave) signal. Typically you would use a synthesizer for the input signal, although in fact you can use any sound source. You must have the sampling option to be able to use the vocoder.

Cables and Connections

Using the K2600 for Both Input and Slave Signals

You'll need an insert cable (Y cord) with a 1/4-inch stereo (Tip/Ring/Sleeve) plug (male) on one end and 2 mono jacks (female) on the other end. The right side mono jack should be 1/4-inch. The left side can be either 1/4-inch or XLR. (You will be plugging a Mic into the left side, so if the insert cable has a 1/4-inch jack, you'll need an adapter from XLR to 1/4-inch.)

1. Plug the stereo side of the insert cable into the 1/4-inch stereo Sample Input.
2. Connect a microphone to the left mono jack of the insert cable. It *must* be the *left* input.
3. Connect the B Right output of the K2600 to the right mono jack of the Insert cable.

Using the K2600 for Input Signal and External Source for Slave Signal

There are two setup methods for this configuration. Here's the first:

You'll need an insert cable (Y cord) with a 1/4-inch stereo (Tip/Ring/Sleeve) plug (male) on one end and 2 mono jacks (female) on the other end. The right side mono jack should be 1/4-inch. The left side can be either 1/4-inch or XLR. (You will be plugging a Mic into the left side, so if the insert cable has a 1/4-inch jack, you'll need an adapter from XLR to 1/4-inch.)

1. Plug the stereo side of the insert cable into the Stereo Analog Input of the sampler.
2. Connect a microphone to the left mono jack of the insert cable. It *must* be the *left* input.
3. Connect the output of your external sound source to the right mono jack of the insert cable.

The second method:

1. Connect a microphone into the left low impedance input (XLR) of the sampler.
2. Connect your external sound source to the right low impedance input (XLR) of the sampler.

Final Audio Output

You must have audio cables connected from the A outputs on the K2600 to your mixer or amp. Don't use the Mix outputs.

MIDI

If you have a keyboard model, and your external slave is a rack (or it is a keyboard but you want to use the K2600's keyboard to control the slave), connect a MIDI cable from the MIDI Out port of the K2600 to the MIDI In port of the slave.

Setting Up the K2600

1. Go to Sample mode (press the **Sample** soft button in Program, Setup, or Quick Access mode).
2. Set the Input parameter to a value of **Analog**.
3. Set the value of the Source (Src) parameter to **External (Ext)**.
4. Set the value of the Mode parameter to **LiveIn**.

5. Verify that mic signal is on the left side only. Adjust the Gain parameter as needed, to get a good signal level.
6. Verify that your sound source (either the K2600 or external source) is on the right side only.
7. Go the Effects-mode page and make sure that the FX Mode parameter is set to **Auto** and the FX Chan parameter is set to **Current**.

Enabling Vocoder Mode

1. Load the file VOCODER.K26 into any bank. It's on the disk called *K2600 Farm & Extras*. See Chapter 13 if you need help loading a file.
2. Go to Master mode.
3. Press the **MAST2** soft button.
4. Set the value of the Vocoder parameter to **On**.
5. Exit from Master mode (press any of the other Mode buttons).

Note that enabling the vocoder activates special software, which replaces the software used for the SHAPE2 and AMP MOD OSC functions in the F3 block of an algorithm. Therefore any programs that use SHAPE2 and AMP MOD OSC in the F3 block will sound different while the vocoder is active. Turning the Vocoder parameter Off will restore those DSP functions and disable vocoding.

Using the Vocoder

Go to Setup Mode and select one of the setups in the memory bank where you just loaded the vocoder file. If you are using an external sound source for your slave, choose the setup **Vocoder-ExtSlave**. If you are using the K2600 as the input source for the slave, then you can choose either **Vocoder-22 Band** or **Vocoder-20 Band**. The 22-band vocoder will allow you to play up to 4 voices of polyphony on the slave program; the 20-band vocoder will allow you to play up to 8 voices of polyphony on the slave program.

Play a note or chord on your keyboard and speak into the microphone. You should be able to hear what you are speaking, but the sound will be a string sound (assuming you are using the K2600 as the slave source), pitched to the note or chord you are playing.

If you have a keyboard model, try moving Sliders A, B, and C, and listen for changes in the sound. If you have a rack-mount model, you can send MIDI Controller numbers 6, 12, & 13 from your controller. Since the setups contain entry values for these sliders, you may have to move the slider across its full range before it begins to take effect.

Effects Issues and Output Issues

The studio assigned to the vocoder setups is configured in the following manner: If you are using the K2600 for the slave signal, the slave program (in zone 3) has its output assigned to KDFX-B, which is being routed to the FXBus2, with no effect. On the OUTPUT page in the Setup Editor, Output B is set to **FXBus2**, thereby sending the signal from the slave program to the B outputs and from B Right into the right side of the sample input.

The slave program has its output panned hard right within the program, so if you decide to try using a different slave program, you will probably want to edit the program itself to pan its output hard right, so you get 100% of the signal. You don't need to worry about setting the

output pair within the program, because the Out parameter on the CH/PRG page of the Setup Editor is set to KDFX-B in zone 3, thereby overriding any settings from within the program.

The vocoder programs themselves are assigned to KDFX-A, which is being routed to FXBus1. On the OUTPUT page in the Setup Editor, Output A is set to **Mix**. So the final output of the vocoder programs is run through the effect and then comes out the A Outs and the Mix Outs. Don't use the Mix audio outputs, however, or you'll hear the slave program along with the vocoder.

If you choose to change the effects, you may find it easier to edit the vocoder studio, and try changing the effects assigned to FXBus1, FXBus2, and AuxFX. But if you want to change to a different studio, you will need to make sure the following parameters are set correctly: on the FXBUS page, for FXBus2, set the Level parameters for both Aux and Mix to **Off**, and on the OUTPUT page, set Output B to **FXBus2**.

How Vocoding Works

A vocoder is a device that analyzes the time-varying audio spectrum of one signal (the master) and imposes that spectrum as a filter on a second signal (the slave.) The method we use is an emulation of the traditional analog technique involving banks of bandpass filters and envelope followers.

The master signal is what you send from the microphone, and the slave signal is what you send from an external synthesizer or other sound source, or a program from the K2600.

The master signal is sent to a number of bandpass filters in parallel. The center frequencies are spaced to cover the most useful frequencies. The lowest frequency filter is a low pass rather than a bandpass, which groups all low-frequency components together. Likewise, the highest filter is a high pass. The outputs of all these bandpass filters go into individual envelope followers, which detect the level of signal present in each band. The output of the envelope follower is then used as a control for the slave signal.

The slave signal is also sent to the same number of bandpass filters. These generally have the same center frequencies as the master bandpasses. The output signals from the slave bandpasses are multiplied, one by one, by the outputs of the envelope followers (from the master signal). The resulting products are all added together for the final output.

Since each band requires two layers (one for master and one for slave), the largest number of bands you can have for vocoding is 24. ($24 \times 2 = 48$, which is your maximum polyphony.) The programs in the Setup called **Vocoder-ExtSlave** use 24 bands. If you want to use the K2600 to generate your slave signal, then you have to use either the 22- or 20-band vocoder setups, which have fewer bands, and therefore leave 4 or 8 voices of polyphony available for the slave signal program.

Since 48 (or 44 or 40) layers are used, and a drum program has a maximum of 32 layers, we use two 24 (or 22 or 20) layer programs, on different MIDI channels, that are combined in a setup.

Each of the setups has 3 zones. In the 22- and 20-band vocoder setups, the first two zones are used for the vocoding programs and the third zone plays the internal program that is used for the slave signal. In **Vocoder-ExtSlave**, the third zone is set to transmit via MIDI only, on Channel 1. (This allows you to play your external sound source, if it is a rack, but won't play a K2600 internal program.)

Layers are grouped in pairs, with the master signal going to the first layer, and the slave to the second. All odd numbered layers are master and all even numbered layers are slave. If you look at the algorithms in the vocoding programs, you will see that the first two DSP blocks (after PITCH) of each layer are a bandpass filter (or low pass or hi pass filters for the first and last bands). The first layer then has a DSP called MASTER, while the second layer has a DSP called

SLAVE. These stages are then followed by an AMP stage. These DSP blocks perform the function of an envelope follower and gain multiplication.

The signal flows from the odd numbered (master) layer to its associated even numbered (Slave) layer (for example, from layer 1 to 2), which is something that does not happen in other algorithms. The low pass frequencies controlled by the third time slot for each layer set the response speed of the envelope follower. They are normally set to the same frequency. The master layer controls the frequency of one pole of low pass filtering, and the slave layer controls two more poles.

The AMP page on the master layer does nothing. There is no output from this layer, so any settings on the OUTPUT page don't matter. The slave layer's AMP page does do an actual amplitude control. The output pages for slave layers are active, and can be used to choose the output group and set the step panning.

All of the master layers use the LiveIn Left keymap and all of the slave layers use the LiveIn Right keymap. That is why you must plug the microphone into the left side of the sample input and the slave source into the right side.

As is always the case with Live mode, a note message is required in order for an incoming signal to be processed through VAST. Therefore, the two layers in the setup assigned to the vocoding programs have Pswitch2 set to generate a C4 with a velocity of 127, as soon as the setup is selected. That note remains on until you select a different setup. The setups are edited so that none of the notes on an 88 note keyboard are assigned to either of the two vocoding programs

Real-time Control of the Vocoding Programs

The most important control parameter is the envelope follower speed, set by the third time slot low pass parameters. These are set to C 6 on all the layers for the initial level. Slider A (MIDI 6) lowers the cutoff up to 8 octaves (9600 cents). Therefore, the higher you raise the slider, the slower the envelope follower speed. C 6, as a filter cutoff, has a time constant on the order of one millisecond. This is generally too fast. For best results, this should be lowered about 4 octaves to C 2 (half the range of the Data Slider), to a time constant of 16 milliseconds. Too slow and the vocoder will not respond to quick transients, like consonants, and too fast will result in a jittery sort of sound, as the envelopes follow every little fluctuation. At the fastest possible setting, the envelopes follow the master audio signal itself, and an extremely harsh intermodulation is heard between master and slave. The vocoder setups have an entry value of 64 for this slider, so when the setup is selected it is the equivalent of having the slider halfway up.

Slider B (MIDI 12) is used to control the width of the band pass filters (for all bands except the lowest and highest). The vocoder setups have an entry value of 10 for this slider, the equivalent of having the slider at the first dot above the bottom.

Slider C (MIDI 13) transposes the center frequencies of all the slave bandpasses upward together. It gives you the same result as pitch shifting the master signal up. Vocal formants will be munchkinized as you bring the slider up. The vocoder setups have an entry value of 0 for this slider, the equivalent of having the slider at the bottom.

Additional Notes and Programming Suggestions

The classic application of a vocoder is to make instrumental sounds talk/sing. The slave signal has to have a lot of high frequency content, or the consonants will not be heard clearly. However, there is no rule set in stone that you must speak words into the microphone. Using the vocoder just as a timbral control can be just as interesting. You can get very expressive results by using your voice to control a lead line, doing the articulation and filter control by talking, singing, or just making various vocal sounds. You can get some of the same types of results you would by using a breath controller. It's a little like having a 24-band graphic equalizer, but instead of controlling it with your hands, you use your voice.

Furthermore, you don't even have to use a microphone as the master. You can send a signal from anything else that has varied timbral content and get interesting results. For example, the master signal could be a drum loop or some other recorded sound that changes timbres regularly.

The analog sample inputs on the K2500 are line level, not mic level. This means you have to boost the gain on the sample page to get a good signal. But this also increases the general noise level of the input signal. If you have a mic preamp, or plug the mic into a mixing board before sending the signal to the K2600, you can lower the Gain parameter and start with a much cleaner signal. This is highly recommended.

In addition, you will find you get better results if you run the preamped mic signal into a compressor before sending it to the K2600. This can also be done for the slave signal. Using compressors will give you a much more even dynamic result, making it easier to play and control your sound. This is because the dynamic range of the master and slave signals is added together. For example, let's say both the master and slave signals have a dynamic range of 20 dB. The resulting signal will have a dynamic range of 40 dB, giving you a very wide range between the softest and loudest signals you can produce.

One way to improve intelligibility is to mix in a little of the master signal into the final audio output. This can be done in a couple of ways. If you run the mic into a mixer, you can split the signal, sending it both to the K2600 as well as to your final mix.

A second way is to include it in the vocoder program. You can do this by editing one of the programs in the 22- or 20-band vocoder setups. You would want to add a layer to the program (it doesn't matter which one of the two programs you edit). Set the Keymap for the layer to LiveIn L and choose Algorithm 1 with the DSP function set to **NONE**. You could then control the amount of the signal by editing the Adjust parameter on the F4 AMP page (or even assign a control source to vary the amount).

You could then try various algorithms and DSP functions to further modify the signal. Running the signal through a high pass DSP to emphasize vocal articulations is one obvious example. Just make sure that you don't use the SHAPE 2 or AMP MOD OSC DSP functions. In that case, the master signal won't be output.

If you are using the K2600 for the slave signal, try editing the slave vocoder program. A simple thing to try is to choose a different keymap. The AMPENV in this program has been set to User, with a lengthy decay, so you can even choose decaying sounds such as guitar, and get interesting results. And of course, you can choose other programs as the slave.

And of course, you should try making some of your own programs to use as a source. Just edit the setup and change the program in zone 3 to your new program. For example:

- Use an LFO to modulate the center frequencies of the slave bandpasses, or the master bandpasses.
- Try panning alternate bands of the slave layers to L and R to create a "fake stereo" program.
- Try different center frequencies from the ones used in the preset programs.
- Currently the center frequencies of the slave layers match the master layers. Try scrambling the slave frequencies relative to the master frequencies.
- If you are using the K2600 for the slave signal and need more polyphony, you can delete some of the layers in the vocoding programs. Make sure to delete matching sets of master and slave layers. You will probably want to readjust the frequencies and widths of the remaining layers accordingly.

More applications

Instead of using a microphone or other external source for your master, you could use the K2600 to generate *both* the master and slave signals. There are two ways you could set this up. You can either edit the setup to add another program on a 4th zone, or you could edit the slave source program to add more layers. Then split the keyboard so that one side plays the master zone/layers and the other side plays the slave zone/layers. On the OUTPUT page, make sure all the master layers are assigned to B and panned hard left and the slave layers assigned to B and panned hard right. You will then have to alter the wiring setup described at the beginning of this document so that the B Left jack is going to the left side of the stereo sample input.

If you edit width of the master layers so that they are extremely narrow, and set the frequencies to a specific scale pattern, then if you sing into the microphone, you will only hear sound as you sing the specific pitches in that scale.

If you edit the width of the slave layers so that they are extremely narrow, then you will get a very pure tonal sound, hearing only very specific pitches depending on the harmonic content of the master.

Another possibility for using very narrow width master layers: Edit the slave layers so that instead of using a series of bandpass filters, each slave layer uses different DSP functions in the F1 and F2 slots (remember that the F3 slot still needs to be set to LPCLIP in order for the vocoding function to work—you can change algorithms as long as the algorithm allows LPCLIP to be selected for the F3 slot). Now, if you sing various pitches, the slave signal will be played through the various corresponding VAST algorithms.

It is actually possible to use samples in RAM (or ROM) instead of the Live Mode In for either the master or slave signals (or even both of them). Just change the Keymap parameter on the KEYMAP Page. (Remember that you need to edit the Keymap parameter on all master and/or slave layers.) In this case, the keymap would be playing a single held sample, so you will want to use a looped sample. Loops with changing harmonic content will work best. The note used in the setups is C 4, so you would want the sample root at C 4 to hear it back without transposition. You will need to edit the layers, save the programs, and reselect the setup before you will hear the change. If both the master and slave layers call up samples in the unit, then as soon as you select the setup, you will hear sound without even touching the keyboard! You might want to assign a slider to the F4 AMP page on the slave layers to control the amount of output. If the master and slave layers are loops of slightly different lengths, then you will hear a continually changing sound that could appear to go in indefinitely without changing.

Continuing with the previous suggestion, you could set the slave layers to different keymaps, each layer assigned to a different sample loop. Edit the DSP functions on the slave layers so that F1 and F2 are set to **NONE**, or some other DSP function. Set the master layers to very narrow widths. Now, as your master signal changes frequencies you will hear different sample loops fading in and out.

The Soft Buttons in Master Mode

Object

This soft button brings up the object utilities. They're described beginning on page 11-13.

Delete

This soft button brings up the delete bank 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 K2600'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 11-13).

When you press the **Voices** soft button, the display shows the K2600's active voice channels as you play. Blocks of capital Xs in six columns of eight represent the 48 notes that the K2600 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, then there's a good chance that when voice stealing takes place an audible voice will be reallocated.

The Voices utility works a bit differently for KB3 programs. The K2600 uses one voice of polyphony for every two tone wheels in a KB3 program. In the Voices utility, the voices used by the tone wheels appear as Xs, meaning that the voices are used for the KB3 program. They don't get reallocated at any time, since they're always on, even if you're not playing any notes. Any voices not dedicated to a KB3 program behave normally. So if you have a setup that contains a KB3 program in one zone, and VAST programs in one or more other zones, you can monitor the voice allocation of the non-KB3 voices in the section of the display that isn't constantly filled with Xs.

Use the **Stealer** soft button to select a display that will show how the K2600 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 set the view mode to large type. Press it again to return to the normal view. The large-type view affects Program, Setup, and Quick Access modes. When you’re in these modes, program names, setup names, and Quick Access-bank entry names appear in large, easy-to-read type.

Sample

Press the **Sample** soft button to enter the K2600’s sampler. Refer to Chapter 14 for complete information on the sampler.

Reset

This button is on the MAST2 page. Press the **Reset** soft button if you want to return your K2600’s memory to the state it was in when you bought it.



***Caution!** The K2600 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 K2600 will return to the Program-mode page.*

Guitar/Wind Controller Mode

If you are using a wind controller or guitar controller with your K2600, 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 not get generated properly.

Consequently, the K2600 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’re hearing a glitch in the attack of notes from your guitar or wind controller, you should try setting your K2600 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 K2600 under some circumstances.

Restarting the K2600 or performing a soft reset disables Guitar / Wind Controller mode.

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 dialog:

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 *Saving Individual Objects* on page 13-26. Here is what the **Move** page 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
```

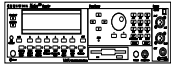
If you press **Cancel** while in one of the object utilities, you return to the Object Utilities page (the "Select database function:" dialog pictured above). Any objects that were selected when you pressed **Cancel** will still be selected if you subsequently enter a different object utility (by pressing a different soft button such as **Name**, for example). 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 13-36) 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, or song objects by pressing either the **Left** or **Right** cursor button, when the desired object is highlighted. Songs will play until either cursor button 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). K2600R owners also have the shortcut of holding the **Cancel** button, then pressing the alphanumeric buttons 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 example, 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#198
(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 button 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 400s 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 example, if you were moving Songs 300, 315, 489, and 841 to the 200s 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, Programs 200, 210, and 303. If you copy all of these objects at once into the 700s bank, using Fill mode, you will see the copies at Song 700 and at Programs 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 dialog prompting you for an object name, with a suggested default:

```
<>KbdNaming:Uff
Object Name:   Zither
Delete Insert <<< >>> OK Cancel
```

The default name that you see comes from the highlighted object in the object list, regardless of whether the highlighted object is actually selected. This makes it easy to copy the name of one object on to another.

As in all naming dialogs on the K2600, 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.

You can use the keyboard naming feature when naming objects. See page 5-5 for details. You can also use the string-replacement feature to make multiple changes to object names. See *Renaming Multiple Objects* on page 3-8 for details.

Delete

The Delete Objects utility is very useful for reclaiming unused object and sample RAM in your K2600. 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 K2600. 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 6 of the *Musician’s Reference* 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 K2600 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 K2600’s ROM sample area will reference the same area when you load them back in via MIDI.

Using the Object Utilities from the Editor

You can get to the object utilities 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:

```
editProgram: Save
```

```
Save Train Wreck as: ID#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 reentering 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.