

# ***K2661***

## ***Musician's Guide***

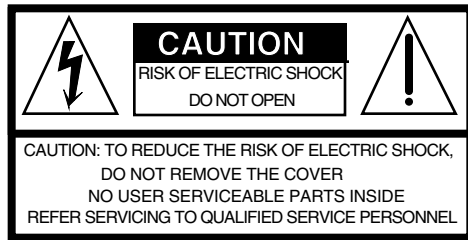


©2003 All rights reserved. Kurzweil® is a product line of Young Chang Co., Ltd. Young Chang®, Kurzweil®, V. A. S. T.®, KDFX®, Pitcher®, and LaserVerb®, KSP8™, K2661™, K2600™, K2500™, and K2000™ are trademarks of Young Chang Co., Ltd. SmartMedia™ is a trademark of Toshiba Corporation. ADAT® is a registered trademark of Alesis Corporation. All other products and brand names are trademarks or registered trademarks of their respective companies. Product features and specifications are subject to change without notice.

You may legally print up to two (2) copies of this document for personal use. Commercial use of any copies of this document is prohibited. Young Chang Co. retains ownership of all intellectual property represented by this document.

**Part Number: 910389 Rev. A**

---



The lightning flash with the arrowhead symbol, within an equilateral triangle, is intended to alert the user to the presence of uninsulated "dangerous voltage" within the product's enclosure that may be of sufficient magnitude to constitute a risk of electric shock to persons.



The exclamation point within an equilateral triangle is intended to alert the user to the presence of important operating and maintenance (servicing) instructions in the literature accompanying the product.

# IMPORTANT SAFETY & INSTALLATION INSTRUCTIONS

## INSTRUCTIONS PERTAINING TO THE RISK OF FIRE, ELECTRIC SHOCK, OR INJURY TO PERSONS

**WARNING:** When using electric products, basic precautions should always be followed, including the following:

1. Read all of the Safety and Installation Instructions and Explanation of Graphic Symbols before using the product.
2. This product must be grounded. If it should malfunction or break down, grounding provides a path of least resistance for electric current to reduce the risk of electric shock. This product is equipped with a power supply cord having an equipment-grounding conductor and a grounding plug. The plug must be plugged into an appropriate outlet which is properly installed and grounded in accordance with all local codes and ordinances.  
**DANGER:** Improper connection of the equipment-grounding conductor can result in a risk of electric shock. Do not modify the plug provided with the product - if it will not fit the outlet, have a proper outlet installed by a qualified electrician. Do not use an adaptor which defeats the function of the equipment-grounding conductor. If you are in doubt as to whether the product is properly grounded, check with a qualified serviceman or electrician.
3. **WARNING:** This product is equipped with an AC input voltage selector. The voltage selector has been factory set for the mains supply voltage in the country where this unit was sold. Changing the voltage selector may require the use of a different power supply cord or attachment plug, or both. To reduce the risk of fire or electric shock, refer servicing to qualified maintenance personnel.
4. Do not use this product near water - for example, near a bathtub, washbowl, kitchen sink, in a wet basement, or near a swimming pool, or the like.
5. This product should only be used with a stand or cart that is recommended by the manufacturer.
6. This product, either alone or in combination with an amplifier and speakers or headphones, may be capable of producing sound levels that could cause permanent hearing loss. Do not operate for a long period of time at a high volume level or at a level that is uncomfortable. If you experience any hearing loss or ringing in the ears, you should consult an audiologist.
7. The product should be located so that its location or position does not interfere with its proper ventilation.
8. The product should be located away from heat sources such as radiators, heat registers, or other products that produce heat.
9. The product should be connected to a power supply only of the type described in the operating instructions or as marked on the product.
10. This product may be equipped with a polarized line plug (one blade wider than the other). This is a safety feature. If you are unable to insert the plug into the outlet, contact an electrician to replace your obsolete outlet. Do not defeat the safety purpose of the plug.
11. The power supply cord of the product should be unplugged from the outlet when left unused for a long period of time. When unplugging the power supply cord, do not pull on the cord, but grasp it by the plug.
12. Care should be taken so that objects do not fall and liquids are not spilled into the enclosure through openings.
13. The product should be serviced by qualified service personnel when:
  - A. The power supply cord or the plug has been damaged;
  - B. Objects have fallen, or liquid has been spilled into the product;
  - C. The product has been exposed to rain;
  - D. The product does not appear to be operating normally or exhibits a marked change in performance;
  - E. The product has been dropped, or the enclosure damaged.
14. Do not attempt to service the product beyond that described in the user maintenance instructions. All other servicing should be referred to qualified service personnel.
15. **WARNING:** Do not place objects on the product's power supply cord, or place the product in a position where anyone could trip over, walk on, or roll anything over cords of any type. Do not allow the product to rest on or be installed over cords of any type. Improper installations of this type create the possibility of a fire hazard and/or personal injury.

## RADIO AND TELEVISION INTERFERENCE

**WARNING:** Changes or modifications to this instrument not expressly approved by Young Chang could void your authority to operate the instrument.

**IMPORTANT:** When connecting this product to accessories and/or other equipment use only high quality shielded cables.

**NOTE:** This instrument has been tested and found to comply with the limits for a Class B digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This instrument generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this instrument does cause harmful interference to radio or television reception, which can be determined by turning the instrument off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the instrument and the receiver.
- Connect the instrument into an outlet on a circuit other than the one to which the receiver is connected.
- If necessary consult your dealer or an experienced radio/television technician for additional suggestions.

### NOTICE

This apparatus does not exceed the Class B limits for radio noise emissions from digital apparatus set out in the Radio Interference Regulations of the Canadian Department of Communications.

### AVIS

Le présent appareil numérique n'émet pas de bruits radioélectriques dépassant les limites applicables aux appareils numériques de la class B prescrites dans le Règlement sur le brouillage radioélectrique édicté par le ministère des Communications du Canada.

## SAVE THESE INSTRUCTIONS

## Important Safety Instructions

- 1) Read these instructions
- 2) Keep these instructions.
- 3) Heed all warnings.
- 4) Follow all instructions.
- 5) Do not use this apparatus near water.
- 6) Clean only with dry cloth.
- 7) Do not block any of the ventilation openings. Install in accordance with the manufacturer's instructions.
- 8) Do not install near any heat sources such as radiators, heat registers, stoves, or other apparatus (including amplifiers) that produce heat.
- 9) Do not defeat the safety purpose of the polarized or grounding-type plug. A polarized plug has two blades with one wider than the other. A grounding type plug has two blades and a third grounding prong. The wide blade or the third prong are provided for your safety. If the provided plug does not fit into your outlet, consult an electrician for replacement of the obsolete outlet.
- 10) Protect the power cord from being walked on or pinched, particularly at plugs, convenience receptacles, and the point where they exit from the apparatus.
- 11) Only use attachments/accessories specified by the manufacturer.
- 12) Use only with a cart, stand, tripod, bracket, or table specified by the manufacturer, or sold with the apparatus. When a cart is used, use caution when moving the cart/apparatus combination to avoid injury from tip-over.
- 13) Unplug this apparatus during lightning storms or when unused for long periods of time.
- 14) Refer all servicing to qualified service personnel. Servicing is required when the apparatus has been damaged in any way, such as power-supply cord or plug is damaged, liquid has been spilled or objects have fallen into the apparatus, the apparatus has been exposed to rain or moisture, does not operate normally, or has been dropped.



**Warning-** To reduce the risk of fire or electric shock, do not expose this apparatus to rain or moisture. Do not expose this equipment to dripping or splashing and ensure that no objects filled with liquids, such as vases, are placed on the equipment.

To completely disconnect this equipment from the AC Mains, disconnect the power supply cord plug from the AC receptacle.

## **Kurzweil International Contacts**

Contact the nearest Kurzweil office listed below to locate your local Kurzweil representative.

Kurzweil Co., Ltd.  
Daerung Technotown 6th, 306  
493-6 Gasan, Gumcheon, Seoul, Korea  
Tel: (+82) 2-2108-5700  
Fax: (+82) 2-2108-5729

A N D Music Corp.  
P.O. Box 99995  
Lakewood, WA 98499-0995, USA  
Tel: (253) 589-3200  
Fax: (253) 984-0245

Young Chang Canada Corp.  
250 Victoria Park Ave. Suite # 105  
Toronto, Ontario Canada M2H 3P7  
Tel: (905) 948-8052

Team Kurzweil Europe  
Gl. Donsvej 8  
6000 Kolding  
Phone: (+45) 75 56 96 44  
Fax: (+45) 75 56 96 55

Official distributors in other countries are listed on the web site.

## **World Wide Web Home Page:**

<http://www.kurzweilmusicsystems.com>

# Contents

Kurzweil International Contacts .....	iv
World Wide Web Home Page: .....	iv

## Chapter 1 Introduction

Welcome! .....	1-1
Overview of the K2661 .....	1-1
VAST Synthesis.....	1-3
KB3 Tone Wheel Emulation .....	1-3
How the K2661 Works .....	1-3
How to Use This Manual .....	1-4
Do I Have Everything? .....	1-4
The Accessory Files .....	1-5
K2661 Base ROM Objects .....	1-5
K2661 Demos .....	1-5
K2661 Farm & Extras .....	1-5
K2600 Compatibility .....	1-6
K2500 Compatibility .....	1-6
K2000 Compatibility .....	1-6

## Chapter 2 Startup

Quick Startup Checklist .....	2-1
Start Jamming! .....	2-1
Startup—the Details .....	2-2
Before You Start... ..	2-2
Connecting the Power Cable (Line Cord).....	2-2
Connecting the Audio Cables.....	2-2
Connecting MIDI.....	2-2
Connecting SCSI.....	2-3
Switching On the Power .....	2-3
SmartMedia.....	2-3
Playing the Presets .....	2-4
Getting Around .....	2-4
Programs .....	2-4
Setups.....	2-6
Quick Access .....	2-7
The Other Modes.....	2-8
Software Upgrades.....	2-8

## Chapter 3 User Interface Basics

Mode Selection.....	3-1
Mode Buttons.....	3-2
Navigation.....	3-2
The Display .....	3-2
Pages .....	3-2
The Top Line .....	3-3
The Bottom Line .....	3-3
The Soft Buttons .....	3-3

---

The Cursor Buttons .....	3-3
The Chan/bank Buttons .....	3-3
The Edit Button .....	3-4
The Exit Button.....	3-4
Data Entry .....	3-4
The Alpha Wheel.....	3-4
The Plus/Minus Buttons.....	3-4
The Alphanumeric Pad .....	3-5
Double Button Presses.....	3-5
Intuitive Data Entry .....	3-6
Changing the Current Layer in Multi-Layer Programs.....	3-7
Search .....	3-7
Renaming Multiple Objects .....	3-7
Mixdown and MIDI Faders Pages.....	3-9
Mixdown Page.....	3-9
MIDI Faders Pages.....	3-9
Quick Song Recording and Playback .....	3-10

## **Chapter 4 The Operating Modes**

What the Modes Are .....	4-1
Selecting Modes.....	4-1
Nested Editors .....	4-3
Using the Modes.....	4-3
Program Mode.....	4-3
Setup Mode .....	4-3
Quick Access Mode.....	4-4
Effects Mode .....	4-4
MIDI Mode.....	4-4
Master Mode .....	4-4
Song Mode .....	4-4
Disk Mode .....	4-4

## **Chapter 5 Editing Conventions**

Introduction to Editing.....	5-1
What's an Object?.....	5-1
Object Type and ID.....	5-2
Saving and Naming .....	5-3
ROM Objects.....	5-4
RAM Objects.....	5-4
Keyboard Naming.....	5-5
Deleting Objects.....	5-6
Dependent Objects.....	5-6
Memory Banks.....	5-7
Saving and Loading Files—Disk Mode .....	5-7
Special Button Functions.....	5-8

## **Chapter 6 Program Mode and the Program Editor**

Background .....	6-1
VAST Program Structure.....	6-2
KB3 Program Structure.....	6-4

KB3 Mode .....	6-4
KB3 Channel .....	6-4
Real-time Controls in KB3 Mode .....	6-5
Playing KB3 Programs.....	6-5
KB3 Mode Buttons (Mute Buttons) .....	6-5
MIDI Control of KB3 Programs .....	6-6
The Program Mode Page.....	6-9
Program Names in Parentheses .....	6-9
Control Setup .....	6-10
The Soft Buttons in Program Mode .....	6-10
Editing VAST Programs .....	6-11
The Soft Buttons in the Program Editor.....	6-12
Algorithm Basics .....	6-12
Common DSP Control Parameters .....	6-14
Initial Setting Parameters.....	6-14
Hard-wired Parameters.....	6-14
Programmable Parameters .....	6-14
Coarse Adjust.....	6-15
Fine Adjust .....	6-15
Key Tracking .....	6-16
Velocity Tracking.....	6-16
Source 1 (Src1).....	6-16
Source 2 (Src2).....	6-16
Summary of Common DSP Control Parameters .....	6-17
The Algorithm (ALG) Page.....	6-17
The LAYER Page.....	6-19
Low Key (LoKey) .....	6-19
High Key (HiKey) .....	6-19
Low Velocity (LoVel).....	6-20
High Velocity (HiVel).....	6-20
Pitch Bend Mode (PBMode) .....	6-20
Trigger (Trig) .....	6-20
Delay Control (DlyCtl) .....	6-20
Minimum Delay (MinDly), Maximum Delay (MaxDly) .....	6-20
Enable .....	6-21
Enable Sense (S).....	6-21
Opaque .....	6-22
Sustain Pedal (SusPdl).....	6-22
Sostenuto Pedal (SosPdl) .....	6-22
Freeze Pedal (FrzPdl).....	6-22
Ignore Release (IgnRel) .....	6-22
Hold Through Attack (ThrAtt).....	6-22
Hold Until Decay (TilDec) .....	6-23
The KEYMAP Page .....	6-23
Keymap.....	6-23
Transpose (Xpose) .....	6-23
Key Tracking (KeyTrk) .....	6-24
Velocity Tracking (VelTrk).....	6-24
Sample Skipping (SmpSkp) .....	6-24
Stereo.....	6-24
Timbre Shift.....	6-25
Playback Mode .....	6-26

Alternative Switch (AltControl and AltMethod).....	6-26
The PITCH Page .....	6-27
Fine Hz.....	6-27
The F1–F3 Pages .....	6-27
The F4 AMP Page .....	6-28
Adjust.....	6-28
Key Tracking .....	6-28
Velocity Tracking .....	6-29
Pad.....	6-29
Source 1 and Source 2, Depth Controls.....	6-29
The OUTPUT Page.....	6-29
Pair .....	6-30
Pan.....	6-30
Mode .....	6-30
Gain.....	6-31
Crossfade, Crossfade Sense (XFadeSense) .....	6-31
Other OUTPUT Page Configurations .....	6-31
The COMMON Page .....	6-32
Pitch Bend Range .....	6-33
Monophonic.....	6-33
Legato Play.....	6-33
Portamento.....	6-33
Portamento Rate.....	6-34
Attack Portamento .....	6-34
Globals .....	6-34
OutPair .....	6-35
The Amplitude Envelope (AMPENV) Page.....	6-35
Attack Segment Times.....	6-37
Attack Segment Levels .....	6-37
Decay Segment .....	6-37
Release Segments .....	6-37
Loop Type.....	6-37
Number of Loops .....	6-38
The Envelope 2 (ENV2) and Envelope 3 (ENV3) Pages .....	6-38
The Envelope Control (ENVCTL) Page .....	6-39
Adjust.....	6-40
Key Tracking .....	6-40
Velocity Tracking .....	6-40
Source, Depth.....	6-40
Impact .....	6-40
The LFO Page.....	6-40
Minimum Rate.....	6-41
Maximum Rate .....	6-41
Rate Control .....	6-41
LFO Shape .....	6-41
LFO Phase .....	6-42
The ASR Page.....	6-42
Trigger.....	6-42
Mode .....	6-43
Delay .....	6-43
Attack.....	6-43
Release .....	6-43



The Function (FUN) Page .....	6-43
The Velocity Trigger (VTRIG) Page .....	6-44
The KDFX Page.....	6-44
The FXMOD2–FXMOD4 Pages .....	6-46
The FXLFO, FXASR, and FXFUN Pages .....	6-46
The ImportKDFX Page .....	6-46
Function Soft Buttons .....	6-47
Set Range (SetRng).....	6-47
Name.....	6-47
Save .....	6-47
Delete .....	6-47
Dump .....	6-47
New Layer (NewLyr).....	6-47
Duplicate Layer (DupLyr).....	6-48
Import Layer (ImpLyr) .....	6-48
Delete Layer (DelLyr) .....	6-48
Editing KB3 Programs .....	6-48
The TONEWL Page.....	6-48
Upper Tone Wheels.....	6-49
Lower Tone Wheels.....	6-49
Upper Volume Adjust.....	6-49
Number of Tone Wheels .....	6-49
Upper/Lower Swap .....	6-50
Wheel Volume Map .....	6-50
Organ Map .....	6-50
Lower Transpose / Upper Transpose .....	6-50
Lowest Pitch.....	6-50
The DRAWBR Page.....	6-51
Mode .....	6-51
Steps .....	6-51
Volume .....	6-51
Tune.....	6-51
The SetDBR Soft Button.....	6-51
The PITCH Page .....	6-52
The PERC Page .....	6-52
Percussion .....	6-52
Volume .....	6-53
Decay .....	6-53
Harmonic.....	6-53
VelTrack .....	6-53
LowHarm .....	6-53
HighHarm .....	6-53
StealBar .....	6-53
The PERC2 Page .....	6-54
PercLevel, DecayTime, OrgLevel.....	6-54
The KEYCLK Page .....	6-54
KeyClick .....	6-55
Volume .....	6-55
Decay .....	6-55
Pitch.....	6-55
VelTrk .....	6-55
Random .....	6-55

ReTrigThresh.....	6-55
Note Attack .....	6-56
Note Release .....	6-56
The AMP Page .....	6-56
The OUTPUT Page.....	6-56
The MISC Page .....	6-57
PreampResp .....	6-57
Leakage.....	6-57
LeakMode.....	6-57
SpeedCtl .....	6-58
VibChorCtl .....	6-58
VibChorSel .....	6-58
VolAdjust.....	6-58
BendRange .....	6-58
Sustain.....	6-58
Sostenuto .....	6-58
The EQ Page.....	6-59
All the Other Pages .....	6-59
Programming Tips.....	6-59
Using a KB3 Program in a Setup.....	6-60
Programs Using 2PARAM SHAPER .....	6-60
Shaper AMP (!AMP) .....	6-60
Using the VAST Program Editor on a KB3 Program .....	6-61

## Chapter 7 Setup Mode and the Setup Editor

Setup Mode .....	7-1
Loading Older Setup Versions .....	7-2
The Control Setup .....	7-2
Zone-status LEDs in Setup Mode .....	7-4
The Setup Editor.....	7-5
The Channel/Program (CH/PROG) Page .....	7-6
Local Program (LocalPrg) .....	7-6
Channel.....	7-6
MIDI Bank.....	7-7
MIDI Program (MIDIProg) .....	7-8
Status.....	7-8
Output Assignment (Out) .....	7-8
Destination.....	7-9
MIDI Bank Mode.....	7-9
Entry Program Change (EntryProgChg) .....	7-9
Zone Arpeggiation (ZoneArpeg) .....	7-10
The Key/Velocity (KEY/VEL) Page .....	7-10
Low Key (LoKey), High Key (HiKey).....	7-10
Transpose.....	7-11
Note Map.....	7-11
Velocity Scale (VelScale) .....	7-12
Velocity Offset.....	7-12
Velocity Curve (VelCurve) .....	7-14
Low Velocity (LoVel), HighVelocity (HiVel).....	7-16
The Pan/Volume (PAN/VOL) Page.....	7-17
Entry Volume, Exit Volume .....	7-17

Entry Pan, Exit Pan .....	7-17
The BEND Page .....	7-18
Bend Range (Semitones) and Bend Range (Cents).....	7-18
Aux Bend 1Up and Aux Bend 1 Down .....	7-19
Aux Bend 2 Range.....	7-19
Controllers.....	7-19
Continuous Controllers.....	7-20
Switch Controllers.....	7-21
The MIDI Control Source List .....	7-21
Continuous Controller Parameters.....	7-23
Destination (Dest) .....	7-24
Scale.....	7-24
Offset (Add) .....	7-24
Curve (Curv).....	7-24
Entry (Ent) and Exit Values .....	7-25
The SLIDER and SLID/2 Pages .....	7-25
The Continuous Control Pedal (CPEDAL) Page .....	7-27
The RIBBON Page .....	7-27
The WHEEL Page .....	7-29
The Pressure (PRESS) Page .....	7-29
Switch Controller Parameters.....	7-30
Switch Type (SwType) .....	7-30
Destination.....	7-30
On Value .....	7-31
Off Value.....	7-31
Entry (Ent) and Exit States.....	7-31
The Footswitch (FOOTSW) Page .....	7-31
The SWITCH Page .....	7-32
The KDFX and FXMOD Pages .....	7-33
The COMMON Page .....	7-33
Song.....	7-33
Song Control .....	7-34
Sync .....	7-34
Mutes .....	7-34
The Arpeggiator (ARPEG) Page.....	7-35
Active.....	7-36
Sync .....	7-36
Low Key (LoKey) and High Key (HiKey) .....	7-36
Latch.....	7-37
Order.....	7-38
Beats .....	7-38
Tempo .....	7-38
Note Shift.....	7-38
Shift Limit.....	7-38
Limit Option .....	7-39
Glissando.....	7-40
Velocity .....	7-40
Duration .....	7-40
Real-time Control of Arpeggiator Parameters .....	7-41
The Ribbon Configuration (RIBCFG) Page .....	7-42
Large Ribbon Configuration.....	7-42
Position Mode (PosMode) .....	7-42

Spring.....	7-43
Center.....	7-43
The Utility Soft Buttons .....	7-43
Name.....	7-43
Save .....	7-43
Delete .....	7-43
Dump .....	7-44
New Zone (NewZn).....	7-44
Duplicate Zone (DupZn).....	7-44
Import Zone (ImpZn) .....	7-44
Delete Zone (DelZn) .....	7-44
Copy and Paste.....	7-44
Clear .....	7-45
Set Range (SetRng).....	7-45
Editing Hints and Suggestions.....	7-45
Do These Parameters Always Mean Something? .....	7-45
Do I Need All Those Pedals? .....	7-46
Multiple Controllers .....	7-46
Editing Programs .....	7-46

## **Chapter 8 Quick Access Mode and the Quick Access Editor**

## **Chapter 9 Basic Effects Mode**

Introduction .....	9-1
Terminology .....	9-2
Studio .....	9-2
FXBus (also called Insert FXBus) .....	9-3
Aux FX Bus (Aux Bus).....	9-3
FX Preset.....	9-3
Algorithm .....	9-3
Size/PAUs .....	9-3
Parameters.....	9-3
FXMods.....	9-4
Bus Overrides (or BusMods) .....	9-4
FXCtrl.....	9-4
Studio Editor.....	9-4
MAIN Page.....	9-5
Soft Buttons in Effects Mode .....	9-6
Effects Bus Editor .....	9-7
Effects Send Page.....	9-8
The CTRL Page .....	9-9
Effects Channel (FX Chan).....	9-10
Studio .....	9-11
Bypass and Mute pages.....	9-12
Chaining Effects.....	9-15
The Structure of Kurzweil Digital Effects (KDFX) .....	9-16
The Studio .....	9-16
Normal Studio Structure .....	9-19
Software Organization.....	9-19
Controlling Effects.....	9-19
Controlling Effects with a Sequencer .....	9-20

Controlling Effects with SysEx messages .....	9-20
A Tour of Effects Mode .....	9-21
Load the Tutorial files .....	9-21
A Simple Studio .....	9-21
A More Complex Studio .....	9-26
Chaining Effects .....	9-30
A Complex Studio with Real-Time Control .....	9-33
Building Your Own .....	9-39

## Chapter 10 MIDI Mode

The TRANSMIT Page .....	10-1
Control Setup .....	10-2
Channel .....	10-2
Transpose .....	10-2
Control .....	10-2
Velocity Map—Transmit (VelocMap) .....	10-3
Pressure Map—Transmit (PressMap) .....	10-3
Program Change (PChng) .....	10-3
Buttons (Bttns) .....	10-4
Change Setups (ChgSetups) .....	10-4
The RECEIVE Page .....	10-5
Basic Channel .....	10-5
MIDI Receive Mode (MIDI Mode) .....	10-5
All Notes Off .....	10-6
Program Change Type (ProgChgType) .....	10-6
Velocity Map—Receive .....	10-6
Pressure Map—Receive .....	10-6
System Exclusive ID (SysExID) .....	10-6
SCSI ID .....	10-6
Bend Smooth .....	10-7
Local Keyboard Channel (LocalKbdCh) .....	10-7
Bank Select .....	10-8
Power Mode .....	10-9
The Channels Page .....	10-9
Enable .....	10-10
Program .....	10-10
Pan .....	10-10
Volume .....	10-10
Output Pair (OutPair) .....	10-10
Output Gain (OutGain) .....	10-10
Program Lock, Pan Lock, Volume Lock .....	10-11
Program Change Formats .....	10-11
Extended and Kurzweil Program Change Types .....	10-11
0-127 Program Change Type .....	10-13
Quick Access Banks—Extended (QA Ext) .....	10-13
Quick Access Banks—Kurzweil (QA Kurz) .....	10-15
QA 0-127 .....	10-15
QA Formats and MIDI Transmission .....	10-16
The Soft Buttons in MIDI Mode .....	10-17
Program Change (PrgChg) .....	10-17
Reset Channels (RsetCh) .....	10-18

Panic.....	10-18
------------	-------

## Chapter 11 Master Mode

The Master Mode Page .....	11-1
Tune.....	11-1
Transpose.....	11-2
KB3 Channel .....	11-2
Velocity and Pressure Touch (VelTouch and PressTouch) .....	11-2
Intonation .....	11-2
Confirm.....	11-4
Intonation Key (IntonaKey).....	11-4
The MAST2 Page .....	11-4
The Vocoder .....	11-5
Using the Vocoder.....	11-6
View Mode .....	11-10
Digital Output Format.....	11-11
Digital Output Length.....	11-11
The Soft Buttons in Master Mode .....	11-11
Object .....	11-11
Delete .....	11-11
Util.....	11-12
Sample .....	11-13
GM.....	11-13
Reset .....	11-14
Guitar/ Wind Controller Mode.....	11-14
Object Utilities .....	11-15
Move .....	11-16
Copy .....	11-17
Name.....	11-18
Delete .....	11-18
Dump .....	11-19
Using the Object Utilities from the Editor .....	11-20

## Chapter 12 Song Mode and the Song Editor

Getting Started with the Sequencer .....	12-1
What is a Sequencer? .....	12-1
A Word about the Local Keyboard Channel .....	12-1
Tutorial: Recording a song.....	12-2
Tutorial: Arrangements .....	12-12
Creating an Arrangement .....	12-12
Recording Additional Tracks.....	12-15
Maximizing Track Use.....	12-15
Triggering Arrangement Steps From the Keyboard.....	12-16
RAM Tracks.....	12-17
Two Important Concepts .....	12-17
Creating RAM Tracks .....	12-18
Using Song Mode .....	12-19
Selecting a Song for Playback.....	12-19
Effect Selection During Recording and Playback.....	12-20
Memory Limits .....	12-23
Loading MIDI Files From Disk .....	12-23

Recording Multi-timbral Sequences via MIDI .....	12-23
Song Mode: The MAIN Page.....	12-24
Current Song (CurSong) .....	12-25
Tempo .....	12-25
Recording Track (RecTrk).....	12-26
Program .....	12-26
Channel (Chan) .....	12-27
Setup .....	12-27
Volume (Vol) .....	12-28
Pan.....	12-28
Mode .....	12-28
Location (Locat).....	12-28
Mode Indicators (+ and x): .....	12-28
Activity Indicators (  ).....	12-29
Track Status Indicators .....	12-29
Track Channels .....	12-29
Soft Buttons on the MAIN Page.....	12-29
Save this song? Dialog.....	12-31
Save New song? Dialog.....	12-31
Erase Track Dialog .....	12-32
Song Mode: The MISC Page .....	12-32
RecMode.....	12-33
PlayMode .....	12-34
KeyWait .....	12-35
Locate.....	12-35
AutoIn.....	12-35
AutoOut.....	12-35
Input Quantize (Quant).....	12-35
Grid .....	12-36
Swing .....	12-36
Sync .....	12-36
Clock .....	12-37
Tempo .....	12-37
CountOff.....	12-37
Click .....	12-37
ClickCh .....	12-37
ClickPrg .....	12-37
ClickKey .....	12-37
ClickVel.....	12-38
Soft Buttons on the MISC Page .....	12-38
Song Mode: The MIX Page .....	12-39
Pan Position .....	12-39
Volume Level .....	12-39
Track Status Indicators .....	12-40
Quick Mute Track Bank Indicator.....	12-40
Soft Buttons on the MIX Page.....	12-40
The Song Editor .....	12-41
Soloing the Current Track .....	12-41
Song Editor: The COMMON Page .....	12-41
Tempo .....	12-42
TimeSig .....	12-42
EffectChan .....	12-42

ChainTo.....	12-42
GM.....	12-43
Parameters Used with the Arrangement Feature.....	12-43
Soft Buttons on the COMMON Page.....	12-45
Song Editor: The EVENT Page.....	12-45
Location.....	12-46
Bar, Beat, and Tick.....	12-46
Event Type and Value.....	12-46
Soft Buttons on the EVENT Page.....	12-47
Song Editor: The TRACK Page.....	12-48
Common Parameters for Edit Song: Track Functions.....	12-49
Region / Criteria Box Parameters.....	12-50
Soft Buttons on the TRACK Page.....	12-51
Song Editor: Track Functions.....	12-52
Erase.....	12-52
Copy.....	12-52
Bounce.....	12-53
Insert.....	12-54
Delete.....	12-54
Quantize.....	12-55
Reference Quantize.....	12-56
Shift.....	12-57
Transpose.....	12-57
Change.....	12-57
Thin.....	12-59
Remap.....	12-59
Grab.....	12-60
Song Editor: The STEP Page.....	12-61
Recording With the Step Editor.....	12-61
Soft Buttons on the STEP Page.....	12-63
Song Editor: The ARRANGE Page.....	12-64
Step.....	12-66
Song.....	12-66
Mutes.....	12-66
Xpose.....	12-66
Times.....	12-66
Mode.....	12-67
Triggering Steps from a Key.....	12-67
Soft Buttons on the ARRANGE Page.....	12-68

## Chapter 13 Disk Mode

Disk Mode Page.....	13-2
Using SmartMedia Cards.....	13-2
Connecting a SCSI Device.....	13-3
SCSI Termination.....	13-3
Using your K2661 in a SCSI System.....	13-4
Directories.....	13-4
Path.....	13-4
Startup.....	13-5
Library.....	13-5
Verify.....	13-5



Disk Drive Information .....	13-5
Macro On Indicator.....	13-6
Disk Mode Soft Buttons .....	13-6
File List Dialog.....	13-7
The File Index .....	13-8
Soft Buttons in the File List Dialog .....	13-9
Total.....	13-10
Quick Scrolling to Subdirectories .....	13-11
Creating Directories .....	13-11
Creating a Directory From the Disk Mode Page .....	13-12
Creating a Directory in the Save Dialog .....	13-12
The Directory Selection Dialog.....	13-13
Disk Mode Functions.....	13-14
Loading Files.....	13-14
Loading Individual Objects .....	13-14
Shortcuts when Loading Objects .....	13-17
Loading Dependent Objects .....	13-18
Auditioning Samples from a Disk File.....	13-18
Load Function Dialog .....	13-19
Bank Status Indicator.....	13-19
Loading Methods .....	13-19
Selecting Multiple Files to Load.....	13-22
Aborting a Multiple File Load.....	13-22
More Load Function Features .....	13-23
Saving Files .....	13-24
Soft Buttons in the Save Selection Dialog.....	13-25
Saving Individual Objects.....	13-25
Shortcuts when Saving Objects.....	13-25
More Features of the Save Dialog.....	13-26
Auditioning Objects in RAM.....	13-27
Saving Dependent Objects.....	13-28
The Name Table.....	13-29
Working with Relink-by-Name.....	13-30
Not Loading the Name Table .....	13-34
Relink-by-Name Processing Time .....	13-34
Storing Objects in the Memory Banks.....	13-34
The Multiple Object Selector Page.....	13-35
Using the Multiple Object Selector: An Overview .....	13-35
Operating Modes: The Select Parameter .....	13-35
Multiple Object Selector Soft Buttons .....	13-37
Entering Selection Criteria in the Multiple Object Selector .....	13-39
Macros.....	13-41
The Macro Page.....	13-41
Macro Modes .....	13-42
The Macro Table .....	13-43
How to Make a Macro File.....	13-44
Macro Entries.....	13-48
Using the Bank and Mode Fields.....	13-49
Viewing the Object List for a Macro Entry .....	13-50
Unspecified Disk ID.....	13-50
The Library Disk .....	13-51
Loading Selected Entries from a Macro File .....	13-52

Editing Macros .....	13-53
Macro Insert .....	13-55
Executing the RAM Macro Table .....	13-56
Saving and Loading a Macro Table in a .K26 file .....	13-57
Aborting a Macro Load .....	13-57
Remote Macro Load .....	13-58
Disk Utilities .....	13-59
Find Files .....	13-60
List .....	13-62
Free .....	13-63
Moving Files Between Directories .....	13-63
Renaming Files .....	13-65
Deleting Files and Directories .....	13-66
Backup and Copy Functions .....	13-66
File Backup .....	13-66
File Copy .....	13-68
Creating a Startup File .....	13-70
Deleting Banks in a Startup File .....	13-70
MS-DOS File System Compatibility .....	13-71
Filename Compatibility .....	13-71
Importing and Exporting Data Using Standard File Formats .....	13-72
AIFF and AIFF-C Files .....	13-73
WAVE Files .....	13-73
Standard MIDI Files (MIDI Type 0 and Type 1 Files) .....	13-73
Importing Samples from Other Manufacturers .....	13-74
Akai .....	13-74
Roland .....	13-75
EPS .....	13-75
Disk Partitioning .....	13-76
Doing a Hard Format .....	13-77
The Partitioning Procedure .....	13-77
Partitioning With a Computer .....	13-78
Partitioning Large Disks .....	13-79
Working With Partitions .....	13-79
Managing a SCSI Chain that Includes a Computer .....	13-80
Configuring Device Drivers (Windows 95 and 98 Only) .....	13-80
If a Partition is Inaccessible .....	13-82
Other Macintosh Issues .....	13-82
Partition Support in Disk Mode .....	13-83
Basic Operations .....	13-83
Navigation .....	13-83
Backups and Copies .....	13-84

## Chapter 14 Sampling and Sample Editing

Setting Up For Sampling .....	14-1
Cables and Input Jacks .....	14-1
Entering The Sampler .....	14-1
From Program, Setup, Master, or Quick Access Mode .....	14-2
From the Keymap Editor .....	14-2
Sampling Analog Signals .....	14-2
Input .....	14-3

Recording Samples .....	14-5
Sampling the K2661's Output.....	14-8
Sampling Digital Signals .....	14-8
Editing Samples .....	14-10
The Function Soft Buttons in the Sample Editor .....	14-10
The Page Buttons.....	14-13
The Miscellaneous (MISC) Page .....	14-13
The TRIM Page .....	14-16
The LOOP Page .....	14-17
The DSP Page (RAM samples only) .....	14-19
The Soft Buttons on the DSP Page .....	14-20
DSP Functions.....	14-21
Crossfade and Volume Adjust Curves .....	14-34
The Keymap Editor.....	14-34
The Soft Buttons in the Keymap Editor .....	14-36
Special Double Button Presses in the Keymap Editor .....	14-37
Keymap Editor Parameters .....	14-37
Building a Keymap .....	14-39
Using the Analog Inputs to Trigger Samples .....	14-40
Live Mode.....	14-42
Creating a Live Mode Program.....	14-42
Live Mode Programs .....	14-43
Some Ideas for Using Live Mode.....	14-43

## Chapter 15 Advanced KDFX

Personalities .....	15-1
Navigating KDFX.....	15-1
FXBus .....	15-2
Parameters.....	15-3
The Compare and FX Bypass Buttons.....	15-5
Exploring the Studio Parameters .....	15-6
Input Section .....	15-6
The Arrow Meter .....	15-7
EQ.....	15-7
FXBus Sends.....	15-7
Effects Buses.....	15-9
The Arrow Meter .....	15-9
FX Preset .....	15-9
Bus Outputs .....	15-9
The Aux Bus.....	15-10
About FX Presets .....	15-11
Bus Overrides (Bus Mods) .....	15-12
Allocation .....	15-15
Metering .....	15-17
Output Section .....	15-18
Separate Analog and Digital Outputs.....	15-19
Analog Mix Output.....	15-19
Digital Output .....	15-19
Saving Studios and Other objects .....	15-20
Saving .....	15-20
Disk Functions .....	15-20

Real-time Control of KDFX.....	15-20
Linking to Programs and Setups .....	15-21
Setting Up FXMods.....	15-22
Importing Studios From Another Program or Setup.....	15-27
Using KDFX Live From the K2661 Keyboard .....	15-27
Using KDFX With a Sequencer .....	15-29
Preventing Glitches When Changing Studios .....	15-30
Tempo-based Control of KDFX.....	15-32
KDFX in Master Mode.....	15-34
KDFX in Auto Mode .....	15-34
Using the Algorithms.....	15-35
Conventional Studio Structure.....	15-35
General Parameters.....	15-35
Reverbs .....	15-36
Delays .....	15-37
Complex Echo.....	15-38
Equalizers (EQ).....	15-38
Compressors, Expanders, and Gates.....	15-39
Chorus.....	15-41
Flanger .....	15-42
LaserVerb.....	15-42
Filters .....	15-43
Distortion.....	15-44
Rotating Speakers .....	15-45
Tremolo and AutoPan.....	15-46
Pitch Shifter (Pitcher).....	15-47
Ring Modulation .....	15-47
SRS (Sound Retrieval System).....	15-48
Stereo Simulation .....	15-48
Stereo Analyze .....	15-49
FXMod Diagnostic .....	15-49
Special Topics.....	15-50
Song Mode .....	15-50
KB3 Effects .....	15-51

## **Chapter 16 DSP Functions**

Introduction to Algorithm Programming.....	16-1
Additional Parameters .....	16-3
The DSP Functions .....	16-6
Filters .....	16-6
Filter Terminology.....	16-7
How to Read the Graphs.....	16-7
Equalization (EQ).....	16-25
Pitch / Amplitude / Panner.....	16-34
Mixers .....	16-37
Waveforms .....	16-38
Added Waveforms .....	16-41
Nonlinear Functions .....	16-42
Waveforms Combined with Nonlinear Functions .....	16-52
Mixers with Nonlinear Inputs.....	16-54
Hard Sync Functions .....	16-55

## Chapter 17 FUNS

The Mechanics of Control Sources.....	17-1
Unipolar and Bipolar Control Sources.....	17-1
Programming the FUNs .....	17-2
The FUN Equations.....	17-3
Basic FUN Equations .....	17-4
Warp FUN Equations .....	17-11
Boolean FUN Equations.....	17-14
Sawtooth LFO FUN Equations.....	17-14
Random / Chaotic LFO FUNs .....	17-15
Diode Equation FUNs .....	17-16
The Order of Evaluation for FUNs .....	17-18

## Chapter 18 Other Editors

The Intonation Table Editor .....	18-1
The Velocity Map Editor.....	18-3
Using the Velocity Map Editor .....	18-3
The Pressure Map Editor.....	18-6

## Chapter 19 Audio Outputs

Audio Configurations.....	19-1
Audio Routing: Programs to KDFX.....	19-2
Audio Routing: KDFX to Audio Outputs.....	19-2
Using the Digital Outputs.....	19-3

## Chapter 20 Programming Examples

Trumpet with Delayed Vibrato and Velocity-triggered Falls.....	20-1
Lowpass Filter, Envelopes .....	20-4
Sample and Hold Using a FUN .....	20-6
SHAPER and PANNER.....	20-7
Building a Drum Program; Using the Keymap Editor .....	20-9
Editing a Setup for KB3 Control .....	20-11
Using the KB3 Setup From the SmartMedia Card .....	20-12

## Appendix A K2661 Boot Block

Starting the Boot Block .....	A-1
Updating K2661 Software .....	A-2
Running Diagnostic Tests.....	A-3
Resetting the K2661.....	A-3

## Index



# Chapter 1

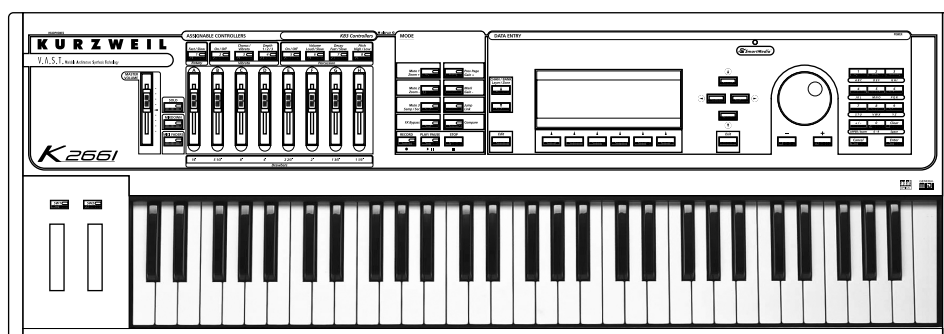
## Introduction

### Welcome!

Congratulations, and thanks for purchasing a Kurzweil/Young Chang K2661 instrument.

The K2661 is packed with great acoustic, electric, and synth sounds—combined with some of the most advanced synthesis features available, which you can use to create almost any sound imaginable. The K2661 incorporates most of the features of the K2600, and provides several new features as well.

This manual and the accompanying *Musician's Reference* will get you started with your new instrument. You'll definitely want to refer to the manuals as you become an advanced user, too.



### Overview of the K2661

The K2661 has been designed to be a versatile instrument both for performance, and for multi-timbral sequencing and recording. Its Variable Architecture Synthesis Technology lets you build sounds from realistic instrumental samples and sampled synth waveforms—then modify the nature of those sounds through a wide variety of digital signal-processing (DSP) functions. The K2661 also generates its own synth waveforms, which can be combined with the samples or used on their own. Onboard sound ROM includes the Orchestral and Contemporary sound blocks, as well as a fine set of General MIDI (GM) sounds. You can add one or both of the two available option ROMs (Stereo Dynamic Piano and Vintage Electric Pianos), and you can load samples from disk into sample RAM.

Before we get into explaining Variable Architecture Synthesis, here are a few of the features that by themselves make the K2661 an impressive stage and studio machine. It's fully multi-timbral—different programs can be played on each MIDI channel. It's 48-note polyphonic, for a full sound no matter how many chords you play. There's an on-board digital effects processor, providing up to five simultaneous effects, including real-time effects control, internally or via MIDI.

In addition to the standard stereo audio output pair, there are four balanced analog outputs and eight channel digital output.

For sample memory, your K2661 has one SIMM (single, in-line memory module) installed in a socket that you can reach through the access panel on the bottom of the instrument. If your K2661 does not already have the maximum of 128 megabytes of Sample RAM installed, you can install a larger SIMM using the instructions in the *Musician's Reference*. The most important thing for you to know about Sample RAM is that it is not battery-backed; RAM samples are permanently erased from memory when you power down.

For offline storage, there's a SmartMedia slot for 3.3v SmartMedia cards, and also a **SCSI** port for connecting an external hard disk or CD-ROM drive. You'll find all this storage potential extremely useful for saving and loading samples, which can also be transferred to and from the K2661 using the standard MIDI sample transfer format, or the faster, parallel SMDI sample transfer format (SCSI Musical Data Interchange). See the *Musician's Reference* for information about MIDI and SMDI sample transfers.

The K2661's battery-backed program RAM can store hundreds of your own programs, or thousands of notes recorded in the sequencer. This sequencer (Song mode) lets you play back MIDI type 0 or 1 sequences, record and play back your own songs, and record multi-timbral sequences received via MIDI.

The Local Keyboard Channel feature enables you to use the K2661's multi-zone setups even if you're using an external MIDI controller that can transmit on only one channel. The K2661 will also rechannelize incoming MIDI information and send it to its MIDI Out port, enabling you to control additional synths on eight different channels.

An optional sampling feature is available, allowing you to make your own mono or stereo samples using analog or digital inputs. With the sampling option, you can also use Live mode, which enables you to take an input signal and route it through the K2661's VAST algorithms—so you can apply Kurzweil DSP and effects to any sound.

There's also digital input/output (I/O) in ADAT format, which provides eight channels for digital audio input and output, as well as a stereo digital output in AES/EBU or S/PDIF formats.

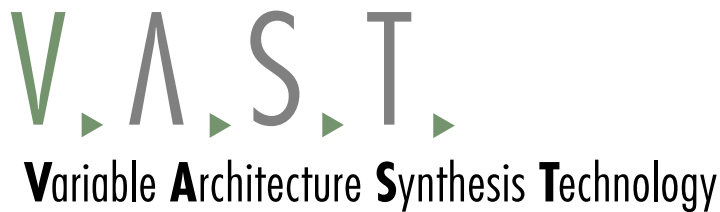
And, of course, there's the incomparable Kurzweil sound. The K2661 comes to you with hundreds of programs (called patches, presets, voices, etc. on other synths). There are also about 200 multi-zone performance setups. Many of these setups use note triggers to play factory-recorded songs that provide grooves and arpeggiation that make great templates for performance or recording.



## VAST Synthesis

*Variable Architecture Synthesis Technology* gives the K2661 its unprecedented flexibility. While many other synthesizers offer a fixed set of **DSP** tools (typically filtering, pitch, and amplitude modulation) the K2661's Variable Architecture lets you arrange a combination of any five DSP functions from a long list of choices. The functions you choose define the type of synthesis you use.

Each layer of every program has its own DSP architecture, which we call an *algorithm*. Within each algorithm, you can select from a variety of DSP functions. Each function can be independently controlled by a variety of sources including LFOs, ASRs, envelopes, a set of unique programmable functions (FUNs), as well as any MIDI control message. The many different DSP functions and the wealth of independent control sources give you an extremely flexible, truly vast collection of tools for sound creation and modification. When you're ready to jump in and start creating and editing programs, turn to Chapter 6.



## KB3 Tone Wheel Emulation

In addition to VAST synthesis, the K2661 offers many oscillator-based programs that give you the classic sound of tone-wheel organs like the Hammond B 3™. KB3 mode, as we call it, is completely independent of VAST, and has its own set of editing procedures. You'll find details in Chapter 6.

## How the K2661 Works

The K2661 integrates three MIDI-driven components: a MIDI controller (the keyboard, or an external MIDI controller), a sound engine, and a global effects processor (KDFX). The sound engine responds to the MIDI events generated by the MIDI controller, and turns them into sounds that are processed within the variable architecture of the algorithms—or by oscillators for KB3 programs. The resulting sound can then be routed through KDFX and to the audio outputs.

## How to Use This Manual

This manual includes the following:

- How to connect and power up your K2661, getting around the front panel, and a brief description of the operating modes
- Basic editing, including the normal operations of each operating mode
- The advanced programming features that make the K2661 so powerful—a number of familiar synthesis tools and quite a few new ones

Even if you're a complete techie, you should read Chapter 3, *User Interface Basics*. Here you'll get a tour of the K2661's front panel and learn how to navigate through the major functions.

Chapter 4, *The Operating Modes* describes the concept and operation of the K2661's eight operating modes, with a brief description of each. Chapters 6 through 13 describe each mode in detail, including the editors contained within them. Chapters 14 through 17 discuss the advanced editing features. Chapter 19 shows you how to use the multiple audio outputs to suit your needs. Chapter 20 is a programming tutorial, giving you specific examples of many of the K2661's programming functions. Some are basic, some are advanced. By working through the tutorials, you'll become familiar with many synthesis techniques, and you'll get a first-hand look at how to unleash the power of the K2661.

When manual text appears in boldface italic (*like this*), you'll find it described in the Glossary in the *Musician's Reference*. Only the first one or two occurrences of these words are highlighted.

You'll also see text in bold face (**like this**). We use bold face to indicate parameter values, K2661 objects (like programs and setups), filenames (like songs and samples), and front-panel buttons (like the **Program** button and the soft buttons).

The best way to read this manual is with your K2661 in front of you. By trying the examples we give to illustrate various functions, you can get a quick understanding of the basics, then move on to the more advanced features.

If you like to figure out your equipment for yourself, and normally use manuals only as reference sources, you'll probably get the most use out of the *Musician's Reference* that accompanies this manual, which contains brief descriptions of the K2661's major operating features, and all sorts of useful lists—*programs*, *keymaps*, *algorithms*, effects, *control sources*, and more. A glossary and complete specifications for the K2661 are also included in the *Musician's Reference*.

## Do I Have Everything?

Your K2661 shipping carton should include the following in addition to your instrument:

- Power cable
- Sustain pedal
- *Getting Started* manual
- SmartMedia card
- Compact disc with documentation and accessory files
- Warranty card

If you don't have all of these components, please call your Kurzweil/Young Chang dealer.

## The Accessory Files

The accessory files included with your K2661 (on SmartMedia and Compact Disc) include the following. Additional files may be included; see the appropriate README files for these.

### K2661 Base ROM Objects

These provide a backup of the ROM Objects installed in your K2661 at the factory.

You will not normally need to use these, however you should keep them in a safe place in case you need to reinstall this version of the operating system. For example, if you have installed a newer version of the objects (we make new versions available for download from our web site) and you encounter any problems, you might need to go back and install the version on SmartMedia and Compact Disc.

OBJKB contains all of the objects in a stock K2661 without sampling. OBJKL contains the Live Mode objects, which are included in a unit with the sampling option installed.

When you install a new version of the operating system, you'll need to load all the object files for the options you have. This includes the object files for the SD Piano and Vintage Electric Piano ROM blocks, if you have them. Load the system files first, then all of the object files, then exit the Boot Loader by pressing the **Run System** soft button. See Appendix A if you need help with operating the Boot Loader.

### K2661 Demos

This includes several directories of demo song files, which show off some of the capabilities of your K2661. In addition, **KDFXTUTR.K26** contains objects used in the tour of Effects mode that starts on page 9-22.

Most of the song demos can simply be loaded into the K2661 and played from Song mode. However, two demo files require that your K2661 have the sampling option installed. The demo files **LMFBDMO1.K26**, and **LMFBDMO2.K26** demonstrate internal feedback loops using a chain of VAST->KDFX->sampler->VAST->KDFX—known as Live mode. Because a Master table is included, your SampleMode-page settings will be set automatically when you load the demo file.



**Caution:** Programs that cause internal feedback can get extremely loud, so it's important to be careful after loading these objects. Live-mode programs are inherently unstable when you're in Live mode and the sampling source (the Src parameter in the Samplemode page) is set to **Int**. These programs are capable of producing tremendously loud feedback loops with a relatively small amount of gain. In modes where the provided studio is not the active studio, the compressor that keeps these feedback chains under control isn't operating, and the loop can grow out of control very quickly. If you intend only to listen to the demo, and not to examine the programming, we recommend that you delete the song file (and its dependent objects) from your K2661 when you are done listening. We also recommend that you go to the SampleMode page and set the Src parameter to **Ext** or the Mode parameter to something other than **LiveIn**, to help to avoid any unintended feedback. If you want to look at the program settings, lower the volume on your instrument, and read the more detailed description of Live mode beginning on page 14-41.

### K2661 Farm & Extras

This includes over 1000 extra programs and effects, all cultivated from the "farm" where we grow our programs and effects.

The **K2500FRM** directory contains over 1000 extra programs, organized in files by type of sound. This set of programs was originally developed for the K2500, hence the name.

The **KDFXFARM** directory contains two subdirectories. **LIVEMODE** contains a set of files containing extra programs using Live mode (which requires having the sample option in your K2500). **MOREKDFX** contains some FX Presets and a number of Programs that use KDFX, as well as a couple of blank studios.

**PIANOFRM** contains extra programs that specifically use the 4-megabyte stereo piano.

**VOCODER** contains programs and setups that are designed to be used with the Vocoder feature. You can read about the Vocoder in detail on page 11-4.

The **MOREPRGS** directory contains several files of extra programs.

## K2600 Compatibility

This file includes all the ROM objects in the original K2600. You would use this file if you have songs created for a K2600 with v1 objects. See *Loading Older Setup Versions* on page 7-2 for information about using older objects with the K2661.

## K2500 Compatibility

This contains a file of all the ROM objects in the K2500. You would use this file if you have songs created originally for a K2500. See *Loading Older Setup Versions* on page 7-2 for information about using older objects with the K2661.

## K2000 Compatibility

This file includes all the ROM objects in the K2000. You would use this file if you have songs created originally for a K2000. Again, see *Loading Older Setup Versions* on page 7-2.

# Chapter 2

## Startup

If hooking up new gear is familiar to you, and you just want to get going, here's a quick description of all the basic things you need to cover to get started with your K2661. If you need more information, thorough descriptions of each step follow. In either case, check out *Playing the Presets* on page 2-4.

### Quick Startup Checklist

1. Set the keyboard on a hard, flat, level surface. Make sure to leave plenty of room for ventilation.
2. Four adhesive-backed rubber feet are provided with your K2661. If you want to attach them to the bottom of the K2661, carefully turn the keyboard over, remove the paper backing from the rubber feet and attach them now. There are four pairs of holes on the bottom of the keyboard – a pair in each corner – showing the best places to attach the rubber feet.
3. Connect the power cable.
4. Make sure your sound system is at a safe volume level.
5. Plug in a pair of stereo headphones or run standard (1/4-inch) audio cables from your amplifier or mixer to the MIX audio outputs on the K2661. (Use the MIX L out for mono.)

### Start Jamming!

1. Power up your K2661 and check out some of the programs and setups. The K2661 starts up in either Program mode or Setup mode—whichever of the two modes it was in when it was last shut down (or given a soft reset). Press the button labeled **Program** or **Setup** to switch modes.
2. If you hear distortion, reduce the gain on your mixing board, or use the pad if it has one.
3. Scroll through the program list with the Alpha Wheel.
4. Press the **Quick Access** mode button and use the alphanumeric buttons to select from programmable banks of ten programs or setups.
5. If you don't hear anything, review these steps, or check the Troubleshooting section in the *Musician's Reference*.

## Startup—the Details

This section walks you through the hookup of your K2661. We'll take a look at the rear panel, then describe the power, audio, and other cable connections.

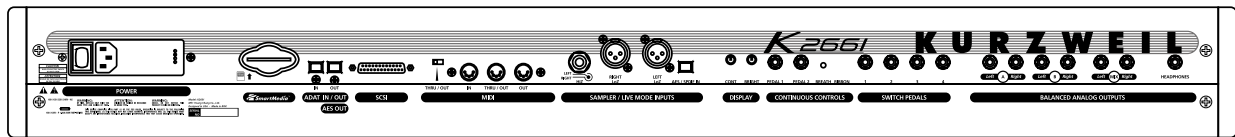
### Before You Start...

Don't connect anything until you make sure your K2661 is properly and safely situated. Also, if your K2661 has been out in the cold, give it time to warm up to room temperature before starting it, since condensation may have formed inside the K2661. *Never* block the ventilation openings on the bottom or rear panels; doing so can cause overheating and serious damage. To provide adequate ventilation, the rear panel should be at least four inches from any vertical surface.

### Connecting the Power Cable (Line Cord)

The K2661 runs on AC power: 100, 120, 230, or 240 volts at 50–60 Hz. Your dealer will set the voltage switch to match the voltage in your area. The voltage level is set with a selector on the rear panel of the K2661. Unless you are sure it needs to be changed, you shouldn't adjust this.

When you've connected the cable at the K2661 end (as you face the back of the K2661, the power connection is at the left), plug it into a grounded outlet. If your power source does not have the standard three-hole outlet, you should take the time to install a proper grounding system. This will reduce the risk of a shock.



### Connecting the Audio Cables

After you've turned down the level on your sound system, connect the K2661's analog audio outputs to your sound system using a pair of stereo or mono audio cables. Mono cables will always work, but if you're going into balanced inputs, use stereo cables for a better signal-to-noise ratio. The K2661's analog outputs are balanced, and generate a "hotter" signal than previous Kurzweil instruments.

You'll find six 1/4-inch jacks near the top of the rear panel. For now, connect one end of each audio cable to your mixing board or PA system inputs, and connect the other end to the jacks marked MIX L and R on the rear panel of the K2661. If you have only one input available, use the K2661's MIX L output to get the full signal in mono. You'll find more about audio configurations—including digital input and output—in Chapter 19.

### Connecting MIDI

The simplest MIDI configuration uses a single MIDI cable: either from the MIDI Out port of your K2661 to the MIDI In port of another instrument, or from the MIDI Out port of your MIDI controller to the MIDI In port of the K2661. There are all sorts of possible configurations, including additional synths, personal computers, MIDI effects processors, and MIDI patch bays. Depending on your system, you may want to use the K2661's MIDI Thru port to pass MIDI information from a MIDI controller to the K2661 and on to the next device in your system. You can also connect MIDI devices to the K2661's MIDI Out port, which can send channelized MIDI information from the keyboard or through the K2661 from your MIDI controller. See the discussion of the Local Keyboard Channel parameter on page 10-7.

## Connecting SCSI

You may not have a hard disk or other SCSI device to connect to your K2661 right away, but if you do, you can connect it to the SCSI port. Please read the following information carefully; it's very important. Also, there's a collection of SCSI tips in Chapter 6 of the *Musician's Reference*.



**Note:** SCSI on the K2661 is always terminated. Turn to Chapter 13 and read the section called SCSI Termination if you require more information. You can lose data if your system isn't terminated properly.

## Switching On the Power

The power switch is on the rear panel. If you're facing the keyboard, it's just to the right of the power cable connection.

When you power up, the display briefly show some startup information. The Program mode display then appears. It looks like the diagram below (the programs shown in the diagram don't necessarily exist).

```

ProgramMode  XPose:M51  <>Channel:1
998 James Jams
999 Default Program
1 Righteous Piano
2 Mondo Bass
3 Killer Drums
4 Weeping Guitar
Octav- Octav+ Panic Sample Chan- Chant

```

The first time you power up (or after a reset), your instrument will be set to operate on MIDI Channel 1 (as shown in the top line of the diagram). After that, it will power up on whatever channel you were on when you powered down. If you'd rather have the K2661 reset certain parameters to default values when you power up, you can do that, too. See *Power Mode* on page 10-9.

Set the volume at a comfortable level. You'll get the best signal-to-noise ratio if you keep the K2661 at full volume, and adjust the level from your mixing board. You may also want to adjust the display contrast and brightness. There are two small knobs on the rear panel of the K2661 for this purpose.

## SmartMedia

You can use 3.3v SmartMedia cards (4 megabyte and larger) for backing up, archiving, and sharing your work. The SmartMedia card slot is on the back panel of the K2661, but it is easily accessible from the front of the instrument. The gold contacts on the card must be facing up when you insert it; the K2661 can't read a card when it is inserted upside down.



**Caution:** Do not remove a SmartMedia card while the blue LED is lit. Removing a card while the blue LED is lit can cause data corruption.

## Playing the Presets

There are three things you'll want to check out right away: programs, setups and Quick Access banks. In performance situations, you'll be selecting your sounds using one of these three methods. There's a mode (and a mode button) corresponding to each method.

### Getting Around

In all three of these modes, the bottom line of the display identifies the function of each of the buttons beneath the display. We call these buttons *soft* buttons, because they do different things depending on what's currently showing in the display.

In Program and Quick Access modes, you can change MIDI channels with the **Chan-** and **Chan+** buttons under the display. In Program mode, you can also change channels using the **Chan/Bank** buttons to the left of the display. (In Quick Access mode, the **Chan/Bank** buttons change Quick Access banks; more about that on page 2-7.) There are two more soft buttons that appear in these modes: Panic and Sample.

The **Panic** button sends an All Notes Off message and an All Controllers Off message—both to the K2661 and over all 16 MIDI channels. You won't need it often, but it's nice to have.

The **Sample** button takes you to Sample mode (the SampleMode page), which, as you've probably guessed, is where you make your own samples—if you have the sampling option in your instrument.

### Programs

The K2661 powers up in Program mode, where you can select and play programs stored in **ROM** or **RAM**. Programs are preset sounds composed of up to 32 *layers* of samples or waveforms. If you've left Program mode, just press the **Program** mode button or **Exit** button to return.

Take a minute to familiarize yourself with the Program-mode display. It gives you some helpful basic information, like the MIDI transposition, what MIDI channel you're on, and which program is currently selected.

There's a box at the left side of the display. The info box, as it's called, displays information about the current program (there's also an info box for Setup mode). The following diagrams shows how the information differs depending on the type of program you've selected.

#### VAST Programs

First is a "normal" VAST program, which is what most of the factory programs are. They have from one to three layers. In this case, there are two. For programs of up to three layers, the info box contains one line per layer, indicating the keymap used in that layer. The line under the keymap name indicates the layer's keyboard range. In this case, both layers extend across the entire keyboard (A 0 to C 8).

```

ProgramMode  XPose:051  <>Channel:1
5 Piano for Layers
6 DrkPno^ArakisPno
7 Honky-Tonk
8 Pno&Syn/HcString
9 ClassicPiano&Vox
10 E Grand Stack
Octav- Octav+ Panic Sample Chan- Chan+
Keymap Info
Grand Piano
Grand Piano

```



VAST programs with more than three layers are known as *drum* programs, as shown in the following diagram. Drum programs can use any sound you like, but the most common use for programs with more than three layers is to create a number of different sounds across the keyboard—which is perfect for combinations of percussion sounds (hence the name). For drum programs, the info box simply indicates the number of layers in the program—in this case, 26.

```

ProgramMode  XPose:051  <>Channel:1
Drum Program  48 OG
26 layers     49 Lowdown Bass
              50 SquashStudio Kit
              51 Retro Skins MW
              52 2 Live Kits 2 MW
              53 Garage Kit II MW
Octav- Octav+ Panic Sample Chan- Chan+

```

### KB3 Programs

The architecture of KB3 programs is different from that of VAST programs. KB3 programs don't have layers; they rely on oscillators that mimic the tone wheels used in many popular organs. Consequently, the info box shows only the waveform used in the program.

```

ProgramMode  XPose:051  <>Channel:1
KB3 Program  751 Prog Rock Organ
Mellow Vox   752 Syn Rock Organ
              753 Dirty Syn B
              754 CleanFullDrawbar
              755 Loungin
              756 MildGrunge
Octav- Octav+ Panic Sample Chan- Chan+

```

Because of their architecture, KB3 programs require different processing within the K2661, and they don't work on "regular" channels. They require a special channel to handle the KB3 program's voices. You can choose any of the 16 MIDI channels to be the KB3 channel, but you can have only one KB3 channel, and KB3 programs play *only* on that channel (VAST programs work just fine on the KB3 channel, by the way).

By default, Channel 1 is designated as the KB3 channel (you can change it in Master mode). If the current channel isn't the same as the KB3 channel, and you select a KB3 program, you won't get any sound, because a non-KB3 channel can't handle a KB3 program.

In the following diagram, Channel 2 is the current channel, but Channel 1 is the KB3 channel. The info box tactfully lets you know what the KB3 channel is, and parentheses appear around the names of all KB3 programs, to further remind you that KB3 programs aren't available on the current channel. To get Program 753 to work in this case, you'd have to change the current channel back to Channel 1, or go to Master mode and change the KB3 channel to Channel 2.

```

ProgramMode  XPose:051  <>Channel:2
KB3 Program   751 (Prog Rock Organ)
Mellow Vox    752 (Syn Rock Organ)
              753 (Dirty Syn B)
KB3Chan is Ch 1 754 (CleanFullDrawba)
              755 (Loun9in)
              756 (Mild Grunge)
Octav- Octav+ Panic Sample Chan- Chan+

```

### Live Mode Programs

Programs 740–749 are designed expressly for Live mode, a feature that enables you to take a signal from the Analog sampling input (this requires the sampling option, of course), and route it through the DSP algorithms. Programs 740–749 will not play unless you're receiving a signal at the Analog sampling input. See page 14-42 for more about Live mode.

### Selecting Programs

When you want to change programs, you have several options.

The easiest method is the Alpha Wheel. Turning it left or right will scroll through the program list. You can also change programs using the cursor buttons (the arrow buttons to the right of the display), or the **Plus/Minus** buttons under the Alpha Wheel. To save time, you can enter the program number directly from the alphanumeric buttonpad to the right of the Alpha Wheel, pressing the **Enter** button to complete the selection. If you make a mistake, press **Clear**, then start over.

The K2661 has various settings for responding to MIDI Program Change commands from external sources. These are explained in Chapter 10, so we won't go into them here. You should be able to change programs by sending Program Change commands from your MIDI controller.

If you don't hear anything, see the troubleshooting section in the *Musician's Reference*. When you're ready to start doing your own programming, check out Chapter 6.

## Setups

Setups are preset combinations of programs. Setups can have up to eight zones, each of which can be assigned to any range of the keyboard (overlapping or split). Each zone can have its own program, MIDI channel, and MIDI control assignments.

Press the **Setup** mode button to the left of the display. Its LED will light, telling you that you're in Setup mode. Notice that the Setup-mode display is similar to the Program-mode display. If the setup has three or fewer zones, the box at the left shows you the programs assigned to each of the setup's three zones, and which MIDI channel is used for each program. If the setup is composed of more than three zones, then the box displays a series of horizontal lines illustrating the approximate key ranges of the zones. See page 7-1 for a more detailed description.

```

SetupMode      XPose:GST
Chan/Program Info
1      9 Cool Traps
2     18 Sly Acoust
3     22*Izit Jimmy
Octav- Octav+ Panic Sample
201*Friday Gig
202*Bop Rock Reggae
1 Jazz Trio
2 All Percussion
3 Heavy Metal
4 To Sequencer

```

Many setups include arpeggiation and note-triggered songs to create some pretty amazing grooves that you can use as is, or as templates for your own material. As you play with these setups, experiment with the sliders and other controllers for a wide range of effects. Some of these grooves keep playing after you've released the keys that got them going. When you want to stop them, select another setup, or press the **Setup** mode button.

## Quick Access

A really convenient way to select programs and setups is to use Quick Access mode, where you select a Quick Access *bank* from a list of factory preset or user-programmed banks. Each bank contains ten memory slots, or entries, where you can store any combination of programs or setups. While you're in Quick Access mode, you can select any program or setup in the bank with buttons 0 through 9.

The K2661 comes with a few Quick Access banks already programmed so you can get an idea of how they work. You'll probably create your own Quick Access banks to help you select programs and setups with a minimum of searching. Press the **Quick Access** mode button to the left of the display. Its LED lights, to tell you you're in Quick Access mode. You'll see a display that looks like this:

```

QuickAccessMode  <>Bank:1 For Show 1
Silk Rhodes     POLY TOUCH      Voice+String
NastyTrombone   SINK MONSTA      Waterflute
Soon            Jazz Trio 2      Fretless Bas
                VELVETEEN
XPose:GST      SINK MONSTA      Chan:1
Octav- Octav+ Panic Sample Chan- Chan+

```

The top line of the display tells you which Quick Access bank is selected. Use the **Chan/Bank** buttons (to the left of the display) to scroll through the banks. The names of each of the ten entries in the bank are listed in the center of the display. Many of their names will be abbreviated. The currently selected entry's full name is shown near the bottom of the display. The amount of transposition is displayed to the left of the entry name. If the current entry is a program, you'll see the current keyboard (MIDI) channel displayed to the right of the entry's name. If it's a setup, you'll see the word Setup.

The entries on the Quick Access page are arranged to correspond to the layout of the numeric buttons on the alphanumeric pad. On the page above, for example, the program **SINK MONSTA** is entry 5, and can be selected by pressing 5 on the alphanumeric pad.

When you're ready to create your own Quick Access banks, turn to Chapter 8 to learn about the Quick Access Editor.

## The Other Modes

There are six other modes, five of which have mode buttons on the front panel (Sample mode doesn't have a front-panel button). See page 3-1 and Chapter 4 for more detailed descriptions of each mode.

<b>Effects mode</b>	Create and modify effects presets, and define how they're applied to the K2661's programs and setups.
<b>MIDI mode</b>	Configure the K2661 for sending and receiving MIDI information.
<b>Master mode</b>	Define performance and control settings.
<b>Song mode</b>	Record and edit sequences (songs); play Type 0 and Type 1 MIDI sequences.
<b>Disk mode</b>	Load and save programs, setups, samples, and other objects.
<b>Sample mode</b>	Edit ROM samples; create and edit RAM (user-defined) samples.

## Software Upgrades

Part of the beauty of the K2661 is the ease with which you can upgrade its operating system and objects (programs, setup, etc.) using the boot loader to install upgrades into flash ROM.

At Kurzweil and Young Chang, we have a long history of support for our instruments; the K250, K1000, K2000, and K2500 have been repeatedly enhanced, and these improvements have always been made available to instrument owners in the form of software upgrades.

Upgrading your K2661's software is simple, painless, and—generally—free! As upgraded software becomes available, you can either get files from your Young Chang dealer or download the new stuff from the Web.

To stay in touch, check out our Web site:

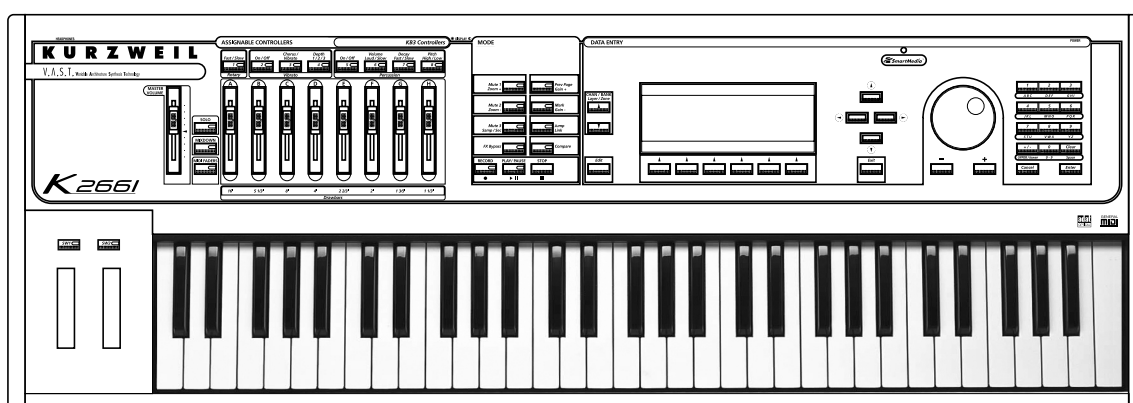
<http://www.kurzweilmusicsystems.com/>

When you've acquired an upgrade, you can install it yourself in a matter of minutes. See Appendix A for details.

# Chapter 3

## User Interface Basics

This chapter will show you how to get around the front panel of your K2661. Your interactions can be divided into three primary operations: mode selection, navigation, and data entry. There is also an assignable control section.



## Mode Selection

The K2661 is always in one of eight primary operating modes. Select a mode by pressing one of the mode buttons — they're to the left of the display. Each mode button has an LED that lights to indicate the current mode. Only one mode can be selected at a time.

<b>Program mode</b>	Select and play programs, and modify them with the <i>Program Editor</i> . Rearrange and modify samples in the Keymap and Sample Editors.
<b>Setup mode</b>	Select and play setups (eight keyboard zones with independent MIDI channel, program and control assignments), and modify them with the Setup Editor.
<b>Quick Access mode</b>	Select from a list of preset banks, each containing a list of ten programs and/or setups that can be viewed in the display for easy selection. Modify the preset banks and create your own with the Quick Access Editor.
<b>Effects mode</b>	Define the behavior of the on-board effects. Modify the preset effects and create your own with the Effects Editor.
<b>MIDI mode</b>	Define how your K2661 sends and receives MIDI information, and configure each channel to receive independent program, volume, and pan messages that override the normal Program-mode settings.
<b>Master mode</b>	Define performance and control characteristics for the entire K2661.

<b>Song mode</b>	Use the K2661's sequencer to record and play back your keyboard performance, play Type 0 and Type 1 MIDI sequences, and record multi-timbral sequences received via MIDI.
<b>Disk mode</b>	Interface with the K2661's SmartMedia drive, or an external SCSI device to load and save programs, setups, samples, and more.

There are two more modes that don't have dedicated buttons on the front panel: Sample mode and Live mode. In Sample mode, you can create and edit samples (if you have the sampling option). To get to Sample mode, press the **Sample** soft button in Program, Setup, Quick Access, or Master modes. See Chapter 14 for details.

Live mode lets you route an input signal through the K2661's DSP algorithms. See page 14-42.

Mode Buttons

The mode buttons are labeled in white. When you press a mode button, its LED lights up to indicate that the mode has been selected. If pressing a mode button does not light its LED, press the **Exit** button one or more times, then try again.

The colored labeling for each mode button indicates special functions that relate to some of the K2661's editors. These functions are described on page 5-8.

Navigation

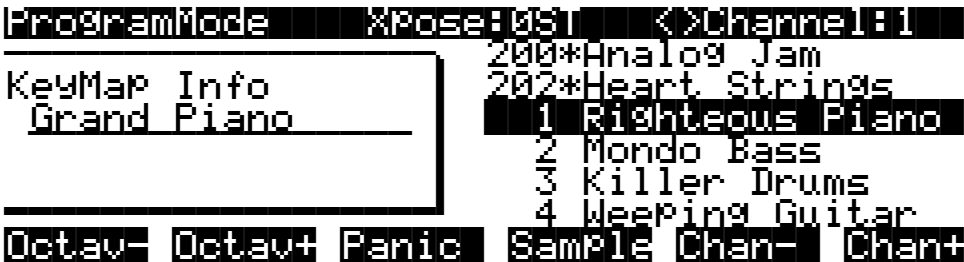
The navigation section of the front panel consists of the display and the buttons surrounding it. These navigation buttons will take you to every one of the K2661's programming parameters.

The Display

Your primary interface with the K2661 is its backlit graphic display. As you press various buttons, this fluorescent display reflects the commands you enter and the editing changes you make. The ample size of the display (240-by-64 *pixels*) enables you to view lots of information at one time.

Pages

Within each mode, the functions and parameters are organized into smaller, related groups that appear together in the display. Each one of these groups of parameters is called a *page*. Each mode has what we call an entry-level page; it's the page that appears when you select that mode with one of the mode buttons. Within each mode and its editor(s), the various pages are selected with the navigation buttons. There are many pages, but there are a few features common to each page. The diagram below shows the entry-level page for Program mode.



## The Top Line

On the top line of most pages, there's a reminder of which mode you're in and which page you're on. Many pages display additional information in the top line, as well. The Program-mode page above, for example, shows you the current amount of MIDI transposition and the currently selected MIDI channel. The top line is almost always "reversed"—that is, it has a white background with blue characters.

## The Bottom Line

The bottom line is divided into six (sometimes fewer) sets of reversed characters that serve as labels for the six buttons directly beneath the display. These labels—and the functions of the buttons—change depending on the currently selected page. Consequently the buttons that select these functions are called "soft" buttons.

## The Soft Buttons


The soft buttons are called "soft" because their functions change depending on the currently selected mode. Sometimes they perform specific functions, like changing MIDI channels in Program mode. In the Program Editor and other editors, they're also used to move to different pages of programming parameters. If a soft button's label is in all capital letters (**KEYMAP**, for example), pressing the corresponding soft button takes you to a page of parameters. If the soft button is labeled in lower-case or mixed-case letters (**Save**, for example), the soft button performs some kind of function.

## The Cursor Buttons

To the right of the display are four buttons arranged in a diamond fashion. These are called the cursor buttons. They move the cursor around the currently selected page, in the direction indicated by their labels. The cursor is a highlighted (reversed) rectangle (sometimes it's an underscore). It marks the value of the currently selected parameter.

Programming the K2661 involves selecting various parameters and changing their values. Select parameters by highlighting their values with the cursor. You can change the highlighted value with any of the data entry methods described in the data entry section below.

## The Chan/bank Buttons

To the left of the display are two buttons labeled **Chan/Bank**. Their function is related to the two small arrows——that appear in the top line of the display when there are multiples of the current page—for example, the LAYER pages in the Program Editor. When you see these arrows, you can use the **Chan/Bank** buttons to scroll the values of the parameter that appears to the right of the arrows. In Program mode, for example, they shift through the MIDI channels, showing the program assigned to each channel.

When you're in the Program Editor, the **Chan/Bank** buttons let you view each layer in the program. You can see the corresponding parameters in each layer by scrolling through the layers with these buttons. In the Keymap Editor, **Chan/Bank** scroll through key ranges of the current keymap. In the Setup Editor, the **Chan/Bank** buttons scroll through the zones in the current setup. In Quick Access mode, they scroll through the Quick Access banks, and in Song mode they scroll through recording tracks.

We'll let you know, when applicable, what the **Chan/Bank** buttons do.

## The Edit Button

The **Edit** button activates each of the K2661's editors, and acts as a shortcut to many pages within the Program Editor. Pressing the **Edit** button tells the K2661 that you want to change some aspect of the object marked by the cursor. For example, when a program is selected and you press **Edit**, you enter the Program Editor. If a setup is selected, you enter the Setup Editor.

There are editors accessible from just about every operating mode. To enter an editor, choose one of the modes (mode selection), and press **Edit**. An editing page for that mode will appear. You can then select parameters (navigation) and change their values (data entry). If the value of the selected parameter has its own editing page, pressing the **Edit** button will take you to that page. For example, in the Program Editor, on the PITCH page, you might see **LFO1** assigned as the value for Pitch Control Source 1. If you select this parameter (the cursor will highlight its value—**LFO1** in this case), then press the **Edit** button, you'll jump to the page where you can edit the parameters of LFO1. Naturally, you can find every page in the current editor by using the soft buttons, but often it's easier to use the **Edit** button shortcut.

## The Exit Button

Press **Exit** to leave the current editor. If you've changed the value of any parameter while in that editor, the K2661 will ask you whether you want to save your changes before you can leave the editor. See page 5-3 for information on saving and naming. The **Exit** button also takes you to Program mode if you're on the entry level page of one of the other modes. If at some point you can't seem to get where you want to go, press **Exit** one or more times to return to Program mode, then try again.

## Data Entry

The data entry section of the front panel includes the Alpha wheel, the **Plus/Minus** buttons, and the 14-button alphanumeric pad.

### The Alpha Wheel

The Alpha Wheel is especially useful because it can quickly enter large or small changes in value. If you turn the Alpha Wheel one click to the right, you'll increase the value of the currently selected parameter by one increment. One click to the left decreases the value by one increment. If you turn it rapidly, you'll jump by several increments. You can also use the Alpha Wheel to enter names when you're saving objects.

### The Plus/Minus Buttons

These buttons are located just under the Alpha Wheel. The **Plus** button increases the value of the currently selected parameter by one, and the **Minus** button decreases it by one. These buttons are most useful when you're scrolling through a short list of values, or when you want to be sure you're changing the value by one increment at a time. One press of the **Plus** or **Minus** button corresponds to one click to the right or left with the Alpha Wheel. These buttons will repeat if pressed and held.




Pressing the **Plus** and **Minus** buttons simultaneously will move you through the current list of values in large chunks instead of one by one. Often this is in even increments (10, 100, etc.). Don't confuse these buttons with the +/- button on the alphanumeric pad. This button is used primarily for entering negative numeric values and switching from uppercase to lowercase letters (and vice versa).



## The Alphanumeric Pad

As its name implies, this set of 14 buttons lets you enter numeric values, and to enter names one character at a time. Depending on where you are, the K2661 automatically enters letters or numerals as appropriate (you don't have to select between alphabetic or numeric entry).

When you're entering numeric values, press the corresponding numeric buttons, ignoring decimal places if any (to enter 1.16, for example, press **1**, **1**, **6**, **Enter**). The display will reflect your entries, but the value won't actually change until you press **Enter**. Before pressing **Enter**, you can return to the original value by pressing **Cancel**. Pressing **Clear** is the same as pressing **0** without pressing **Enter**.

When entering names, you can use the **Left/Right** cursor buttons or the  /  soft buttons to move the cursor to the character you want to change. Use the labels under the alphanumeric buttons as a guide to character entry. Press the corresponding button one or more times to insert the desired character above the cursor. The **Cancel** button is equivalent to the  soft button, and **Enter** is the same as **OK**. The **Clear** button replaces the currently selected character with a space. The **+/-** button *toggles* between uppercase and lowercase letters.

There's also a convenient feature called keyboard naming, which lets you use the keyboard to enter characters in names. See page 5-5.

## Double Button Presses

Pressing two or more related buttons simultaneously executes a number of special functions depending on the currently selected mode. Make sure to press them at exactly the same time.

In this mode or editor...	...pressing these buttons simultaneously...	...does this:
Program mode	Octav-, Octav+	Reset MIDI transposition to 0 semitones. Double-press again to go to previous transposition.
	Chan-, Chan+	Set current MIDI channel to 1.
	Plus/Minus	Step to next Program bank (100, 200, etc.)
Master mode	Chan/Bank	Enables Guitar/Wind Controller mode.
Song mode	Left/Right cursor buttons	Toggle between Play and Stop.
	Up/Down cursor buttons	Toggle between Play and Pause.
	Chan/Bank	Select all tracks on any TRACK page in Song Editor.
Disk mode	2 leftmost soft buttons	Issue SCSI Eject command to currently selected SCSI device.
	Chan/Bank	Hard format SCSI device. List selected objects when saving objects.
	Left/Right cursor buttons	Select all items in a list. Move cursor to end of name in naming dialog.
	up/down cursor buttons	Clear all selections in a list. Move cursor to beginning of name in naming dialog.
Program Editor	Chan/Bank	Select Layer 1.

**Table 3-1 Double Button Presses**

In this mode or editor...	...pressing these buttons simultaneously...	...does this:
Keymap Editor	Plus/Minus	With cursor on the Coarse Tune parameter, toggles between default Coarse Tune of sample root and transposition of sample root.
Sample Editor	2 leftmost soft buttons	Toggle between default zoom setting and current zoom setting.
	Plus/Minus buttons	Set the value of the currently selected parameter at the next <i>zero crossing</i> .
Any Editor	Plus/Minus	Scroll through the currently selected parameter's list of values in regular or logical increments (varies with each parameter).
	2 leftmost soft buttons	Reset MIDI transposition to 0 semitones. Double-press again to go to previous transposition.
	Center soft buttons	Select Utilities menu (MIDIScope, Stealer, etc.).
	2 rightmost soft buttons	Sends all notes/controllers off message on all 16 channels (same as Panic soft button).
	Left/Right cursor buttons	Toggle between Play and Stop of current song.
	Up/Down cursor buttons	Toggle between Play and Pause of current song.
Save Dialog	Plus/Minus buttons	Toggle between next free ID and original ID.

Table 3-1 Double Button Presses

## Intuitive Data Entry

Many parameters have values that correspond to standard physical controllers. In many cases, you can select these values “intuitively,” rather than having to scroll through the **Control Source** list. This is done by selecting the desired parameter, then holding the ENTER button while moving the desired physical control.

For example, on the LAYER page in the Program Editor, you can set the range of the currently selected layer as follows: use the cursor buttons to move the cursor to the value for the LoKey parameter, press (and hold) the **Enter** button, then press the note you wish to be the lowest note for the currently displayed layer. The note you triggered (it has to be between C 0—C 8) will appear as the value for the LoKey parameter. Repeat the process for the HiKey parameter.

Another example: select Program 199 while in Program mode. Press **Edit** to enter the Program Editor. Press the PITCH soft button to select the PITCH page. Move the cursor to the Src1 parameter. Hold the **Enter** button, and move the Pitch Wheel. **PWheel** will be selected as the value for Src1.

You can also use the keyboard to choose control sources, since most key numbers correspond to a value on the control source list. If you have a certain control source that you use over and over (for example, LFO1), this can be the quickest way to enter its value. To do this: highlight a parameter which uses a value from the control source list, hold down **Enter**, then strike the key corresponding to the control source you want to choose. LFO1, for example, is assigned to B5. You'll find a complete chart of these keyboard shortcuts at the end of Chapter 4 of the *Musician's Reference*.

Also, for almost every parameter, you can hold the **Enter** button and move the Data Slider to run through the range of values for the currently selected parameter. This is not as precise as the Alpha Wheel, but much faster.

## Changing the Current Layer in Multi-Layer Programs

When editing a multi-layer program (including drum programs), you can quickly switch between layers by holding the **Enter** button, then striking a key. The K2661 will change the current layer to that key's layer. If the key is part of more than one layer, subsequent key strikes will cycle through each layer that has that key in its range.



***Note:** This method for changing the current layer in a multi-layer program will NOT work if the currently highlighted parameter has a note number or control source for its value. In this case, the key you strike will function as described in Intuitive Data Entry, above.*

## Search

There's a convenient way to find any string of characters within the currently selected list, or range of values. Hold the **Enter** button and press any of the numeric buttons. A *dialog* like the one below appears.

```
<>KbdNaming:Off
```

```
Search string:  Soul Rebels
```

```
Delete Insert <<< >>> OK Cancel
```

Type in the string of characters you want to find. For example, if you're looking at the program list and you want to find all programs containing the word "Horn," you would type h-o-r-n. This function is not case-sensitive; it will find upper and lower case characters regardless of what you type.

When you've typed the string of characters you want to find, press **Enter**. The K2661 searches through the current list of objects or values, finds all items that match the string of characters you typed, and displays the first one it finds. Hold **Enter** and press one of the **Plus/Minus** buttons to search for the next higher- or lower-numbered object that contains the string of characters.

The string you select remains in memory. You can store and select a string of characters with each of the numeric buttons. Hold **Enter** and press one of the numeric buttons at any time to select that string for a search. When the string appears, you can change it, or just press **Enter** to find that string.

## Renaming Multiple Objects

You can automatically rename multiple RAM objects with one operation, by substituting one string of characters with another.

1. Press the **Master** mode button to enter Master mode.
2. Press the **Object** soft button, then press the **Name** soft button. The NAME page appears, showing a list of all RAM objects.

- ```
<>KbdNaming:Off
```

Delete Insert &lt;&lt;&lt; &gt;&gt;&gt; OK Cancel

- ```
<>KbdNaming:Off
```

Delete Insert &lt;&lt; &gt;&gt; OK Cancel

- ```
<>KbdNaming:Off
```

Delete Insert &lt;&lt; &gt;&gt; OK Cancel

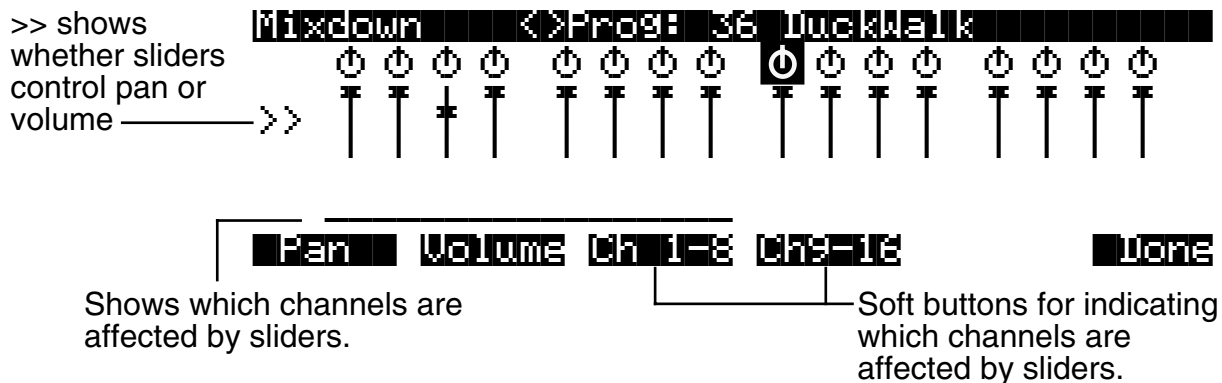
- 3-8**

## Mixdown and MIDI Faders Pages

There are two buttons below the **Solo** button, labeled **Mixdown** and **MIDI Faders**. These call up pages that let you use the sliders to control MIDI Pan and Volume, or any MIDI Controller that you choose. This is useful for real-time control over setups.

### Mixdown Page

This page temporarily turns the K2661's 8 sliders into MIDI pan and volume controls. You can get to this page from any mode by pressing the **Mixdown** button, located below the **Solo** button. Press **Mixdown**, and the display looks similar to what you see on the Song-mode MIX page. However, there are a few differences. On the Mixdown page, the sliders relate to MIDI channels instead of tracks. If you highlight 1-8 for example, moving slider D will affect volume or pan for MIDI Channel 4 while you're on this page.



On the Mixdown page, pan and volume messages are sent to the internal K2661 sounds, to MIDI Out, or both. All Mixdown-page channels have the same destination (local, MIDI, or both), which is determined by the Control parameter on the MIDI-mode TRANSMIT page. You can assign the sliders to control either volume or pan. The soft buttons at the bottom of the Mixdown page let you select pan or volume control, on channels 1-8 or 9-16, allowing you to temporarily use the 8 sliders as pan or volume faders. Keep in mind that MIDI volume is a channel-specific message. Therefore, if you have more than one zone in a setup assigned to the same channel, moving any one of the associated sliders will affect all those zones.

### MIDI Faders Pages

The button just below **Mixdown** is called **MIDI Faders**. The MIDI Faders page allows you to create a preset configuration of any Controller assignment for each slider, along with an initial preset value for that Controller. For each slider, you assign the MIDI channel, MIDI Controller number, and a value. You can enter the value using any normal data entry method, or by moving the corresponding slider on keyboard models. You can also assign several sliders to the same channel, but assign each one to a different Controller number.

There are four different configurations (pages 1-4), which you select with the soft buttons. Each configuration remembers the last value you set for each slider. If you want to send those values without moving the slider, press the **Send** button.

The available values for the Channel parameter on this page are **1–16** (local and MIDI), **1L–16L** (local only, no MIDI), and **1M–16M** (MIDI only, no local).

| MIDI Faders: Page1 |       |       |       |      |      |    |    |    |
|--------------------|-------|-------|-------|------|------|----|----|----|
| Chan :             | 9     | 10    | 11    | 12   | 13   | 14 | 15 | 16 |
| Ctl1 :             | 6     | 6     | 6     | 6    | 6    | 6  | 6  | 6  |
| Value:             | 50    | 0     | 50    | 0    | 50   | 0  | 50 | 0  |
|                    | ↑     | ↑     | ↑     | ↑    | ↑    | ↑  | ↑  | ↑  |
| Page1              | Page2 | Page3 | Page4 | Send | Done |    |    |    |

## Quick Song Recording and Playback

There are three buttons—labeled **Record**, **Play/Pause**, and **Stop**—below the mode selection buttons. They control the recording and playback of songs from any mode; you don't have to be in Song mode to record or play back.

Using these buttons affects the current track of the current song—that is, the song and track that were selected the last time you were in Song mode. When you record, the recording track and recording mode are determined by the current settings in Song mode; likewise for the playback mode when you're playing a song.

When the sequencer status is STOPPED (neither the **Record**-button LED nor the **Play/Pause**-button LED is lit or flashing), press **Record** to put the sequencer in REC READY status. The **Record**-button LED lights (red). Then press **Play/Pause** to start recording. The **Play/Pause**-button flashes (green) to indicate the tempo. Any countoff is determined by the current Song-mode setting for the CountOff parameter. Press **Play/Pause** or **Stop** to end recording and go to the Save dialog, where you can save the song, or trash it.

When the sequencer status is STOPPED, press **Play/Pause** to begin playing the current song. Press **Play/Pause** again to pause playback, and again to resume. Press **Stop** to end playback.

# Chapter 4

## The Operating Modes

In this chapter we'll discuss the theory behind the mode concept, and we'll describe the basic operating features of each mode.

### What the Modes Are

The modes exist to make the K2661 logical to work with. With as many performance and programming features as the K2661 has, it's helpful to break them into groups. These groups are called modes. There are eight primary modes (plus Sample mode and Live mode); they're described briefly in the section called *Using the Modes* on page 4-3. Chapters 6 through 13 are dedicated to explaining each primary mode in turn. Chapter 14 describes Sample mode and Live mode.

Each mode is named for the kind of operations you perform while in that mode, and each mode's editor (if any) contains all of the parameters related to editing the type of *object* found in that mode. In Setup mode, for example, you select setups (and only setups) for performance or editing. All of the setup-editing parameters are grouped together on the Setup-Editor page, which is accessible through Setup mode.

### Selecting Modes

When the K2661 is on, it's almost always operating in one of the eight primary modes represented by the LED-highlighted buttons beneath the display—or in one of the editors corresponding to the current operating mode. There are two exceptions: Sample mode (and the Sample Editor), and Live mode. Pressing one of the mode buttons selects the corresponding mode. This is the mode's entry level. At the entry level, the LED of the selected mode is lit. Only one mode can be selected at a time.

From any primary mode, you can get to any other primary mode simply by pressing one of the mode buttons. If you're in an editor, however, you must press **Exit** to return to the mode's entry level before selecting another mode.

All of the modes except Disk mode and Live mode give you access to one or more editors for changing the values of the parameters within that mode. Press the **Edit** button to enter the editor of the currently selected mode. When you do this, the mode LED goes out.

It's possible to enter another mode's editor without leaving the currently selected mode. For example, if you press **Edit** while in Setup mode, you'll enter the Setup Editor. The Setup-Editor page will appear, and the LocalPrg parameter will be highlighted by the cursor. If you press **Edit** again, you'll enter the Program Editor, where you can edit the currently selected program. While you can edit and save programs as you normally would, you're still in Setup mode, and you can't select another mode at this point. When you exit the Program Editor, you'll return to the Setup-Editor page. Press **Exit** again, and you'll leave the Setup Editor, returning to the Setup-mode page.

This method of entering editors works for any parameter whose value is an editable object (program, setup, keymap, FX preset, etc.)

## The Operating Modes

### Selecting Modes

The following table lists the procedures for moving between modes and editors. Note that the **Exit** button won't always take you where the table says it will; it often depends on how you got where you are. The table assumes that you've entered a given editor via its corresponding mode. You'll always return to Program mode eventually if you press **Exit** repeatedly.

| Current Mode/<br>Editor Status | Available Modes/<br>Editors | How to Get There                                                                  |
|--------------------------------|-----------------------------|-----------------------------------------------------------------------------------|
| Any mode                       | All other modes             | Press corresponding mode button                                                   |
| Program mode                   | Program Editor              | Press <b>Edit</b>                                                                 |
|                                | Sample mode                 | Press <b>Sample</b> soft button                                                   |
| Program Editor                 | Program mode                | Press <b>Exit</b>                                                                 |
|                                | Studio Editor               | On KDFX page: select Studio parameter; press <b>Edit</b>                          |
|                                | Keymap Editor               | On KEYMAP page: select Keymap parameter; press <b>Edit</b>                        |
| Keymap Editor                  | Program Editor              | Press <b>Exit</b>                                                                 |
|                                | Sample mode                 | Press the <b>MIDI</b> mode button                                                 |
|                                | Sample Editor               | Select Sample parameter; press <b>Edit</b>                                        |
| Sample Editor                  | Keymap Editor               | Press <b>Exit</b>                                                                 |
| Setup mode                     | Sample mode                 | Press <b>Sample</b> soft button                                                   |
|                                | Setup Editor                | Press <b>Edit</b>                                                                 |
| Setup Editor                   | Setup mode                  | Press <b>Exit</b>                                                                 |
|                                | Program Editor              | On CH/PRG page: select LocalPrg parameter; press <b>Edit</b>                      |
|                                | Studio Editor               | On KDFX page: select Studio parameter; press <b>Edit</b>                          |
|                                | Song Editor                 | On COMMON page, select Song parameter; press <b>Edit</b>                          |
| Quick Access mode              | Sample mode                 | Press <b>Sample</b> soft button                                                   |
|                                | Quick Access Editor         | Press <b>Edit</b>                                                                 |
| Quick Access Editor            | Quick Access mode           | Press <b>Exit</b>                                                                 |
|                                | Program Editor              | Set value of Type parameter to <b>Program</b> ; select program; press <b>Edit</b> |
|                                | Setup Editor                | Set value of Type parameter to <b>Setup</b> ; select setup; press <b>Edit</b>     |
| Effects mode                   | Studio Editor               | Press <b>Edit</b>                                                                 |
| Studio Editor                  | Previous mode               | Press <b>Exit</b>                                                                 |
|                                | FX Preset Editor            | Select FX preset block; press <b>Edit</b>                                         |
| FX Preset Editor               | Studio Editor               | Press <b>Exit</b>                                                                 |
| MIDI mode                      | Program Editor              | On CHANNELS page: select Program parameter; press <b>Edit</b>                     |
|                                | Setup Editor                | On TRANSMIT page: select CtlSetup parameter; press <b>Edit</b>                    |
|                                | Velocity Map Editor         | On TRANSMIT or RECEIVE page: select Veloc(ity)Map parameter; press <b>Edit</b>    |
|                                | Pressure Map Editor         | On TRANSMIT or RECEIVE page: select Press(ure)Map parameter; press <b>Edit</b>    |
| Master mode                    | Sample mode                 | Press <b>Sample</b> soft button                                                   |
|                                | Velocity Map Editor         | Select VelTouch parameter; press <b>Edit</b>                                      |
|                                | Pressure Map Editor         | Select PressTouch parameter; press <b>Edit</b>                                    |
|                                | Intonation Table Editor     | Select Intonation parameter; press <b>Edit</b>                                    |
| Song mode                      | Song Editor                 | Select CurSong parameter; press <b>Edit</b>                                       |
|                                | Program Editor              | Select Program parameter; press <b>Edit</b>                                       |
| Sample mode                    | Sample Editor               | Select Sample parameter; press <b>Edit</b>                                        |
|                                | Previous mode               | Press <b>Exit</b>                                                                 |
| Most editors                   | Previous mode or editor     | Press <b>Exit</b>                                                                 |



## Nested Editors

Starting at the Program-mode level, there are three “nested” editors, each related to the parameters that make up different components of a program. The first is the Program Editor, which you enter when you press **Edit** while in Program mode. Programs consist, among other things, of *keymaps*; they determine which samples play on which keys. Keymaps can be edited as well. The Keymap Editor is entered from within the Program Editor, by selecting the KEYMAP page with the soft buttons, then pressing **Edit**.

Similarly, keymaps consist of *samples*, which also can be edited. The Sample Editor is entered from the Keymap Editor, by selecting the Sample parameter and pressing **Edit**. When you enter the Sample Editor, you’ve worked through three nested levels of editors, all related to the components that make up a program. And in fact, you’re still in Program mode (if that’s where you started from). Pressing **Exit** while in the Sample Editor will return you to the Keymap Editor. Pressing **Exit** again will return you to the KEYMAP page of the Program Editor. Once more, and you’re back to Program mode’s entry level.



*Note: You can also get to the Sample Editor directly from Sample mode (by pressing **Edit**). If you’re editing a single sample, this might make more sense, since it’s more direct. If you’re planning on editing a number of samples within a given keymap, however, you’ll find it more convenient to enter the Sample Editor through the Program and Keymap Editors, because once you’ve finished with one sample, you can easily return to the KEYMAP page and select another sample within the keymap.*

### Finding Square One

If, at any time, you don’t know where you are, and the mode LEDs are all unlit, press **Exit** one or more times. This will return you to the entry level of whatever mode you were in, and if you press **Exit** enough times, you will always return to Program mode, the startup mode. If you’ve made any changes, you’ll be asked whether you want to save before leaving any editor. Press the **No** soft button or the **Exit** button if you don’t want to save. If you want to save, press the **Rename** or **Yes** soft button, and you’ll see the Save dialog, which is described in *Saving and Naming* on page 5-3.

## Using the Modes

You can play your K2661 regardless of the mode you’re in. In fact, the only times you can’t play it are when you’re in the middle of a disk operation (loading, saving, formatting) or a SMDI sample transfer. With these two exceptions, the K2661’s MIDI response is almost always active. Even so there are three modes that are more performance-oriented than the others. These are Program, Setup, and Quick Access modes. We’ll describe each of the eight modes briefly in this section.

### Program Mode

The K2661 starts up in Program mode, where you can select, play, and edit programs. The Program-mode entry-level page shows the currently selected program, as well as a small segment of the program list. Program mode gives you access to Sample mode via the **Sample** soft button.

The Program, Keymap, and Sample Editors are nested within Program mode. They take you to the core of the K2661’s sound editing parameters. We’ll discuss them in Chapters 6 and 14.

### Setup Mode

Setup mode lets you select, play, and edit setups. Setups consist of up to eight separate zones, split or overlapping, each having its own program, MIDI channel, and control parameters. Setups are great for performance situations, whether you’re playing multiple K2661 programs or controlling additional synths connected to the K2661’s MIDI Out port. Chapter 7 describes Setup mode and the Setup Editor in detail.

If you're using a different MIDI controller, you can make use of Setup mode even if your MIDI controller can transmit on only one MIDI channel at a time. To do this, go to the RECEIVE page in MIDI mode (by pressing the **RECV** soft button while in MIDI mode), and set the Local Keyboard Channel parameter to a value that matches the transmit channel of your MIDI controller. When you select Setup mode, the K2661 will interpret incoming MIDI information according to the settings for the currently selected setup. See the discussion of the Local Keyboard Channel parameter in Chapter 10 for details.

Setup mode also gives you access to Sample mode via the **Sample** soft button.

## Quick Access Mode

Another feature for live performance, Quick Access mode enables you to combine programs and setups into banks of ten entries. Each of these programs or setups can be selected with a single alphanumeric button. Different banks are selected with the **Chan/Bank** buttons. There's a selection of factory preset banks, and you can use the Quick Access Editor to create your own banks and store them in RAM. There's a full description in Chapter 8.

Quick Access mode gives you access to Sample mode via the **Sample** soft button. You can also use Quick Access banks as a way to remap incoming or outgoing Program Change commands. See *Program Change Formats* on page 10-11.

## Effects Mode

Effects mode sets the behavior of the KDFX effects processor. The Effects-mode page lets you tell the K2661 how to select preset studios (programmed effects configurations) when you change programs or setups, or lets you choose a preset studio that's applied to *every* K2661 program. The Studio Editor allows you to tweak the preset studios, and create your own. Chapters 9 and 15 show you how. You can also listen to the sounds of various effects while in Effects mode, without selecting different programs.

## MIDI Mode

You'll use MIDI mode to configure the K2661's interaction with other MIDI instruments, by setting parameters for transmitting and receiving MIDI. You'll also use it to configure your K2661 for multi-timbral sequencing. On the CHANNELS page, you can assign a program to each channel, and enable or disable each channel's response to three types of MIDI control messages: Program Change, volume and pan. You can also override program output assignments, and adjust overall program gain. See Chapter 10.

## Master Mode

Master mode, described in Chapter 11, contains the parameters that control the entire K2661. Global settings for tuning, transposition, velocity and aftertouch sensitivity, and other preferences are adjusted here. You can also get to the SampleMode page from here, using the **Sample** soft button.

## Song Mode

Song mode enables you to play sequences (songs) stored in the K2661's RAM, and provides a fully featured sequencer that you can use to record songs. You can also record multi-timbrally via MIDI, or load standard MIDI files (Type 0 or 1). The Song Editor enables you to modify existing RAM sequences, do step recording, and create arrangements (by linking two or more songs together). See Chapter 12.

## Disk Mode

Disk mode lets you load and save programs and other objects using the K2661's SmartMedia drive or a disk (or CD-ROM drive) connected to the K2661's SCSI port. See Chapter 13.

# Chapter 5

## Editing Conventions

### Introduction to Editing

Programming (editing) the K2661 always involves three basic operations: mode selection, navigation, and data entry.

First, select the mode that relates to the object you want to edit—a program, a setup, etc. Then select the object you want to edit, and press the **Edit** button to enter the editor within that mode. An editor contains all the parameters that define the object you're programming.

Next, you navigate around the editor's page(s) with the soft buttons, and select parameters with the cursor (arrow) buttons. When you've selected a parameter (its value is highlighted by the cursor), you can change its value with one of the data entry methods. When you change a value, you'll normally hear its effect on the object you're editing. The K2661 doesn't actually write your editing changes to memory until you save the object you're working on. It then allows you to choose between writing over the original object, or storing the newly edited version in a new memory location.

### What's an Object?

If you've been wondering what we mean by the term "object," it's an expression we use for anything that can be named, saved, deleted, or edited. Here's a list of all the types of objects:

|                           |                                                                                                                                                                                                                                                                                 |
|---------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Samples</b>            | Digital recordings of instrumental sounds or waveforms. Samples actually have two separate parts: the actual sample data and the sample header information, which contains start, alternative start, loop, and end points, as well as other information like tuning and volume. |
| <b>Keymaps</b>            | Collections of samples assigned to specific velocity ranges and/or keys.                                                                                                                                                                                                        |
| <b>Programs</b>           | Factory-preset or user-programmed sounds stored in ROM or RAM. A program is one or more layers of sound, with programmable DSP functions applied to the keymaps within each layer.                                                                                              |
| <b>Setups</b>             | Factory-preset or user-programmed MIDI performance presets consisting of up to eight zones, each with its own program, MIDI channel, and controller assignments, and (optionally) arpeggiation specifications.                                                                  |
| <b>Songs</b>              | Sequence files loaded into RAM, or MIDI data recorded in Song mode.                                                                                                                                                                                                             |
| <b>Studios</b>            | Factory-preset or user-programmed configurations of the K2661's onboard digital audio effects processor.                                                                                                                                                                        |
| <b>FX presets</b>         | Factory-preset or user-programmed data—including effects algorithms, levels and balance—that define the nature of studios.                                                                                                                                                      |
| <b>Quick Access banks</b> | Factory-preset or user-programmed banks of ten entries each, that store programs and setups for single-button access in Quick Access mode.                                                                                                                                      |
| <b>Velocity maps</b>      | Factory-preset or user-programmed curves that affect the K2661's response to, and MIDI transmission of, attack velocity values.                                                                                                                                                 |

|                          |                                                                                                                                                                                                                            |
|--------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Pressure maps</b>     | Factory preset or user-programmed curves that affect the K2661's response to, and MIDI transmission of, pressure (aftertouch) values.                                                                                      |
| <b>Intonation tables</b> | Factory preset or user-programmed tables that affect the intervals between the twelve notes of each octave.                                                                                                                |
| <b>Master tables</b>     | The values that are set for the global control parameters on the Master-mode page, as well as the settings for the parameters on the CHANNELS page in MIDI mode, and the programs currently assigned to each MIDI channel. |
| <b>Fader tables</b>      | MIDI Controller assignments defined on the MIDI Faders pages.                                                                                                                                                              |
| <b>Name tables</b>       | Contains a list of dependent objects needed by the other objects in a file at the time the file was saved.                                                                                                                 |
| <b>Macros</b>            | List of disk files to be loaded into the K2661's memory at start-up time.                                                                                                                                                  |

## Object Type and ID

The K2661 stores its objects in RAM using a system of ID numbers that are generally organized into banks of 100. Each object is identified by its object type and object ID; these make it unique. An object's type is simply the kind of object it is, whether it's a program, setup, song, or whatever. The object ID is a number from 1 to 999 that distinguishes each object from other objects of the same type. For example, within the 200s bank, you can have a setup, a program, and a preset effect, all with ID 201; their object types distinguish them. You can't, however, have two *programs* with ID 201.

| Object Type  | Object ID | Object Name   |
|--------------|-----------|---------------|
| Program      | 201       | Hot Keys      |
| Setup        | 404       | Silicon Bebop |
| Velocity Map | 1         | Linear        |
| Sample       | 3         | Hey Moe       |

ROM (factory preset) objects have ID numbers in a number of banks. When you save objects that you've edited, the K2661 will ask you to assign an ID. If the original object was a ROM object, the K2661 will suggest the first available ID. If the original object was a RAM object, you'll have the option of saving to an unused ID, or replacing the original object.

Objects of different types can have the same ID, but objects of the same type must have different IDs to be kept separate. When you're saving an object that you've edited, you *can* assign the same ID to an existing object of the same type, but if you do, the new one will be written over the old one. For example, if you assign an ID of 1 to a program you've edited, the K2661 will ask you if you want to "replace" the ROM program currently stored with that ID. We'll discuss this further in *Saving and Naming* on page 5-3.

Many parameters have objects as their values—the VelTouch parameter on the Master-mode page, for example. In this case, the object's ID appears in the value field along with the object's name. You can enter objects as values by entering their IDs with the alphanumeric pad. This is especially convenient for programs, since their ID numbers are the same as their MIDI program change numbers (when you're using the Extended or Kurzweil Program Change format—see *Program Change Formats* on page 10-11).

The object type and ID enable you to store hundreds of objects without losing track of them, and also to load files from disk without having to replace files you've already loaded. See *Memory Banks* on page 5-7 for more information on object type and ID.

## Saving and Naming

When you've edited an object to your satisfaction, you'll want to store it in RAM. There's a standard procedure for saving and naming, which applies to all objects.

You can press the **Save** soft button, of course, but it's easier to press the **Exit** button, which means "I want to leave the current editor." If you haven't actually changed anything while in the editor, you'll simply exit to the mode you started from. If you *have* made changes, however, the K2661 will ask you if you want to save those changes. This is the first Save dialog. A dialog is any display that asks a question that you need to answer before the K2661 can proceed.

The following diagram shows the Save dialog for the Program Editor. The top line tells you that you got to this dialog by pressing Exit. If you had pressed the Save soft button instead of Exit, you'd see Save in the top line, instead of Exit. If you were in another editor, the top line would indicate that as well.

```
EditProgram:Exit
Save changes to Righteous Piano?
Rename Cancel Yes No
```

The best way through this process is to press the **Rename** soft button. This takes you immediately to the naming dialog, where you assign a name to the object you're saving. You haven't saved yet, but you'll be able to after you've named the program.

```
EditProgram:Rename <>KbdNaming:Off
Program Name: Righteous Piano
Delete Insert <<< >>> OK Cancel
```

Like the Save dialog, the top line identifies the current editor and function, and also indicates whether the keyboard naming feature is enabled (see *Keyboard Naming* on page 5-5).

The cursor underlines the currently selected character. Press the **<<<** or **>>>** soft buttons to move the cursor without changing characters. Press an alphanumeric button one or more times to enter a character above the cursor. The characters that correspond to the alphanumeric buttons are labeled under each button. If the character that appears is not the one you want, press the button again. Press the +/- button on the alphanumeric pad to switch between upper and lower case characters.

Press **0** one or more times to enter the numerals 0 through 9. Press **Clear** (on the alphanumeric pad) to erase the selected character without moving any other characters. Press the **Delete** soft button to erase the selected character. All characters to the right of the cursor will move one space left. Press the **Insert** soft button to insert a space above the cursor, moving all characters to the right of the cursor one space to the right.

Press the **Cancel** soft button if you decide not to name the object. Press **OK** when the name is set the way you want to save it.

In addition to the letters and numerals, there are three sets of punctuation characters. The easiest way to get to them is to press one of the alphanumeric buttons to select a character close to the one you want, then scroll to it with the Alpha Wheel. Here's the whole list:

! " # \$ % & ' ( ) \* + , - . / 0 1 2 3 4 5 6 7 8 9


: ; < = > ? @ A through Z

[ \ ] ^ \_ ` a through z. (space).

Pressing the **Plus/Minus** buttons simultaneously is a short cut to the following characters:  
**0, A, a** and (space).

If you're wondering how we came up with this sequence of characters, it's composed of ASCII characters 33 through 122.

When you press **OK**, the final Save dialog appears, where you assign an ID to the edited object. If you change your mind about the name, press the **Rename** soft button for another try.

A screenshot of a black rectangular dialog box with the text 'Edit Program: Save' in white monospace font.A screenshot of a black rectangular dialog box. The text 'Save Righteous Piano as:' is in white, followed by a white rectangular field containing the text 'ID#200'.A screenshot of a black rectangular soft button with the word 'Object' in white monospace font.A screenshot of a black rectangular soft button with the word 'Rename' in white monospace font.A screenshot of a black rectangular soft button with the word 'Save' in white monospace font.A screenshot of a black rectangular soft button with the word 'Cancel' in white monospace font.

## ROM Objects

If the object you started from was a ROM (factory preset) object, the K2661 will automatically suggest the next available (unused) ID as the ID for the edited object. If that's the ID you want, press the **Save** soft button, and the object will be stored in RAM with that ID. Otherwise, you can select any ID from 1 to 999. This page also gives you the opportunity to return to the naming dialog (as described in the previous section) or, by pressing the **Object** soft button, to access the Object Utilities (described in Chapter 13).

If you select an ID that's already in use, the K2661 will tell you that you're going to replace the ROM object that's already been assigned that ID. If you don't want to do that, you can select a different ID. Or you can press the **Plus/Minus** buttons simultaneously to toggle between the ID that the K2661 suggested and the original ID. Or press the **Cancel** soft button to cancel the operation.

If you decide not to cancel or change the ID, and you press the **Replace** soft button, the K2661 will write your newly edited object over the existing ROM object. Actually, it only appears that way, since you can't truly write to ROM. The ROM object will reappear if you delete the newly edited object (there are soft buttons in each editor for deleting objects).

## RAM Objects

If the original object was a RAM object, the K2661 will assume you want to replace it, and will suggest the same ID as the original object (if it has an asterisk—\*—between its ID and its name, it's a RAM object). As with ROM objects, you can cancel, replace, or change the ID and save to an unused ID. If you replace a RAM object, however, it's definitely gone!

The keyboard naming feature makes naming objects convenient (and musical!). With keyboard naming enabled, you can use the keyboard (or your MIDI controller) to enter the name of the object you're modifying.

**On** requires you to move the cursor to enter each letter, just as you have to do when using the normal data entry methods for naming. **Adv** automatically moves the cursor one space to the right each time you strike a key, just like a typewriter or computer keyboard. This is the most convenient setting.

When you're in the Rename dialog, use either of the **Chan/Bank** buttons to change the keyboard naming state.

| Function                     | Key | Symbol                          | Function                                |
|------------------------------|-----|---------------------------------|-----------------------------------------|
|                              | A0  |                                 |                                         |
|                              |     |                                 |                                         |
|                              |     |                                 |                                         |
|                              |     |                                 |                                         |
|                              |     |                                 |                                         |
|                              |     |                                 |                                         |
|                              |     |                                 |                                         |
|                              |     |                                 |                                         |
|                              |     |                                 |                                         |
| Move cursor to start of name | C2  |                                 | Delete; move characters left one space  |
| Move cursor left one space   |     |                                 | Insert; move characters right one space |
| Move cursor right one space  |     |                                 |                                         |
| (Shift)                      |     | 1 !                             |                                         |
| (Space)                      |     | 2 @                             |                                         |
| a A                          |     | 3 #                             |                                         |
| b B                          |     |                                 |                                         |
| c C                          | C3  | 4 \$                            |                                         |
| d D                          |     | 5 %                             |                                         |
| e E                          |     |                                 |                                         |
| f F                          |     | 6 ^                             |                                         |
| g G                          |     | 7 &                             |                                         |
| h H                          |     | 8 *                             |                                         |
| i I                          |     |                                 |                                         |
| j J                          | C4  | 9 (                             |                                         |
| k K                          |     | 0 (zero) )                      |                                         |
| l L                          |     |                                 |                                         |
| m M                          |     | - (Hyphen) _ (Underscore)       |                                         |
| n N                          |     | = (Equals) +                    |                                         |
| o O                          |     | (Backspace)                     |                                         |
| p P                          |     |                                 |                                         |
| q Q                          | C5  | ; (Semicolon) : (Colon)         |                                         |
| r R                          |     | ' (Apostrophe) " (Quote)        |                                         |
| s S                          |     |                                 |                                         |
| t T                          |     | , (Comma) <                     |                                         |
| u U                          |     | . (Period) >                    |                                         |
| v V                          |     | / (Slash) ?                     |                                         |
| w W                          |     |                                 |                                         |
| x X                          | C6  | [ (Left bracket) ` (Back quote) |                                         |
| y Y                          |     | ] (Right bracket) \ (Backslash) |                                         |
| z Z                          |     |                                 |                                         |
| (Space)                      |     | (OK, Enter)                     |                                         |
| (Shift)                      |     |                                 |                                         |
| Move cursor left one space   |     |                                 | Delete; move characters left one space  |
| Move cursor right one space  |     |                                 | Insert; move characters right one space |
| Move cursor to end of name   | C7  |                                 |                                         |
|                              |     |                                 |                                         |
|                              |     |                                 |                                         |
|                              |     |                                 |                                         |
|                              |     |                                 |                                         |
|                              |     |                                 |                                         |
|                              |     |                                 |                                         |
|                              |     |                                 |                                         |
|                              | C8  |                                 |                                         |

Use the (Shift) keys or Sustain pedal to enter upper-case and special characters

## Deleting Objects

Within most editors, there are soft buttons for deleting objects. When you want to delete an object, press the **Delete** soft button, and the K2661 will ask you if you want to delete the object. (At this point in the dialog, you can select another object with any of the data entry methods.) Press **OK** if you want to delete it, or press **Cancel** if you don't. Although it seems that you can delete ROM objects, you can't actually do it. The K2661 will behave as if it's deleting the ROM object, but it will still be there the next time you select it. (What actually happens is that the ROM object is copied to RAM as soon as you press **Edit**, and when you "delete" the ROM object, you're actually deleting the RAM copy. The original ROM object remains in memory.)

RAM objects, on the other hand, are *gone* when you delete them! If you've "replaced" a ROM object by saving a RAM object with the same ID, the ROM object is invisible, but still there. Deleting the RAM object stored at the same ID will restore the ROM object.

You'll often delete objects to gain RAM space, or to organize the *memory banks* before saving objects to disk. To delete multiple objects, use the Delete Objects utility available in Master mode. It's described on page 11-18.

## Dependent Objects

A dependent object is an object that's linked in memory with at least one other object. For example, if you create a setup that uses a program that you also created, that program is a dependent object of the setup.

When you start to delete an object that has dependent objects, the Delete dialog gives you a choice: Delete dependent objects? If you press **Yes**, the K2661 will delete the object and all its dependent objects when you execute the Delete function. In our example, if you were deleting the setup you created, and you chose to delete dependent objects, the dependent program would get deleted as well. If you press **No** at the Delete dependent objects? prompt, the K2661 deletes only the object, but keeps the dependent objects. In our example, the setup would get deleted, but the dependent program would remain.

When deleting objects and their dependents, the K2661 deletes only those dependent objects that aren't dependent on other objects that you're *not* deleting. For example, suppose you have two setups that contain the same program. If you delete one of the setups, and delete dependent objects with it, the setup gets deleted, but the program that's contained in the other setup remains in memory.



## Memory Banks

To help organize the storage of your edited objects, the K2661's memory is divided into ten banks, each of which stores objects having IDs within a certain range. Objects within the same range of IDs are stored in the same memory bank, regardless of their types. The banks are in increments of 100, that is, objects with IDs from 1 through 99 are stored in the first bank, IDs from 100 to 199 in the second bank, and so on. We refer to them simply as the "Zeros bank," "100s bank," "200s bank," etc. If you save an object as ID 203, for example, it's in the 200s bank.

You can store up to 100 objects of each type in each memory bank. The number of objects of a given type that can be saved in a memory bank depends on its type. For example, you can store 20 Quick Access banks in each memory bank. As you begin to save objects that you've edited, you'll notice that the IDs suggested by the K2661 sometimes increase in large chunks—from 219 to 300, for example. This is due to the limit to the number of objects of a given type that can be stored in a single memory bank. This limit can be important in terms of organizing your objects for storing to disk. Check out the section called *Storing Objects in the Memory Banks* on page 13-34 for lists of how many objects of each type can fit into each memory bank.

You'll want to think about organizing your edited objects in the memory banks when you start using the SmartMedia drive or a SCSI device to store your programs, samples, and other objects. Objects that are stored in the same memory banks can be easily stored in the same *file* on disk—which generally is a good idea. You can also store all the banks to one file by selecting "Everything" in the Disk-mode Bank dialog.

The memory banks work automatically, that is, you don't have to select the different banks to gain access to the objects stored in them. The K2661 selects the appropriate bank when you enter the object ID you want to work with. To select Program 201 while in Program mode, for example, just press **2, 0, 1, Enter** on the alphanumeric pad. The 200s bank is automatically selected, and the program list will show programs numbered in the 200s. If your MIDI controller can send program change commands from 0 to 127 or 1 to 128 only, you'll probably want to adjust the way the K2661 responds to program change commands. See the discussion of the Program Change Type parameter (ProgChgType) in Chapter 10.

When you do a save operation in Disk mode, you're creating a file to be saved to disk or SmartMedia. This will save either individually selected objects or an entire bank of objects from the K2661's RAM. If you choose to save a complete bank, then all objects with IDs in the range of the selected bank, regardless of type, are saved as part of the file. For example, if you save the 200s bank (objects with IDs from 200–299), then every object with an ID from 200 to 299 will be saved to the file.

This system makes it easy for you to keep track of everything you save. The first program you save, for example, will have an ID of 200 (unless you specify another ID). The first setup you create will also have an ID of 200 (since they're different *types* of objects, the IDs can be the same). If you were to save the 200s bank, both your program and your setup would be saved to the same file.

## Saving and Loading Files—Disk Mode

Saving to disk or SmartMedia simply involves selecting objects or a complete bank of objects to be stored as a single file. All objects with IDs within that range will be saved to the file. When you *load* a file, the K2661 asks you which bank will receive the file. You can load a file into any of the ten banks, regardless of the bank it was saved from. The K2661 will automatically reassign the object IDs. A file saved from the 200s bank, for example will be stored on disk with its objects numbered from 200–299. If you load it back into the 300s bank, its objects will be renumbered from 300–399.

See Chapter 13 for more information on loading and saving files.

## Special Button Functions

The Mode buttons and the **Chan/Bank Down** button have additional functions, depending on the mode or editor you're in. When you're in the Program or Setup Editor, they function according to the orange labeling under each button. They also work as track mutes on the MIX page of Song mode.

When you're in the Sample Editor, the **Program**, **Setup**, **Q Access**, **MIDI**, **Master**, and **Song** mode buttons function according to the orange labeling near each button. The table below describes all of the special button functions.

| Button                           | Mode or Editor                                                                                                                                                                                                                                                                                                                |                                                                                                                   |                                  |                                                                                                                                   |
|----------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------|----------------------------------|-----------------------------------------------------------------------------------------------------------------------------------|
| White<br>Orange<br>Light Grey    | Program Editor (Orange)                                                                                                                                                                                                                                                                                                       | Setup Editor (Orange)                                                                                             | Song Mode                        | Sample Editor (Light Grey)                                                                                                        |
| Program<br>Mute 1<br>Zoom-       | Mutes Layer 1 of current program, or mutes current layer of current drum program                                                                                                                                                                                                                                              | Mutes Zone 1 of current setup if 3 or fewer zones; mutes current zone of current setup if more than 3 zones       | On MIX page, mutes Track 1 or 9  | On TRIM and LOOP pages, decreases horizontal dimension of current sample in display                                               |
| Setup<br>Mute 2<br>Zoom+         | Mutes Layer 2 of current program, or solos current layer of current drum program                                                                                                                                                                                                                                              | Mutes Zone 2 of current setup if 3 or fewer zones; solos current zone of current setup if more than 3 zones       | On MIX page, mutes Track 2 or 10 | On TRIM and LOOP pages, increases horizontal dimension of current sample in display                                               |
| Q Access<br>Mute 3<br>Samp / Sec | Mutes Layer 3 of current program, or solos current layer of current drum program                                                                                                                                                                                                                                              | Mutes Zone 3 of current setup if 3 or fewer zones; solos current zone of current setup if more than 3 zones       | On MIX page, mutes Track 3 or 11 | Toggles between units used to identify location within sample— either number of samples from start, or time in seconds from start |
| Effects<br>FX Bypass             | Bypasses (mutes) current program's FX preset (plays program dry)                                                                                                                                                                                                                                                              | Bypasses (mutes) current setup's studio (plays studio dry)                                                        | On MIX page, mutes Track 4 or 12 |                                                                                                                                   |
| MIDI<br>Previous Pg<br>Gain -    | Successive presses take you back to four most recent editor pages; 5th press takes you to ALG page                                                                                                                                                                                                                            | Successive presses take you back to four most recent editor pages; 5th press takes you to CH/PRG page             | On MIX page, mutes Track 5 or 13 | On TRIM and LOOP pages, decreases vertical dimension of current sample in display                                                 |
| Master<br>Mark<br>Gain +         | "Remembers" current editor page, so you can recall multiple pages with Jump button; asterisk appears before page name to indicate that it's marked; unmark pages by pressing Mark when page is visible                                                                                                                        | Same as for Program Editor; pages common to both editors are marked or unmarked for <i>both</i> editors           | On MIX page, mutes Track 6 or 14 | On TRIM and LOOP pages, increases vertical dimension of current sample in display                                                 |
| Song<br>Jump<br>Link             | Jumps to marked pages in order they were marked                                                                                                                                                                                                                                                                               | Jumps to marked pages in order they were marked                                                                   | On MIX page, mutes Track 7 or 15 | Preserves interval between Start, Alt, Loop, and End points of current sample; press again to unlink                              |
| Disk<br>Compare                  | Negates effect of unsaved edits and plays last-saved (unedited) version of object being edited                                                                                                                                                                                                                                | Same as for Program mode; display reminds you that you're comparing; press any button to return to edited version | On MIX page, mutes Track 8 or 16 |                                                                                                                                   |
| Chan / Bank<br>Layer / Zone      | In Program Editor, these two buttons scroll through layers of current program; in Effects Editor, scroll through FX presets; in Keymap Editor, scroll through key ranges of current keymap; in Setup Editor, scroll through zones of current setup; in Quick Access mode, scroll through entries in current Quick Access bank |                                                                                                                   | Change recording track           |                                                                                                                                   |
| Edit                             | Whenever cursor is highlighting an editable object or parameter, takes you to corresponding editor or programming page                                                                                                                                                                                                        |                                                                                                                   |                                  |                                                                                                                                   |

# Chapter 6

## Program Mode and the Program Editor

Programs are the K2661's performance-level sound objects. They're preset sounds equivalent to the patches, presets, voices, or multis that you find on other synths.

Program mode is the heart of the K2661, where you select programs for performance and editing. The K2661 is packed with great sounds, but it's also a synthesizer of truly amazing depth and flexibility. When you're ready to start tweaking sounds, the Program Editor is the place to start. But first there's a bit more general information about Program mode.

In Chapter 2 we briefly discussed the difference between VAST programs and KB3 programs. You'll remember that VAST programs contain up to 32 layers, each of which contains a keymap, which in turn consists of a number of samples assigned to a particular keyboard range—typically five or six notes, depending on the samples.

We mentioned drum programs, which are VAST programs with more than three layers. There's no real difference between "normal" VAST programs and drum programs—consequently this chapter doesn't make any further distinctions between them. There are also Triple Mode programs, where three layers of DSP information are used for one massive DSP chain.

OK, one further distinction: there's no keymap information about drum programs in the info box on the Program-mode page—there simply isn't room for information about more than three layers. If you're wondering why we even *have* the concept of a drum program, it's actually a carryover from the K2000, which had less processing power than the K2661, and required a special channel to handle more than three layers—and you need lots of layers, each with a different sound and keyboard range, to make a convincing drum program. The name stuck.

You'll also recall from Chapter 2 that KB3 programs use a much different architecture: no layers or algorithms, just a bunch of oscillators that start running as soon as you select a KB3 program. This keeps the K2661's sound engine rather busy, and that's why there's a special channel dedicated to KB3 programs; "regular" channels don't have the processing ability to generate that many voices on a constant basis. By default, Channel 1 is the KB3 channel, but you can make any channel the KB3 channel (with the KB3Chan parameter on the Master-mode page).

## Background

There's a lot to digest in this chapter, so it might help to explain how we've set it up. The next two sections give more detailed descriptions of the differences in structure between VAST programs and KB3 programs. Then, since there are several performance features (and a few issues) unique to KB3 programs, we'll talk about those (*KB3 Mode* on page 6-4). After that, there are descriptions of the Program-mode features that are common to both types of programs.

Then it's on to the Program Editor. When it comes to editing, there are more differences between the two types of programs than there are similarities, so there are two separate sections: *Editing VAST Programs* on page 6-11, and *Editing KB3 Programs* on page 6-48.

## VAST Program Structure

You might want to take a look at Figure 6-1 on page 6-3, which depicts the hierarchy of a VAST program, from individual samples all the way up to setups, which can contain up to eight programs.

Every VAST program contains at least one layer. A layer consists of a keymap and an algorithm for processing the samples contained in the keymap. Samples are stored in the K2661's ROM, or are loaded into Sample RAM via Disk mode, MIDI standard sample transfer, SMDI sample transfer, or by your own sampling efforts. Each sample is a separate digital recording of some kind of sound: musical, vocal, industrial, any sound at all. Individual samples are assigned to specific key ranges (from A 2 to D 3, for example), and are also assigned to be triggered at specific attack velocities. These assignments constitute the keymap.

When you trigger a note, the K2661 looks to the keymap of each layer of the currently active VAST program(s) to determine which samples to play. The sound engine then fetches the requested samples and generates a digital signal representing the sound of the samples. This signal first passes through the five DSP functions (more in triple mode) that make up the algorithm. It then passes through the KDFX effects processor, and finally appears—with some level of effects applied to it—at one or more of the audio outputs.

The layer is the VAST program's basic unit of polyphony, that is, each layer constitutes one of the 48 voice channels the K2661 can activate at any time. If you have a program that consists of two layers covering the note range from A 0 to C 8, each key you strike triggers two voice channels.

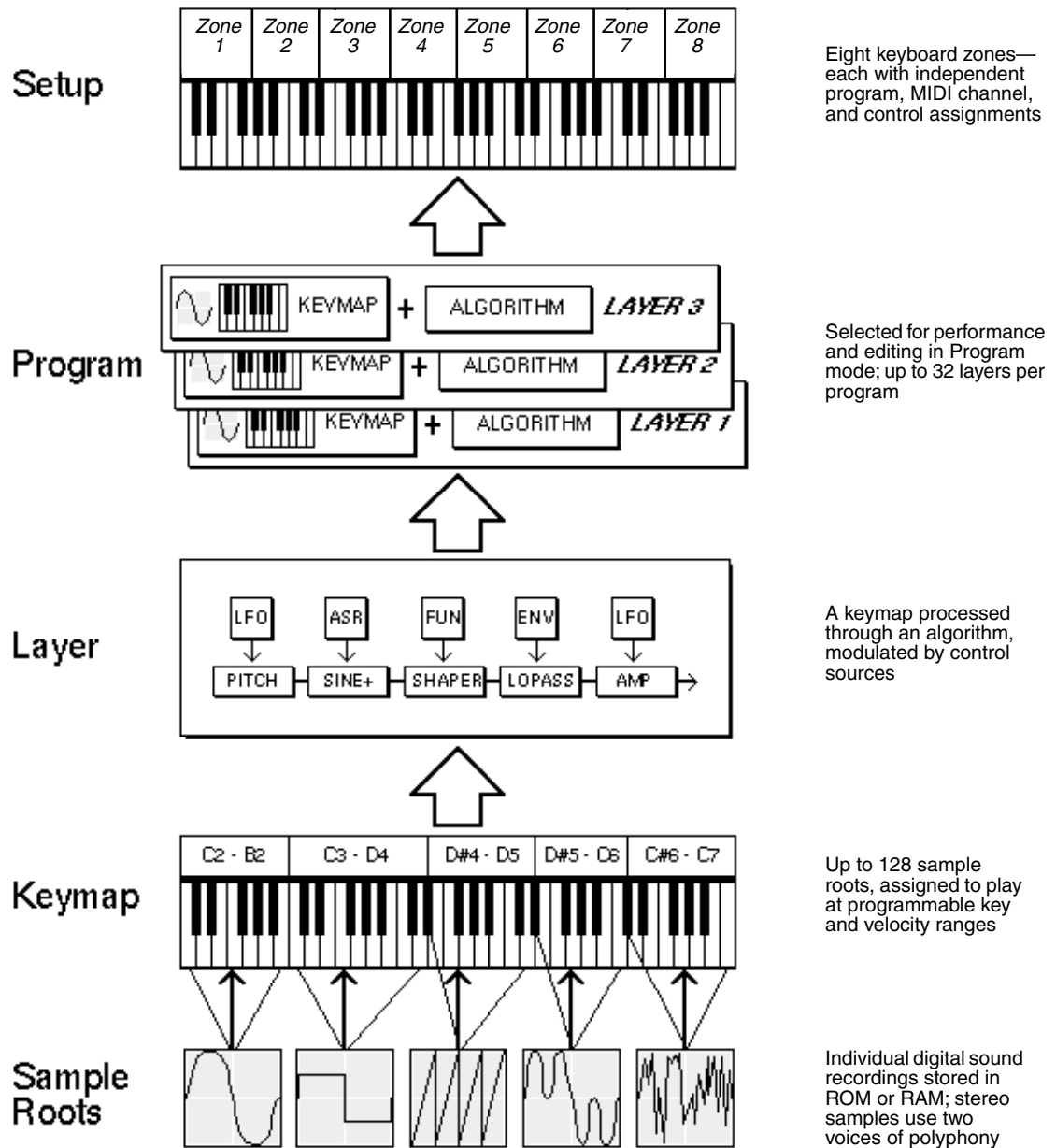


Figure 6-1 VAST Program Structure

## KB3 Program Structure

There's nothing quite like the sound of the classic Hammond™ B-3 tone wheel organ, especially when played through a Leslie™ rotating speaker system. We've done extensive testing and analysis with several tone wheel organs, and created our own models to emulate the unique tone wheel sound. We even took into account the way that older organs start to sound different (and arguably better) as their capacitors begin to leak—and we included a parameter that lets you vary the amount of grunge (leakage) in your sound.

We also recruited some very talented organ players to try out KB3 programs, and we've used their feedback to make the real-time controls as convenient and realistic as possible.

KB3 programs use oscillators to emulate the tone wheel sound. Each oscillator operates independently, and has its own pitch and amplitude control. You can control how many oscillators are used for a KB3 program. There are two oscillators per voice, for a total of 96. You can use up to 95 of them in a KB3 program (the 96th is reserved to produce key click). Because the oscillators start running as soon as you select a KB3 program, there are always voices available—unlike VAST programs, which start “stealing” notes when you reach the 48-voice polyphony limit. In other words, with a KB3 program, you can play and sustain more than 48 notes, and the K2661 will continue to play then all. With VAST programs, once 48 notes are on (for example, when you play and sustain a four-note chord in a 12-layer program), each new note that you play replaces one of the notes that was already on.

The oscillators—we'll call them tone wheels from here on—are divided into an upper and lower group. By default, the upper tone wheels use the samples in the K2661's keymaps (including your own RAM keymaps if you want) to generate sound, while the lower tone wheels use waveforms (like sine, square, or sawtooth). You can switch this around if you like, for even more variety.

## KB3 Mode

KB3 programs are different enough from VAST programs that we use the term KB3 mode to describe what's going on when you play a KB3 program. There are a few important points to consider if you want to get the most out of KB3 mode.

### KB3 Channel

As we mentioned in Chapter 2, you can play KB3 programs only on the KB3 channel, which you define on the Master-mode page. When you're in Program mode, this means that the current MIDI channel must match the KB3 channel, and when you're in Setup mode, any zone that uses a KB3 program must use the KB3 channel. If this isn't the case, the KB3 programs won't make any sound. If this happens in Program mode, all KB3 program names appear in parentheses, and the info box reminds you that you're not on the KB3 channel. If it happens in Setup mode, the display looks normal, but if you go into the Setup Editor, the LocalPrg parameter shows the KB3 program name in parentheses in every zone that's not on the KB3 channel.



**Note:** If you're using a Kurzweil PC88 to control your K2661, you shouldn't use Channel 1 as the KB3 channel. The PC88 sends MIDI Controller 90 on Channel 1 to select effects. In KB3 mode, the K2661 maps Controller 90 to internal controller 90, which controls the emulation of leakage that we mentioned earlier. You probably don't want your leakage level fluctuating every time the PC88 sends Controller 90.

### Real-time Controls in KB3 Mode

You have real-time control over many components of KB3 programs directly from the front panel. The sliders emulate the drawbars that are so essential to the tone wheel sound, while the buttons above them (they're called the **Mute** buttons, because they normally mute and solo zones in Setup mode) can control the KB3 effects: Leslie, vibrato, chorus, and percussion (key click).

When you're in Program mode, the **Mute** buttons always control KB3 effects. In a setup containing a KB3 program, if you want the **Mute** buttons to control KB3 effects, you'll have to edit the setup, because in Setup mode, the **Mute** buttons mute and unmute zones by default.

- 1. Go to Setup mode, and select the setup you want to edit. Press **Edit**.
- 2. Press either **more** soft button until you see the **COMMON** soft button. Press it, and your display should look like this:

```
EditSetup:COMMON All zones
Song : 0 None Sync : Off
Mutes : Zone Mutes
<more COMMON ARPEG RIBCFG more>
```

- 3. Select the Mutes parameter and change its value to **KB3 Control**.
- 4. Don't forget to save.

### Playing KB3 Programs

One of the standard performance features of many tone wheel organs is the set of drawbars for emulating the stops on a pipe organ. Moving the drawbars controls the amplitude of either the fundamentals or the harmonics of the notes (out to increase amplitude, in to decrease it).

The sliders and Mod Wheel serve as the nine drawbars found on most tone wheel organs. Pushing the sliders up is the equivalent of pushing the drawbars in (removing fundamentals or harmonics). The Mod Wheel is the other way around, since you're probably used to the Mod Wheel being off when it's down, and on when it's up. So remember, for the Mod Wheel, down (off) is like pushing the drawbar in (decreasing amplitude), and up (on) is out (increasing amplitude).

| Subharmonics |          | Fundamental | Harmonics |          |          |          |          |           |
|--------------|----------|-------------|-----------|----------|----------|----------|----------|-----------|
| 16'          | 5 1/3'   | 8'          | 4'        | 2 2/3'   | 2'       | 1 3/5'   | 1 1/3'   | 1'        |
| Slider A     | Slider B | Slider C    | Slider D  | Slider E | Slider F | Slider G | Slider H | Mod Wheel |

Table 6-1 Standard Drawbar Settings for the Hammond B3

### KB3 Mode Buttons (Mute Buttons)

When the **Mute** buttons are enabled for KB3 control, their LEDs indicate the status of the various effects for the current KB3 program. This status is saved as part of each program. You

can change the effects in real time by pressing the buttons (or by sending the appropriate MIDI Controller values from your MIDI controller).

In normal operational modes, using the **Mute** buttons to change a program's KB3 effects doesn't affect the program; the effects return to their programmed settings the next time you select the program. If, however, you're in an editor when you change the effects, you're actually editing the program. If you like the changes, you can save the program with the new KB3 effects settings. If you don't like the changes, you can exit without saving, and the program will revert to its previous settings.

The **Mute** buttons also send MIDI Controller information to the K2661's MIDI Out port. See Column 2 of Table 6-3 to check which Controller numbers the buttons send.

Of course, you can change the programmed settings for the KB3-mode buttons. For each of the buttons, there's a corresponding parameter in the Program Editor.

|   | Effect Category | Button Name        | Corresponding Page and Parameter | Comments                    |
|---|-----------------|--------------------|----------------------------------|-----------------------------|
| 1 | Rotary          | Fast / Slow        | MISC: SpeedCtl                   |                             |
| 2 | Vibrato         | On / Off           | MISC: VibChorCtl                 |                             |
| 3 |                 | Chorus / Vibrato   | MISC: VibChorSel                 | Disabled if Button 2 is off |
| 4 |                 | Depth 1 / 2 / 3    | MISC: VibChorSel                 | Disabled if Button 2 is off |
| 5 | Percussion      | On / Off           | PERC: Percussion                 |                             |
| 6 |                 | Volume Loud / Soft | PERC: Volume                     | Disabled if Button 5 is off |
| 7 |                 | Decay Fast / Slow  | PERC: Decay                      | Disabled if Button 5 is off |
| 8 |                 | Pitch High / Low   | PERC: Harmonic                   | Disabled if Button 5 is off |

**Table 6-2 KB3 Mode Buttons and Corresponding Parameters**

## MIDI Control of KB3 Programs

When you're playing a KB3 program from an external MIDI source, there are two things to keep in mind:

- Certain MIDI Controller numbers always control specific KB3 features
- The value of the LocalKbdCh parameter affects how KB3 programs respond to MIDI Controller messages

### Controller Numbers

Table 6-3 lists the MIDI Controller numbers that control KB3 features. The first column lists the Controller numbers that KB3 programs always respond to (the K2661 also sends these Controller numbers to its MIDI Out port when you're using the local keyboard channel—we'll say more about that on page 6-7). KB3 programs also respond to the Controller numbers in the second column; these are the Controller numbers that the Voce™ MIDI Drawbar Controller uses to control common tone wheel organ features. Whatever kind of external MIDI source you're using, you can use the MIDI controller numbers in either the second or third column to control the corresponding KB3 feature in the first column. For example, to control Drawbar 1, you can send either MIDI 6 or MIDI 12.



| KB3 Program Feature        | MIDI Controller Number |      |
|----------------------------|------------------------|------|
|                            | K2661                  | Voce |
| Drawbar1                   | 6                      | 12   |
| Drawbar2                   | 22                     | 13   |
| Drawbar3                   | 23                     | 14   |
| Drawbar4                   | 24                     | 15   |
| Drawbar5                   | 25                     | 16   |
| Drawbar6                   | 26                     | 17   |
| Drawbar7                   | 27                     | 18   |
| Drawbar8                   | 28                     | 19   |
| Drawbar9                   | 1                      | 20   |
| Expression Pedal           | 4                      | 8    |
| Percussion On/Off          | 73                     | N.A. |
| Percussion High/Low        | 72                     | 72   |
| Percussion Loud/Soft       | 71                     | 71   |
| Percussion Fast/Slow       | 70                     | 70   |
| Rotating Speaker Slow/Fast | 68                     | 68   |
| Vibrato/Chorus On/Off      | 95                     | 95   |
| Vibrato/Chorus Selector    | 93                     | 93   |
| Key Click Level            | 89                     | 89   |
| Leakage Level              | 90                     | 90   |

**Table 6-3 KB3 MIDI Controller Assignments**

### Local Keyboard Channel

The local keyboard channel enables the K2661 to receive MIDI information on a single channel, then rechannelize that information so you can play and control all eight zones of a setup, even if your MIDI source transmits on only one channel. When you're in Program mode, the local keyboard channel remaps incoming information to the K2661's current channel (the one shown in the top line of the display).

The LocalKbdCh parameter (on the RECEIVE page in MIDI mode) defines the local keyboard channel. When you're in Program mode, and playing a KB3 program, you may want to leave LocalKbdCh set to **None**, which is its default value. In this case the MIDI Controller messages for KB3 control listed in Table 6-3 are certain to work.

There are some possible disadvantages to this, however. First, the K2661 doesn't relay incoming MIDI to its MIDI Out port. Perhaps more importantly, if you change the channel on your MIDI source, the K2661 plays the program on the channel used by your MIDI source—regardless of the K2661's current channel. For example, if your MIDI source transmits on Channel 1, and you set the K2661's current channel to 2, you'll still play the program assigned to Channel 1. If that's the way you like it, there's no problem.

You may find it more convenient to use the local keyboard channel. In this case, the K2661 remaps incoming MIDI to the K2661's current channel, so in Program mode, you'll always play the program on the K2661's current channel. Incoming MIDI also gets sent to the K2661's MIDI Out port. On the other hand, in this case your MIDI source's transmitting channel must match the K2661's local keyboard channel for anything to work. Furthermore, for KB3 programs, some of the MIDI Controller numbers listed in Table 6-3 won't necessarily work.

Things are a bit different for playing setups. In this case, you *must* use the local keyboard channel to be able to play and control all of the setup's zones. Set LocalKbdCh to match the channel your external MIDI source is using (so if, for example, your MIDI source transmits on Channel 1, set LocalKbdCh to 1). All MIDI information that the K2661 receives on the local keyboard channel gets remapped to the channels and control destinations used by the zones in the setup.

The K2661 also remaps certain MIDI Controller messages that it receives on the local keyboard channel, so that they correspond (in most cases) to the default assignments for the K2661's physical controllers (Mod Wheel, sliders, ribbons, etc.). While this ensures that the physical controllers work in a consistent and relatively standard fashion for most setups and VAST programs, it necessitates a few adjustments to make incoming MIDI Controller messages control the KB3 features listed in Table 6-3. Without these adjustments, some of the KB3 features won't respond to MIDI Controller messages—this is true when you're playing programs as well as when you're playing setups.

To make everything work properly, you need to make sure that all the appropriate physical controllers are assigned for KB3 control. Physical controller assignments are handled by setups, and are defined by parameters on several pages in the Setup Editor (Table 10-1 on page 10-8 lists the physical controllers that get remapped by the local keyboard channel). Each zone of a setup has its own controller assignments. Programs don't have controller assignments, so they "borrow" them from a special setup that's reserved for that purpose. This setup is called the control setup; it's determined by the value of the CtlSetup parameter (on the TRANSMIT page in MIDI mode). You can read about control setups in detail on page 6-10.

When you're playing a setup on the local keyboard channel, each zone that uses a KB3 program must have the appropriate physical controller assignments. When you're playing a KB3 program, Zone 1 of the *control setup* must have the appropriate physical controller assignments.

There are two ways to configure a setup properly for KB3 control:

- Edit an existing setup, adjusting some of the physical controller assignments (see page 20-11).
- Use the KB3 setup that we've provided for your convenience. It's in a file on the SmartMedia card and CD-ROM that came with your K2661 (see page 20-12).

### **KB3 Control: A Summary**

Whenever you want to play a KB3 program, make sure that the KB3 setup is assigned as the control setup. When you want to play a setup containing a KB3 program, make sure that the zone that uses the KB3 program has the same physical controller assignments as the KB3 setup. When you're creating a setup that will use a KB3 program, use the KB3 setup as your starting point.

One final word—for now—about using the local keyboard channel: all the MIDI information received on the local keyboard channel also gets sent—*after being remapped*—to the K2661's MIDI Out port. There's a thorough discussion of the local keyboard channel beginning on page 10-7.

## The Program Mode Page

```

ProgramMode  XPose:US1  <>Channel:1
199 Default
209*Dig it al
1 Righteous Piano
2 Mondo Bass
3 Killer Drums
4 Weeping Guitar
Octav- Octav+ Panic Sample Chan- Chan+
  
```

The top line of the Program-mode entry-level page shows your location, the present MIDI transposition, and the current MIDI channel.

The info box at the left of the Program-mode page gives you information about the current program. For VAST programs of up to three layers, the info box shows the keymap assigned to each layer (Layer 1 on top, with additional layers below). The line beneath the name of the keymap indicates the keyboard range of that layer. In the diagram above, for example, there's one layer that extends from C 0 to C 8—the default range. The representation of these layer ranges is approximate; they're intended to let you know if you have a layered keyboard (lines overlapping) or a split keyboard (lines not overlapping).

For drum programs (VAST programs of more than three layers), the info box shows the number of layers in the program. For KB3 programs, the info box shows the keymap used for the upper tone wheels (or the lower tone wheels, if you have the Upper/LowerSwap parameter set to **On**).

The info box also tells you if the current program makes use of Triple Mode (see Chapter 12 of the *Musician's Reference*.)

### Program Names in Parentheses

While you are scrolling through different programs on various MIDI channels, you may occasionally encounter a program that doesn't make any sound, and whose name is in parentheses. The parentheses tell you that you have selected a KB3 program without being on the KB3 channel. KB3 programs use a different program architecture, and require many more voices to operate. Consequently, they use a special channel with enough throughput to handle those voice requirements. If you select a KB3 program without being on a KB3 channel, the K2661 cannot play the program. As shown in the following illustration, the KB3 channel is 1, while the K2661's current channel is 2. The selected program is disabled.

```

ProgramMode  XPose:US1  <>Channel:2
112 (Hammin Jammin)
113 (Funkie Munkie)
114 (Le's Rock)
115 (Jimmy, Jimmy, )
116 (Inagadadavida)
117 (Grind it, Gran)
KB3 Program
Mellow Vox
KB3Chan is Ch 1
Octav- Octav+ Panic Sample Chan- Chan+
  
```

The Program-mode page illustrates this in two ways: the program names are in parentheses, and the box at the left of the page includes the message "KB3 Chan is Ch 1." To fix this, you could either change the K2661's MIDI channel (with the **Chan/Bank** buttons), or make Channel 2 the KB3 channel (using the KB3Chan parameter in Master mode).

You can play any program on the KB3 channel, but you can play KB3 programs *only* on the KB3 channel.

If you've used a K2000 or K2500, you'll remember that you would occasionally see parentheses around the names of drum programs, as well. The K2661 can play drum programs—up to 32 layers—on any channel. The Drum Channel parameter no longer exists.

## Control Setup

The control setup defines what the K2661's physical controllers (wheels, sliders, pedals, etc.) do while you're in Program mode. It's a convenient way to apply the controller assignments in your setups globally. Just choose an existing setup to be the control setup, using the CtlSetup parameter on the MIDI-mode TRANSMIT page. Then while you're in Program mode, many of the controller assignments for Zone 1 of the control setup also apply to the programs you play (this is true for MIDI control messages as well, unless you have turned off MIDI control).

If you don't like the way the physical controllers work in Program mode, you can either select a different control setup, or edit the existing one. Any changes you make to the current control setup will also affect the way that setup works in Setup mode.

There are a few important points to remember about the control setup:

- The current control setup is used by *all* programs in Program mode.
- You cannot change the control setup from within Program mode.
- The control setup doesn't affect the *sound* of a program, only the assignments of certain physical controllers. The samples and keymaps assigned to a program are unaffected by the control setup. While you're in Program mode, the K2661 ignores the programs assigned to the setup that you choose as the control setup.
- Almost all of the VAST programs in the K2661 are designed to respond to the controller assignments in the default control setup (**97 ControlSetup**). Therefore you'll want to use **97 ControlSetup** as the control setup in most cases, with two exceptions. When you're using the local keyboard channel (that is, when the value of the LocalKbdCh parameter is anything but **None**) and playing a KB3 program from an external MIDI source, use a control setup that's configured for KB3 control, as described on page 6-8 (if LocalKbdCh is **None**, **97 ControlSetup** is fine for playing KB3 programs from an external MIDI source). If you want to change the controller assignments for any program or set of programs (either VAST or KB3), use a control setup that has the controller assignments you want.

See page 7-3 for a table listing the parameters that affect Program-mode controller assignments.

## The Soft Buttons in Program Mode

The **Octav-/Octav+** buttons are a shortcut for quick transposition in 12-semitone increments. You can use them to transpose the entire K2661 as much as three octaves up or down. The top line of the display shows the current amount of transposition (Xpose). Pressing both **Octave** buttons simultaneously returns the transposition to zero.

The **Octave** buttons transpose the K2661, as well as any MIDI devices connected to the K2661's MIDI Out port. Changing the transposition with the soft buttons also changes the corresponding setting on the MIDI-mode TRANSMIT page.

Pressing the **Panic** soft button sends an All Notes Off message and an All Controllers Off message on all 16 MIDI channels.

Use the **Chan-** and **Chan+** soft buttons to change the current MIDI channel. This changes the MIDI channel the K2661 uses internally, as well as the channel you're using to send information to other synths connected to the K2661's MIDI Out port (MIDI slaves). Changing the current MIDI channel with the soft buttons also changes the corresponding setting on the MIDI-mode TRANSMIT page.

The Program Editor is where you begin to modify the K2661's resident sounds, and to build your own sounds around samples (ROM or RAM) and /or waveforms. There's virtually no limit to the sounds you can create using the tools in the Program Editor.

To enter the Program Editor, start in Program mode and press **Edit**. The Program-mode LED will go out, and the ALG (Algorithm) page will appear.

```

Algorithm:1
  ↓      ↓      ↓      ↓      ↓
[PITCH] → [NONE] → [ ] → [ ] → [ ] → [ ] → [ ] → [ ] → [AMP] → [ ]
<more  ALG  LAYER  KEYMAP  PITCH  more>

```

Here's a method for jumping quickly to a specific layer in a program—it's especially useful in multi-layered drum programs. Hold the **Enter** button and strike a key. The display will show the layer(s) assigned to that key. If more than one layer is assigned to the same key, repeatedly striking the key (while continuing to hold the **Enter** button) will cycle through all layers assigned to that key. This method will work in most places within the Program Editor, but there is an exception: if the parameter you have highlighted has a note number or control source as its value, then holding **Enter** and striking a note will call up that note or control source (as described in *Intuitive Data Entry* on page 3-6). For all other parameters, however, this method will switch between layers.

## The Soft Buttons in the Program Editor

The Program Editor's soft buttons are labeled by the words that appear in the bottom line of the display. These buttons have two important jobs in the Program Editor: selecting pages, and selecting specific functions. If a soft button is labeled in all uppercase letters, pressing it will take you to the page it describes. If the button is labeled in mixed uppercase and lowercase letters, pressing it will execute the software function described by the label. Pressing the **PITCH** soft button, for example, will select the PITCH page, while pressing the **Save** soft button will initiate the process for saving the currently selected program.

There are more pages and functions in the Program Editor than there are soft buttons. Therefore, two of the soft buttons are dedicated to scrolling through the list of pages and functions. If you don't see the button for the page or function you want to select, press one of the soft buttons labeled **<more>**, and the labels will change. This doesn't change the currently selected page, it merely changes the selection of available soft buttons.

Five of the soft buttons in the Program Editor are special cases. They're the soft buttons that select the editing pages for the five control-input pages for the *DSP functions*. One of these soft buttons is always labeled **PITCH**, since the first DSP function in every algorithm is the pitch control. The remaining four vary somewhat depending on the DSP functions you choose for the currently selected algorithm, but they always have the prefixes **F1**, **F2**, **F3**, and **F4**, and they always take you to the pages for the four DSP functions that follow the pitch control function.

## Algorithm Basics

The basic definition: an algorithm is the "wiring" (signal path) of a sample to the audio outputs, through a series of digital signal processing (DSP) functions that you select. The K2661's algorithms are the core of Variable Architecture Synthesis Technology. The DSP functions are synthesis tools (filters, oscillators, etc.) that you assign to the various stages of the algorithm. The DSP functions you choose determine the type of synthesis you use.

Each of the available algorithms represents a preset signal path. You can't change the path of the algorithms, but you can select different algorithms, and assign a wide variety of DSP functions to the individual stages of each algorithm's signal path. Take a look at Algorithm 1 in the diagram below. It's one of the simplest algorithms.

### Algorithm 1



The DSP functions are represented by the rectangular blocks. The horizontal arrows indicate the flow of the digital signal from left to right. They represent what we call the "wire" of the algorithm: the actual physical path that the signal follows through the algorithm. Selecting different algorithms can be compared to connecting different DSP functions with different wiring diagrams.

Think of the left side of each block as its input, and the right side as its output. Depending on the algorithm, the signal may split into two wires, enabling part of the signal to bypass certain portions of the algorithm. Split wires may rejoin within the algorithm, or they may pass all the way through as split signals. If the last block has two wires at its output, we call it a double-output algorithm. If it has one wire, it's a single-output algorithm, even if there are two wires in earlier portions of the algorithm.

The five downward-pointing arrows indicate the five real-time control inputs to the DSP functions. There are usually five inputs, each of which has its own page within the Program Editor. (Algorithms 26-31, which use hard sync oscillation, have only four inputs; you can read about hard sync functions on page 16-55.) Each of these pages has several parameters that can modulate its related DSP function. Often a single DSP function will have more than one input. That's why some blocks are larger, and have more than one arrow pointing to them. Each function can be independently controlled by a variety of sources (the Control Source list), including LFOs, ASRs, envelopes, programmable functions, and external MIDI.

In Algorithm 1, the signal flows first through a one-stage DSP function that controls the pitch of the samples in the keymap. In fact, the first DSP function in each algorithm *always* controls pitch, even though it doesn't apply in every instance. Similarly, the last DSP function always controls the final amplitude of the signal. It can be a one-stage or two-stage function. In Algorithm 1, it's a one-stage function.

The second, larger block indicates a single three-stage DSP function, meaning that it has three control inputs that can be adjusted to modulate the signal.

Once again, you can't change the wiring path of an algorithm—you can simply select a different algorithm to get a different path. And within each algorithm, you can assign a large number of different DSP functions to each of the five control inputs. The diagram below, for example, shows Algorithm 1 with all the possible values for each DSP function lined up under the blocks that represent the DSP functions.

Algorithm 1

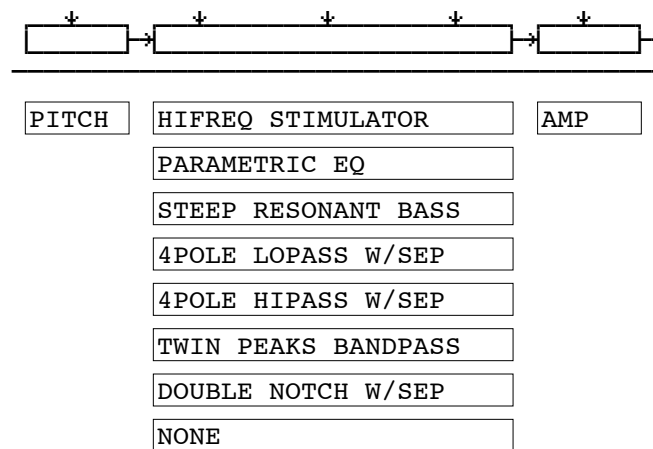


Figure 6-2 Algorithm Wiring

Notice that **PITCH** is the only value available for the first block, and **AMP** is the only value available for the last block. The center, three-stage block, however, allows you to choose from seven DSP functions. An eighth value, **NONE**, deactivates the block.

## Common DSP Control Parameters

The type of DSP function available for any function block depends on the algorithm. Some of the specialized functions like the PANNER are always located just before the final AMP function. Others, like the three-input functions, appear only in algorithms that are structured for three-input functions.

You can change the nature of each layer of a program simply by assigning different DSP functions to the layer's algorithm. Your level of control goes much deeper than that, however. Each DSP function has one or more inputs to which you can patch a variety of control sources to modify the behavior of the DSP functions themselves. These control inputs are represented by the arrows pointing down at the blocks that make up the algorithm. For each input arrow, there's a corresponding control-input page that you can select with the five special soft buttons we mentioned above (**PITCH**, and **F1–F4**). All of the DSP functions have at least one control input, but many of them have two or even three inputs.

The parameters on the various control-input pages are very similar; in fact, there are six parameters that appear on almost every page. Consequently we refer to them as the common DSP control parameters. Although the parameters on the control-input pages differ slightly from function to function, you can expect to see some or all of the common DSP control parameters whenever you select the control-input page for any of the DSP functions.

### Initial Setting Parameters

These have no input, but set the overall level for the function, the starting point from which the other parameters modulate the function.

- Coarse adjust (Coarse)
- Fine adjust (Fine)

### Hard-wired Parameters

These always take their input from MIDI events (either the K2661 or an external MIDI controller)—specifically the note number and the attack velocity values of each Note On event.

- Key tracking (KeyTrk)
- Velocity tracking (VelTrk)

### Programmable Parameters

These can accept any control source as their input, and have related parameters for further control.

- Source 1 (Src1)
- Source 2 (Src2)

Take a look at the **PITCH** page, as an example—we'll look at how these six control parameters are used in the pitch control function. If you're not already on the **PITCH** page, you can get there



by pressing the soft button labeled **PITCH**. If you don't see **PITCH** on the bottom line of the display, press one of the **<more>** buttons until it appears.

```

EditProg:PITCH                                >Layer:1/1
Coarse:0ST                                     Src1 :OFF
Fine :0ct                                     Depth :0ct
FineHz: 0.00Hz                               Src2 :OFF
KeyTrk:0ct/key                               DptCtl:MWheel
VelTrk:0ct                                   MinDpt:0ct
   MaxDpt:0ct
<more>  ALG  LAYER  KEYMAP  PITCH  <more>

```

You'll recognize the common DSP control parameters, along with several other parameters. Keep in mind that there's a set of common control parameters for each of the DSP functions; in this case we're describing them only as they apply to the pitch control function.

## Coarse Adjust

The Adjust parameter (sometimes coarse and fine adjust) is the fixed amount of adjustment you add to any DSP function. On the PITCH page, the Coarse Adjust parameter will change the pitch in semitone increments. Use this as a starting point to set the pitch where you want it to be normally. This will shift the pitch of the currently selected layer, and will affect the playback rate of sampled sounds. Sampled sounds have an upper limit on pitch adjustment. It's normal for the pitches of sampled sounds to "pin" (stop getting higher) when you adjust the pitch upward in large amounts. The oscillator waveforms can be pitched higher. Any sound can be pitched downward without limit.

The primary use of the Adjust parameters (Coarse *and* Fine) is to offset the cumulative effects of the other parameters on the control-input pages. For example, you might set a high value for key tracking (defined below) for a dramatic change in effect across the keyboard. The effect might be too much at one end of the keyboard, however, so you could use one of the Adjust parameters to reduce the initial amount of that effect.

The K2661 always uses real values of measurement, rather than just arbitrary numbers, for adjustable parameters. This means that you specify pitch in semitones and cents, amplitude in dB, and filter cutoff frequency in Hz.

Remember that the parameters on the control-input pages are cumulative—they can add to or subtract from the effects of the other parameters on the page, depending on their values. For example, even if you've adjusted the pitch of a sample so high that it pins, the effects of the other parameters may bring the pitch back down to a workable range.

## Fine Adjust

You can add slight detuning to the pitch with the fine adjust parameter. Notice that there are actually two fine adjust parameters on the PITCH page: one that changes the pitch in cents (100ths of a semitone), and one that changes it according to its frequency (in increments of Hertz—cycles per second). Since we're discussing the universal control sources here, and not specifically pitch, we'll move on for now, as the Fine Hz parameter applies only to pitch-related functions. See *The PITCH Page* on page 6-27 for a more thorough description of Fine Hz.

## Key Tracking

This is a quick way to get additional control based on the MIDI note number of each note you trigger. Key tracking applies a different control signal value for each note number. In the case of pitch, key tracking enables you to change the tuning of each note relative to its normal pitch.

Middle C is the zero point. Regardless of the key tracking value, there is no effect on Middle C. If you set a nonzero value for key tracking, the effect increases for each note above or below Middle C. In the case of pitch, for example, say you assign a value of **5 cents per key** for the key tracking parameter. Triggering Middle C (C 4 on the K2661) will play a normal C 4. Triggering C<sup>#</sup> 4 will play a note 5 cents higher than C<sup>#</sup> 4. Triggering D 4 will play a note 10 cents higher than D 4, and so on. Notes below Middle C will be tuned lower than their normal pitches. If you set a negative value for key tracking, notes above Middle C will be tuned lower than their normal pitches.

Keep in mind that key tracking on the PITCH page works in conjunction with the key tracking parameter on the KEYMAP page. This is why you can set the KeyTrk parameter on the PITCH page to **0ct/key**, and notes still increase in pitch by 100 cents/key as you go up the keyboard. It's because the KeyTrk parameter on the KEYMAP page is already set at **100 cents per key**.

## Velocity Tracking

A positive value for velocity tracking will raise the pitch as you trigger notes with higher attack velocities. This is great for getting a trace of detuning based on your attack velocity, especially in drum programs, where you can make the pitch of the drum samples rise slightly with higher-velocity Note Ons, just as drums do when you strike them harder. Negative values will lower the pitch as you increase the attack velocity.

## Source 1 (Src1)

This parameter takes its value from a long list of control sources (you can find it in the *Musician's Reference*—it's called the Control Source list) including every MIDI control number, a host of LFOs, ASRs, envelopes and other programmable sources.

Src1 works in tandem with the parameter beneath it on the page: Depth. Choose a control source from the list for Src1, then set a value for Depth. When the control source assigned to Src1 is at its maximum, the pitch will be altered to the full depth you set. For example, if you set Src1 to **MWheel**, and set Depth to **1200 ct**, the pitch will rise as you push the Mod Wheel up on your K2661 or MIDI controller, reaching a maximum of 1200 ct (12 semitones, or one octave).

## Source 2 (Src2)

This one's even more programmable. Like Src1, you choose a control source from the list. But instead of setting a fixed depth, you can set a minimum and maximum depth, then assign another control source to determine how much depth you get. Try this example. (Make sure Src1 is set to **OFF** first, so the two sources don't interact.) Start with Program 199, and press **Edit**. Press the **PITCH** soft button to select the PITCH page. Set the Src2 parameter to a value of **LFO1**, then set the Minimum Depth parameter to **100 ct**, and Maximum Depth to **1200 ct**. Then set the Depth Control parameter to **MWheel**. This lets you use the Mod Wheel to vary the depth of the oscillation in pitch generated by the LFO.

Now, when the Mod Wheel is down, the pitch will oscillate between a semitone (100 ct) up and a semitone down (the default waveform for LFO1 is a sine wave, which goes positive and negative—if this perplexes you, see the *Musician's Reference*, where there's an explanation of how the K2661 generates and interprets control source signals). With the Mod Wheel up, the pitch will oscillate between an octave up and an octave down.

Since the Mod Wheel is a continuous control, you can achieve any amount of depth control between the minimum and maximum. If you had set the Depth Control to **Sustain**, for example, then you'd get only two levels of depth control: the maximum (1200 cents) with your MIDI controller's sustain pedal down, or the minimum (100 cents) with the sustain pedal up.

## Summary of Common DSP Control Parameters

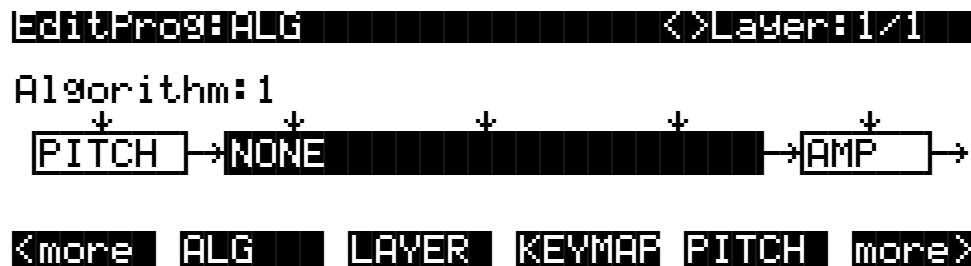
These six control source parameters are just a few of the control sources available throughout the Program Editor. We've given them special attention because they appear on *all* the pages relating to the DSP functions, not just on the PITCH page.

As with the PITCH parameters, you can go to each of the DSP functions' control-input pages, and set a similar set of parameters to control each of those functions as well. The units of measurement may differ, but you'll almost always find one or two adjustment parameters, key and velocity tracking, and two programmable control sources. And remember, we've been talking about one layer in one program here. You can add one or two more layers to your program, and start all over with another identical set of control sources for each layer, each of which can be programmed independently.

On any given page, the settings for the control parameters are added to each other before the signal leaves the DSP function. Depending on the values you set, they may cancel each other out, or they may add up to huge amounts of modulation. If things get out of control, the easiest way to get a handle on the situation is to set some of the parameters to values of **0** or **OFF**. Adjust the value for one parameter at a time to hear the effect of that one parameter.

## The Algorithm (ALG) Page

The ALG page is the first page you see when you enter the Program Editor. It enables you to select from among the possible algorithms, and assign the DSP functions within the current algorithm.



The top line of the display gives you the usual mode reminder, and tells you which layer you're looking at, as well as how many layers are in the current program (in the diagram above, it's the first layer of a one-layer program). You can view the ALG pages of any other layers in the program by using the **Chan/Bank** buttons.

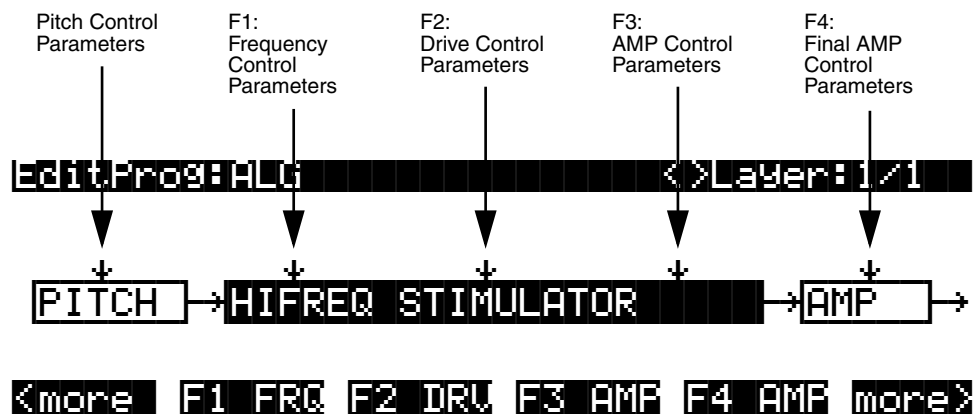
The central portion of the page shows the algorithm for the currently selected layer. You see the number of the algorithm (from 1 to 31) and a graphic representation of the signal path, as well as the currently selected DSP functions within the signal path.

To use a different algorithm, select the Algorithm parameter and use any data entry method to select a different one. To change the DSP function within an algorithm, move the cursor to the block you want to change, then use the Alpha Wheel or **Plus/Minus** buttons. There's a staggering number of combinations of algorithms and DSP functions alone, not to mention the numerous controls that can be used to modify the DSP functions. The *Musician's Reference* contains a list of all 31 algorithms and the DSP functions available for each one.



**Note:** Changing a layer's algorithm can affect the layer's sound drastically. It's a good idea to bring down the volume of your K2661 or your sound system before changing algorithms.

The five downward-pointing arrows represent inputs to the DSP functions that are available for the current algorithm. Each input arrow has its corresponding page. The first arrow points to the PITCH function. The soft button for the PITCH page is already visible. Press it to view the parameters affecting pitch for the currently selected layer. The buttons for the other four DSP functions are not visible when you first enter the Program Editor. To see them, press the **more>** soft button (on the right side of the page). You're still on the same page, but the soft buttons' labels change to let you select a different set of pages, as the diagram below shows.



**Figure 6-3 DSP Function Inputs**

The pages (F1–F4) that control the DSP functions are described later in this chapter. *Algorithm Basics* on page 6-12 gives general information on algorithms, while Chapter 16 gives a thorough description of each the DSP functions and the parameters found on their editing pages.

## The LAYER Page

Press the **LAYER** soft button to call up the LAYER page. Here you'll set a number of parameters that affect the current layer's keyboard range, attack and release characteristics, and response to various controls.

```

EditProg:LAYER          <>Layer:1/1
LoKey :C 0    DlyCtl:OFF    SusPdl:On
HiKey :C 8    MinDly:0.000s  SosPdl:On
LoVel :ppp    MaxDly:0.000s  FrzPdl:On
HiVel :fff    Enable:ON     IgnRel:Off
PBMode:All    S:Norm 64    127 ThrAtt:Off
Trig  :Norm   Opaque:Off    TilDec:Off
<more  ALG    LAYER  KEYMAP  PITCH  more>

```

| Parameter           | Range of Values     | Default |
|---------------------|---------------------|---------|
| Low Key             | C -1 to G 9         | C 0     |
| High Key            | C -1 to G 9         | C 8     |
| Low Velocity        | ppp to fff          | ppp     |
| High Velocity       | ppp to fff          | fff     |
| Pitch Bend Mode     | Off, Key, All       | All     |
| Trig                | Normal, Reversed    | Normal  |
| Delay Control       | Control Source list | Off     |
| Minimum Delay       | 0 to 25 seconds     | 0       |
| Maximum Delay       | 0 to 25 seconds     | 0       |
| Layer Enable        | Control Source list | On      |
| Enable Sense        | Normal, Reversed    | Normal  |
| Enable Min          | -128 to 127         | 64      |
| Enable Max          | -128 to 127         | 127     |
| Opaque Layer        | Off, On             | Off     |
| Sustain Pedal       | Off, On, On2        | On      |
| Sostenuto Pedal     | Off, On             | On      |
| Freeze Pedal        | Off, On             | On      |
| Ignore Release      | Off, On             | Off     |
| Hold Through Attack | Off, On             | Off     |
| Hold Until Sustain  | Off, On             | Off     |

### Low Key (LoKey)

This sets the lowest active note for the current layer. This parameter's value cannot be set higher than the value for HiKey. The standard MIDI key range is C -1—G 9 (0-127). Middle C is C 4 (ISP).

### High Key (HiKey)

Here you set the highest active note for the current layer. This parameter's value cannot be set lower than the value for LoKey.

## Low Velocity (LoVel)

With this parameter you define the lowest attack velocity at which the layer will be enabled (generate a sound). The values for this parameter and the next are expressed in the standard musical dynamics markings, similar to the values available for the velocity maps. Attack velocities that are below this threshold will not trigger notes. If you set this parameter's value higher than the HiVel value, the layer will not play at all.

## High Velocity (HiVel)

Similarly, this will set the highest attack velocity at which the layer will be enabled. Attack velocities above this threshold will not trigger notes in this layer.

Using LoVel and HiVel, you can set up velocity switching between up to eight layers. If you need even more, you can do it using the Enable and Enable Sense (S) parameters (page 6-21).

## Pitch Bend Mode (PBMode)

This determines how Pitch bend control messages will affect the current layer. A value of **All** bends all notes that are on when the Pitch bend message is generated. A value of **Key** bends only those notes whose triggers are *physically* on when the Pitch bend message is generated (notes held with the sustain pedal, for example, won't bend). This is great for playing guitar solos on top of chords—play a chord, hold it with the Sustain pedal, then play your licks and bend them all you want; the chord won't bend with it. A value of **Off** disables Pitch bend for the current layer.

## Trigger (Trig)

Set Trig to **Rvrs** to have notes triggered on key-up. The initial velocities of notes triggered this way are determined by the release velocities of the keys that trigger them. The default setting is **Norm**.

## Delay Control (DlyCtl)

Here you select, from the Control Source list, a control source that will delay the start of all notes in the current layer. The length of the delay is determined by MinDly and MaxDly (described below). You'll assign a continuous control like MWheel for the DlyCtl parameter when you want to vary the delay time, and a switch control if you want the delay to either be its minimum value (switch off), or its maximum (switch on). The delay control will affect only those notes triggered *after* the delay control source is moved; the delay time is calculated at each note start, based on the status of the delay control source at that time.

## Minimum Delay (MinDly), Maximum Delay (MaxDly)

The length of the delay is determined by these two parameters. When the control source assigned to DlyCtl is at its minimum, the delay will be equal to the value of MinDly. The delay will be equal to the value of MaxDly when the control source is at its maximum. If DlyCtl is set to **OFF**, you get the minimum delay. If it's set to **ON**, you get the maximum delay. This doesn't change the note's attack time, just the time interval between the Note On message and the *start* of the attack. The delay is measured in seconds.

## Enable

This assigns a control source to activate or deactivate the layer. When the value of the assigned control source is between the minimum and maximum thresholds set by the Sense (S) parameter, the layer is active. When the value of the assigned control source is below the minimum or above the maximum, the layer is inactive. By default, many layers have the Enable parameter set to **ON**, so the minimum and maximum thresholds don't matter. They're relevant only when Enable is set to a specific control source (like **MWheel**).

Some local control sources (**KeyNum** and **AttVel**, for example) are not valid for the Enable parameter. In these cases, you should use the global equivalent (**GKeyNum** and **GAttVel** in this example).

## Enable Sense (S)

This parameter determines how and when a layer is enabled by the control source assigned for the Enable parameter. Enable Sense has three values: orientation, minimum, and maximum.

Suppose for a moment that you're editing a program, and in the current layer you've set the value of Enable to **MWheel**, which causes the Mod Wheel to control whether the layer is active. The default values for Enable Sense are as follows: orientation is **Norm**; minimum is **64**, and maximum is **127**. This means that when the Mod Wheel is less than halfway up, the layer is disabled. The layer plays only when the Mod Wheel is more than halfway up.

Change the orientation to **Rvrs**, and the layer plays only when the Mod Wheel is *less* than halfway up. Change the orientation back to **Norm**, and change the minimum to **127**. Now the layer plays only when the Mod Wheel is *all* the way up.

You could use this parameter to set up a two-layer program that would let you use a MIDI control to switch between layers, say a guitar sound and a distorted guitar. Both layers would have their Enable parameters set to the same control source, say **MWheel**. One layer would have its Enable Sense orientation set to **Norm**, and the other would have it set to **Rvrs**. Both layers would have their Enable Sense minimums set to 64, and their maximums to 127. The first layer would play when your Mod Wheel was above its midpoint, and the second layer would play when the Mod Wheel was below its midpoint. (You could achieve the same effect by having the Enable Sense orientation in both layers set to **Norm**, and the min and max values set as follows: min **0** and max **63** for one layer; min **64** and max **127** for the other.)

Using this parameter in conjunction with the Enable parameter, you can easily create velocity-switching for as many layers as you have in your program. This is useful for drum programs, since you can define a different velocity-trigger level for each of the 32 layers available in drum programs.

First, set the Enable parameter for the Layer 1 to a value of **GAttVel** (global attack velocity). This causes the layer to play based on the attack velocity of your keystrokes. Then set the Enable Sense (S) parameter to a value of **Norm**, and adjust its minimum and maximum values (the two numerals to the right of **Norm**) to a narrow range. Don't use negative values, since they don't apply when you're using **GAttVel** as the layer enabler.

Repeat this for each layer in the program. Bear in mind that if you want to set up 32 different velocity levels for a program, with equal intervals between each layer, then you have a range of 4 for each level (Layer 1 is 0–3, Layer 2 is 4–7, and so on). It won't be easy to play precisely enough to trigger the layer you want. On the other hand, if you're using Song mode or an external sequencer, you can edit attack velocity levels, and get exactly the results you want.

## Opaque

An opaque layer blocks all higher-numbered layers in its range, allowing only the opaque layer to play. This is an easy way to change a small range of notes in a program, leaving the original sound playing above and below the new sound.

Start with a one-layer program, and create a new layer (Layer 2) with the **NewLyr** soft button. On the KEYMAP page for Layer 2, select the keymap you want to use, then on the LAYER page, set Layer 2's range (say, C 3 to D 3), and set its Opaque parameter to **On**. Then go to Layer 1, and duplicate it (with the **DupLyr** soft button); the duplicate layer becomes Layer 3. You now have a three-layer program. Delete Layer 1 (the original layer); Layer 2 (the new layer you created) becomes Layer 1, and Layer 3 becomes Layer 2. Now Layer 2 blocks out Layer 3 (the duplicate of the original layer) at the notes C 3–D 3.

## Sustain Pedal (SusPdl)

When this parameter is on, the layer will respond to all sustain messages (MIDI 64). When off, the current layer will ignore sustain messages. **On2** means that the sustain pedal will not catch the release of a note that is still sounding when the sustain message is received; this can be very useful in a program that uses amplitude envelopes with a long release time.

## Sostenuto Pedal (SosPdl)

When Sostenuto is on, the layer will respond to all sostenuto messages (MIDI 66). When off, the layer ignores sostenuto messages. Sostenuto, as you may know, is a feature found on pianos that have three pedals. Pressing the Sostenuto pedal on a piano (usually the middle pedal) sustains the notes whose keys you were holding down when you pressed the pedal. Notes played after the pedal is already down do not get sustained.

## Freeze Pedal (FrzPdl)

This parameter activates or deactivates the layer's response to Freeze pedal messages (MIDI 69). The Freeze pedal control causes all notes that are on to sustain without decay until the Freeze pedal control goes off. If a note is already decaying, it will freeze at that level.

## Ignore Release (IgnRel)

When on, the layer will ignore all Note Off messages received by the K2661. This should be used only with sounds that decay naturally, otherwise the sounds will sustain forever. When IgnRel is off, the layer responds normally to Note Off messages. This parameter can come in handy when your K2661 is slaved to a drum machine or sequencer, which sometimes generates Note Ons and Note Offs so close together that the envelope doesn't have time to play before the note is released. You'll also want to use this parameter when you're playing staccato, and the sound you're playing has a long amplitude envelope. This parameter should be used only with notes that eventually decay to silence. Sustaining sounds will sustain forever.

## Hold Through Attack (ThrAtt)

When on, this parameter causes all notes in the layer to sustain through the entire first attack segment of their amplitude envelopes, even if the notes have been released. If you have a sound with a slow attack, or an attack that's delayed with the delay control, setting this parameter to **On** will make sure your notes reach full amplitude even if you're playing fast. When set to **Off**, notes will release as soon as you release the note (generate a Note Off). If the first attack segment of the layer's amplitude envelope is very short, you probably won't notice a difference between values of **On** and **Off**.



## Hold Until Decay (TilDec)

When on, this parameter causes all notes in the layer to sustain through all three attack segments in their amplitude envelopes even if the notes have been released. Looped amplitude envelopes will not loop, however, if the notes are released before reaching the end of the final attack segment. Notes will go into their normal releases if they are released after the envelope has looped. When set to **Off**, notes will release as soon as a Note Off message is generated.

## The KEYMAP Page

Press the **KEYMAP** soft button to call up the KEYMAP page. The parameters on this page affect sample root selection—which samples are played on which keys.

```

EditProg:KEYMAP <>Layer:1/1

KeyMap:1 Grand Piano Stereo:Off
Xpose :0ST TimbreShift :0ST
KeyTrk:100ct/key PlayBackMode:Normal
VelTrk:0ct AltControl :OFF
SmpSkp:Auto AltMethod :Switched
<more> ALG LAYER KEYMAP PITCH >more>
  
```

| Parameter         | Range of Values                            | Default       |
|-------------------|--------------------------------------------|---------------|
| Keymap            | Keymap List                                | 1 Grand Piano |
| Transpose         | ± 60 Semitones                             | 0             |
| Key Tracking      | ± 2400 Cents per key                       | 100           |
| Velocity Tracking | ± 7200 Cents                               | 0             |
| Sample Skipping   | Auto, Off, On                              | Auto          |
| Stereo            | Off, On                                    | Off           |
| Timbre Shift      | ± 60 Semitones                             | 0             |
| Playback Mode     | Norm, Rvrs, Bidirectional, Noise, ADAT 1—8 | Normal        |
| Alt Control       | Control Source List                        | Off           |
| Alt Method        | Switched, Continuous                       | Switched      |

## Keymap

Assign a ROM or RAM keymap to the current layer. Keymaps are collections of samples assigned to note and velocity ranges. There are nearly 200 ROM keymaps to choose from. You'll find a list of them in the *Musician's Reference*.

## Transpose (Xpose)

Transpose the current keymap up or down as much as 60 semitones (5 octaves).

## Key Tracking (KeyTrk)

This is one of the six common DSP control parameters. On the KEYMAP page, key tracking affects the interval between notes. The default value of **100 cents** (a cent is a hundredth of a semitone) gives you the normal semitone interval between each note. Higher values increase the interval; lower values decrease it. Negative values will cause the pitch to decrease as you play higher notes. (You can create a mirror-image piano by setting the key tracking to **-100** and transposing the layer up 4 semitones.)

When you make changes to this parameter, you'll need to keep in mind that KeyTrk on the KEYMAP page works in conjunction with KeyTrk on the PITCH page. Therefore, you'll need to check the KeyTrk value on both pages to see how key tracking works within a program. Unless you're looking for nonstandard note intervals, the values of the KeyTrk parameters on the PITCH and KEYMAP pages should add up to 100 cents.

## Velocity Tracking (VelTrk)

This is another common DSP control parameter. As with the other parameters on the KEYMAP page, this shifts the position of the keymap. Different attack velocities will play different pitch shifts of the sample root assigned to that note range. If the shift is great enough, the next higher or lower sample root will be played, which in some cases (many drum programs, for example) will play an entirely different sound. Positive values will play higher pitches of the sample root when you use hard attack velocities (they shift the keymap downward), while negative values will play lower pitches.

## Sample Skipping (SmpSkp)

Sample skipping allows for increased upward transposition of samples. This is done by using a special sample playback algorithm, which enables the K2661 to increase the maximum playback rate of a sample from 96 Khz to a maximum of 192 Khz (thereby enabling the sample to be played at a higher pitch).

There is a tradeoff, however. Unwanted artifacts may creep into a note's sound as the result of sample skipping. Therefore, the **Auto** value for this parameter is usually the best choice. **Auto** means that the keymap in this layer will employ sample skipping only for those notes whose upward transposition can be increased. Notes below a certain cutoff point would not benefit from sample skipping and, therefore, **Auto** ensures that these notes will not use the feature unnecessarily. The only disadvantage to using Auto sample skipping is that you cannot pitchbend a note from below the cutoff point into the range of the sample-skipped notes.

A value of **On** means that sample skipping will be employed throughout the range of the keymap. This eliminates the pitchbend limitation described above, but may add some artifacts to the sound. Creative types may appreciate this form of distortion, however, so we've made it available.

A value of **Off** means that sample skipping will not be used at all.

## Stereo

You'll use this parameter when you're working with stereo samples. When you use the stereo piano programs or load stereo samples from disk, the K2661 views both sides of the sample as a single sample object. When you select a stereo sample (by setting the value of the Sample parameter in the Keymap Editor), you'll see the letter **S** as part of the sample name (for example, **204\*StratoBlaster E3 S**).

When you set this parameter to **On**, the KEYMAP page changes slightly:

```

EditProg:KEYMAP          <>Layer:1/1
KeyMap1:1 Grand Piano
KeyMap2:None             Stereo:On
Xpose :0ST              TimbreShift :0ST
KeyTrk:100ct/key        AltAttackCtl:OFF
VelTrk:0ct              PlayBackMode:Normal
SmPskp:Auto
<more  ALG  LAYER  KEYMAP  PITCH  more>

```

An additional Keymap parameter appears. The two keymap parameters are distinguished as Keymap 1 and Keymap 2. The KEYMAP page parameters will affect both keymaps. When the Stereo parameter is set to **On**, the OUTPUT page for the current layer will show an additional pair of Pan parameters.

To get the samples to play together, set the Stereo parameter to **On**, and select the same keymap as the value for *both* the Keymap1 and Keymap2 parameters. (With some imported sample formats, such as Akai, you'll have two keymaps for a stereo sample—for example, Piano L and Piano R. In this case, select the left keymap as the value for Keymap1, and the right keymap as the value for Keymap2.) If you select a keymap as the value for both Keymap1 and Keymap2, the K2661 automatically uses the left side for Keymap1, and the right side for Keymap2.

Once you have the keymaps assigned, go to the OUTPUT page and set the panning for each sample as desired. Keep in mind that using stereo keymaps reduces the polyphony of the program. For example, if you had a two-layer program with stereo keymaps in each layer, each note you play would use 4 of your 48 voices, allowing a total of 12 notes before all the voices have been used.

If you're not using stereo samples, you should set this parameter's value to **Off**.

## Timbre Shift

This parameter works only on multi-sample keymaps, and changes the root selection for each key you play. With this parameter you can radically alter the current layer's timbre (basic sound characteristics). The nature of the change depends on the timbre itself, so this parameter calls for experimentation. Basically, timbre shifting changes a note's timbre by imposing different harmonic qualities onto the note. A timbre-shifted note retains its original pitch, but its harmonics are those of the same timbre at a higher or lower pitch. Positive values for this parameter tend to brighten a sound, while negative values darken.

Here's an example. If you shift the timbre up 4 semitones, then playing C 4 will result in the *pitch* C 4, but will actually play the sample normally assigned to G<sup>#</sup> 3, and shift its pitch up four semitones. This will increase the playback rate of the sample, so although the pitch remains normal, the timbre is brighter. You'd get the same effect by setting the Xpose parameter on the KEYMAP page to **-4 semitones**, then setting the Adjust on the PITCH page to **+4 semitones**. For multi-sample layers with narrow key ranges, large amounts of timbre shifting will cause different sample roots to be played back.

## Playback Mode

This gives you numerous options for manipulating the samples in the current layer as you trigger them. **Normal** leaves the samples unaffected, while **Reverse** plays them in reverse. At a value of **Reverse**, the samples will continue to loop as long as notes are sustained. To play them just once in reverse, you would adjust the length of the layer's amplitude envelope (explained later in this chapter). **BiDirect** (bidirectional) causes the samples to loop infinitely, alternating between normal and reversed playback. **Noise** replaces the samples with a white noise generator.

**ADAT In 1** through **ADAT In 8** let you route an external digital signal to the layer's sample. This signal then gets processed by the layers' DSP algorithm and sent to the layer's output. Using this value disables most of the other KEYMAP-page parameters for the layer.



***Note:** To use ADAT In, the K2661's ADAT Out cable must be connected to the sending device. K2661 must be the "master," and the other device(s) must "slave" to it. Output sample rate (and therefore input as well) is fixed at 48 kHz.*

Using the ADAT inputs bypasses the PITCH page in the current program, so nothing on the PITCH page has an effect. If the current program uses a natural amplitude envelope, the signal at the ADAT In uses the current keymap's amplitude envelope and volume adjust parameters.

## Alternative Switch (AltControl and AltMethod)

You can assign a control source to change the sound by using an alternative start point or alternative end point for the current keymap. Whether it is an alternative start or alternative end depends on the position of the Alt parameter for the sample (set in the Sample Editor). When set before the end point, it is used as an alternative start (the Alt point can be before or after the normal Start point). When set after the end it is used as an alt end.

Use the AltControl parameter to specify a control source that will cause the sample to begin or end at the Alt point. Then use the AltMethod parameter to choose between switched and continuous calculation of the Alt point. If the value of AltMethod is **Switched**, the K2661 will use the Alt point when the relevant control source is at a value greater than 64 at Note Start. If AltMethod is **Continuous**, the Alt point will vary depending on the value of the relevant control source at Note Start.

As an example, suppose you're editing a two-second sample. You've set the Start point at **0.000**, and the Alt point at **1.000** (this is done on the TRIM page in the Sample Editor). Now you return to the KEYMAP page in the Program Editor, and you set AltControl to **MWheel**. If you set AltMethod to **Switched**, the sample will begin at the 1-second point if the Mod Wheel is at least halfway up at Note Start. If you set AltMethod to **Continuous**, the K2661 will interpolate the sample's starting point based on the position of the Mod Wheel. If the Mod Wheel is halfway up (64) at Note Start, the sample will begin at the half-second point. If the Mod Wheel is 75% up (96), the sample will begin at the .75-second point, and so on.

### Emulating Legato Play

If you place the Alt point after the initial attack transients of the sample, then you can use the Alt Switch to emulate legato playing in an acoustic instrument. As an example, set Keymap to **14 Flute**. Now set the AltControl parameter to **Chan St** (Channel State). Now if you play notes separately, the initial breathy chuff will be heard. But if you play the notes legato (connecting them smoothly), the Alt point is used and you do not hear the chuff. This is because the Chan St is turned on as long as any note is being held. Most of the K2661's ROM samples have their Alt points set for purposes of legato play. In most cases the difference in attacks is subtle, but for some sounds, like drums, the difference can be more noticeable.

For more information, refer to the discussion of the TRIM page's Alt point on page 14-16.

## The PITCH Page

Press the **PITCH** soft button, and the PITCH page will appear. These parameters adjust the pitch (playback rate) of the samples after the root has been selected by the keymap.

```

EditProg:PITCH <>Layer:1/1
Coarse:0ST Src1 :OFF
Fine :0ct Depth :0ct
FineHz: 0.00Hz Src2 :OFF
KeyTrk:0ct/key DptCtl:MWheel
VelTrk:0ct MinDpt:0ct
MaxDpt:0ct
<more ALG LAYER KEYMAP PITCH more>

```

| Parameter         | Range of Values           | Default |
|-------------------|---------------------------|---------|
| Coarse Adjust     | –120 to 60 ST (Semitones) | 0       |
| Fine Adjust (Cts) | ± 100 Cents               | 0       |
| Fine Adjust (Hz)  | ± 6 Hz                    | 0       |
| Key Tracking      | ± 2400 Cents per key      | 0       |
| Velocity Tracking | ± 7200 Cents              | 0       |
| Source 1          | Control Source List       | Off     |
| Depth             | ± 7200 Cents              | 0       |
| Source 2          | Control Source List       | Off     |
| Depth Control     | Control Source List       | Mwheel  |
| Minimum Depth     | ± 7200 Cents              | 0       |
| Maximum Depth     | ± 7200 Cents              | 0       |

These parameters were described in the section on common DSP control parameters earlier in this chapter, so we won't repeat them here, but a word about the Fine Hz parameter is in order.

### Fine Hz

This measures pitch adjustment by the relative frequency (in Hertz) of each note. This is useful for controlling the beat frequency between layers in a multi-layered program. Using this parameter to detune chorusing layers will keep the beat frequency constant across much of the keyboard. Although the ratio of frequencies between each layer remains constant, the detuning will increase at lower pitches, and can become extreme. The K2661 automatically limits the amount of detuning when it becomes extreme, so you'll notice the beat frequencies moving out of sync when you play low pitches.

## The F1–F3 Pages

These pages are reached by pressing the **F1**, **F2**, and **F3** soft buttons, respectively. They contain the parameters governing the three variable DSP functions in each algorithm. The pages vary depending on the DSP functions selected for the three middle DSP control inputs, represented

by the downward-pointing arrows on the ALG page. See the *Musician's Reference* for a complete list of the algorithms and their available DSP functions.

## The F4 AMP Page

Press the **F4 AMP** soft button to call up this page, which features five of the six common DSP control parameters, in this case controlling the final amplitude of the current layer before it reaches the audio outputs. There's also a parameter that enables you to pad (attenuate) the current layer's signal before its final amplification.

```

EditProg:F4 AMP(FINAL AMP) <>Layer:1/1
Adjust:6dB                               Src1 :OFF
   Depth :0dB
   Src2 :OFF
KeyTrk: 0.00dB/key                      DptCtl:Mwheel
VelTrk:20dB                             MinDpt:0dB
Pad :0dB                                MaxDpt:0dB
<more F1 F2 F3 F4 AMP more>

```

| Parameter         | Range of Values     | Default |
|-------------------|---------------------|---------|
| Adjust            | –96 to 48 Db        | 6       |
| Key Tracking      | ± 2 Db per key      | 0       |
| Velocity Tracking | ± 96 Db             | 20      |
| Pad               | 0, 6, 12, 18 Db     | 0       |
| Source 1          | Control Source List | Off     |
| Depth             | ± 96 Db             | 0       |
| Source 2          | Control Source List | Off     |
| Depth Control     | Control Source List | Mwheel  |
| Minimum Depth     | ± 96 Db             | 0       |
| Maximum Depth     | ± 96 Db             | 0       |

### Adjust

Adjust the overall amplitude (gain) of the currently selected layer. In multi-layer programs, this parameter adjusts the amplitude of the layers relative to each other. This is the final output control for the layer (post-amp pad). Be careful not to set this too high! If one of your layers is too loud, it's generally better to cut its level than to boost the others. This will keep distortion to a minimum.

### Key Tracking

This uses the MIDI note numbers of the notes you play as a control source affecting the individual amplitudes of each note in the current layer. Positive values increase the amplitude as you play higher-pitched notes. For example, if the key tracking is **.20 dB/key**, then C<sup>#</sup>4 will be .20 dB louder than C 4 if triggered with the same attack velocity. If the value for this parameter were negative, C<sup>#</sup>4 would have less amplitude than C 4. A word of caution here: values above **0.30 dB/key** (or below **–0.30**) can generate extremely high amplitude levels. If you set this parameter that high without lowering the value of the Adjust parameter to **–12 dB** or lower, your sound may clip, which can be useful, but it isn't necessarily what you want.

## Velocity Tracking

This uses the MIDI attack velocity value of the notes you play as a control source affecting the individual amplitudes of each note in the current layer. This is the primary parameter to use for adjusting the dynamics of a layer. At a value of **0**, every note in the current layer would have the same amplitude, regardless of its attack velocity. When the value is positive, note amplitude increases as attack velocity increases. When the value is negative, note amplitude *decreases* as attack velocity increases. Larger values increase the range between minimum and maximum amplitude, so with a large positive value, the amplitude will be low when you play softly. Small values decrease the range between min and max, so with a small positive value, you'll get nearly full amplitude even with light attack velocities.

## Pad

Select one of four attenuation levels for cutting the amplitude of the current layer *before* the final amp stage (preamp pad). Use the pad if the layer's sound distorts when played. Note: clipping can occur in earlier algorithm blocks as well. If this is the case, you'll probably want to try to remove the clipping in the earlier block, if possible.

## Source 1 and Source 2, Depth Controls

These are common DSP control parameters, which in this case let you assign control sources to affect the amplitude of the current layer. The functions of common DSP control parameters are explained in their own section earlier in this chapter.

## The OUTPUT Page

Press the **OUTPUT** soft button to get to the OUTPUT page, where you set the layer's output routing. This doesn't assign the layer's audio signal to the *audio* outputs, as you might expect. It assigns the signal to a KDFX *input*, which determines what effects, if any, get applied to the signal before the actual audio output. The routing of the signal from KDFX to the audio outputs happens on the OUTPUT page in the Studio Editor. See page 9-25, as well as Chapter 19, for more information about audio output.

There are actually four different configurations of the OUTPUT page. The one you see depends on whether the current layer uses a stereo keymap, and whether it uses a double-output algorithm. A double-output algorithm is one whose signal path is split into two parts before final amplification.

Regardless of the page's configuration, there are parameters for adjusting the Output Group, the Pan position, the Output Mode, the Gain, the Crossfade control, and the Crossfade sense. Layers that use stereo keymaps, or that use double-output algorithms, have additional sets of Output Group and Pan parameters on their OUTPUT pages.

The following page is for a layer with one keymap and a single-path algorithm.

```

EditProg:OUTPUT <>Layer:1/1
Pair:KDFX-E
Pan:L * R
Mode:+MIDI
Gain:12dB
Crossfade:OFF XFadeSense:Norm
<more OUTPUT COMMON SetRng more>

```

| Parameter         | Range of Values                              | Default |
|-------------------|----------------------------------------------|---------|
| Pair              | KDFX-A, KDFX-B, KDFX-C, KDFX-D               | KDFX-A  |
| Pan               | Left to Right (15 Positions, indicated by *) | Center  |
| Mode              | Fixed, +MIDI, Auto, Reverse                  | +MIDI   |
| Gain              | –12 to 30 Db (6 Db Increments)               | 18 Db   |
| Crossfade Control | Control Source List                          | OFF     |
| Crossfade Sense   | Normal, Reversed                             | Norm    |

## Pair

This parameter sets the output routing of the current layer—not the final audio output, but the KDFX (effects processor) input to which the layer’s signal will flow. Setting a value here is like connecting an audio output to the effects send on a mixer. The actual audio output of the K2661 is determined by the output settings for KDFX.

In a nutshell, the audio signal gets routed from the K2661’s sound engine into KDFX using this parameter. A separate KDFX parameter routes the signal from KDFX to the physical audio outputs on the K2661’s rear panel.

## Pan

Use this parameter to position the current layer’s audio signal from left to right. An additional pan parameter (Pan2) appears if you have the Stereo parameter on the KEYMAP page set to a value of **On**.

## Mode

When the mode is **Fixed** the pan position remains as defined with the Pan parameter, ignoring MIDI pan messages. When the mode is **+MIDI**, MIDI pan messages (MIDI 10) will shift the sound to the left or right of the Pan parameter setting. Message values below 64 shift it left, while those above 64 shift it right. A setting of **Auto** assigns the pan setting of each note based on its MIDI note number. In this case, Middle C (MIDI note number 60) is equivalent to the Pan parameter’s setting. Lower notes shift increasingly left, while higher notes shift increasingly right. A setting of **Reverse** shifts low notes right, and high notes left. MIDI pan messages will also affect the pan position when values of Auto and Reverse are selected.



***Note:** If you’re using the PANNER DSP function in the algorithm for any of the layers in a program, that layer will respond to MIDI pan messages even if the Mode parameter is set to a value of **Fixed**. In this case, leave Mode set to **Fixed**, and set Pan1 fully left, and Pan2 fully right.*



## Gain

Boost (or cut) the amplitude of the current layer. For layers using double-output algorithms, the gain is divided evenly between the two signal paths. Since this gain is not affected by the layer's amplitude envelope, you can use it to add a constant amount of gain to a layer. This is a good way to make a layer louder without the voices on that layer clipping. If you do this, however, keep in mind that increasing the gain reduces your headroom, so if you're playing a lot of voices, you may get some channel clipping (as opposed to voice clipping).

## Crossfade, Crossfade Sense (XFadeSense)

The Crossfade parameter lets you select a control source to fade the current layer's amplitude from zero to maximum. When Crossfade Sense is **Normal**, the layer is at full amplitude when the Crossfade control is at minimum. With Crossfade Sense set to **Reverse**, the layer is at zero amplitude when the Crossfade control is at minimum.

This parameter is similar to the Src1 and Depth parameters on the F4 AMP page, but the attenuation curve for the Crossfade parameter is optimized specifically for crossfades.

To crossfade two layers in the same program, assign the same control source for the CrossFade parameters in both layers, then set one of their XFadeSense parameters to a value of **Norm**, and the other's to **Rvrs**.

## Other OUTPUT Page Configurations

The following page is for a layer with one keymap and a double-output algorithm. The U and L stand for the upper and lower wires (signal paths). You have independent control of the output parameters for each wire.

```

EditProg:OUTPUT <>Layer:1/1
Pair:  Pan:  Mode:  Gain:
U:KDFX-E L  *  R Fixed 12dB
L:KDFX-A L  *  R +MIDI 12dB

CrossFade :OFF  XFadeSense:Norm
<more OUTPUT COMMON SetRns more>

```

Next are the two page configurations for layers with stereo keymaps: the first one uses a single-output algorithm, and the second uses a double-output algorithm.

With a single-output algorithm, stereo keymap layers let you adjust the pan position of each keymap, but all other parameters are identical for both keymaps.

```

EditProg:OUTPUT <>Layer:1/1
Pair:KDFX-E
Pan1:L  *  R
Pan2:L  *  R
Mode:+MIDI
Gain:12dB
CrossFade :OFF  XFadeSense:Norm
<more OUTPUT COMMON SetRns more>

```

The COMMON Page

When a stereo keymap layer uses a double-output algorithm, both keymaps are split between the upper and lower wires. In other words, both wires carry the signal from each of the keymaps. The Output Group (Pair), Output mode, and Gain level of Keymap 1 are mimicked by Keymap 2 (that's why these parameters aren't displayed for Keymap 2 on the OUTPUT page). You can, however, set the pan positions independently for the upper and lower wires of both keymaps.

```
EditProg:OUTPUT <>Layer:1/1
Pair: Pan: Mode: Gain:
U1:KDFX-E L * R Fixed 12dB
L1:KDFX-A L * R +MIDI 12dB
U2: L * R
L2: L * R
CrossFade :OEF XFadeSense:Norm
<more OUTPUT COMMON SetRng more>
```

The COMMON Page

Here's where you find eight frequently-used parameters that affect the entire current program, not just the current layer. The COMMON page is reached by pressing the **COMMON** soft button in the Program Editor.

```
EditProg:COMMON All Layers
Pitch Bend Range:200Oct Globals:Off
Monophonic :Off OutPair:KDFX-A

<more OUTPUT COMMON SetRng more>
```

Notice that when the Monophonic parameter is set to its default value of **Off**, the four monophonic parameters do not appear on the page.

Parameter	Range of Values	Default
Pitch Bend Range	± 7200 Cents	200 Cents
Monophonic	Off, On	Off
(Legato Play)	Off, On	Off
(Portamento)	Off, On	Off
(Portamento Rate)	1 To 3000 keys per second	70
(Attack Portamento)	Off, On	On
Globals	Off, On	Off
OutPair	KDFX-A, KDFX-B, KDFX-C, KDFX-D	KDFX-A

## Pitch Bend Range

Use this parameter to define how much the pitch will change when you move your Pitch Wheel. Positive values will cause the pitch to bend up when the Pitch Wheel is pushed up, while negative values will cause the inverse. Large positive values can cause samples to bend to their maximum upward pitch shift before the Pitch Wheel is fully up. This will not happen when bending pitch down.

Setups include a parameter called Bend Range (on the BEND page in the Setup Editor), which in most setups does not affect the program-level Pitch Bend Range setting. This is because in most setups the default value for Bend Range is **Prog**, which gives control of the pitch bend range to each program in the setup. If you change the value of Bend Range to anything but Prog, however, the setup will control the pitch bend range for all programs in the setup.

Setup 97 (the default control setup for defining the controller setting while you're in Program mode) has its Bend Range parameter set to **Prog**, so unless you change the value, or use a different control setup, individual programs control the pitch bend range while you're in Program mode. For more about the Bend Range parameter, see page 7-18.

## Monophonic

When off, the program is polyphonic—it can play up to 48 notes at a time. Notice that when the Mono-mode parameter is off, the Legato Play parameter and the three Portamento parameters do not appear on the COMMON page. This is because only monophonic programs can use portamento.

When On, the program will play only one note at a time. This makes it possible to use portamento, so the remaining parameters appear.

```

EditProg:COMMON                               All Layers
Pitch Bend Range:200Oct                        Globals:Off
MonoPhonic      :On                          OutPair:KDFX-A
Legato Play      :On
Portamento      :Off
Portamento Rate :70.0key/s
AttackPortamento:On
<more  OUTPUT  COMMON SetRng  more>
  
```

## Legato Play

When Legato Play is on, a note will play its attack only when all other notes have been released. This is useful for realistic instrumental sounds.

## Portamento

This parameter is either on or off. The default value of **Off** means that portamento is disabled for the current program.

Portamento is a glide between pitches. On actual acoustic instruments like violin and bass, it's achieved by sliding a finger along a vibrating string. On most keyboards that offer portamento, it's achieved by holding down a key that triggers the starting note, then striking and releasing other keys. The pitch glides toward the most recently triggered note, and remains at that pitch as long as the note remains on. The K2661 gives you two ways to get portamento. See the Attack Portamento parameter below.

When you're applying large amounts of portamento to multi-sampled sounds (Acoustic Guitar, for example), the K2661 will play more than one sample root as the pitch glides from the starting pitch to the ending pitch. This may cause a small click at each sample root transition. You can reduce the number of clicks you'll hear by entering the Program Editor and adjusting the KeyTrk parameter on both the KEYMAP and PITCH pages. The quickest way is to set the KeyTrk value on the KEYMAP page to **0**, and to **100** on the PITCH page. This will stretch the sample root that plays at C 4 across the entire keyboard. Now any amount of portamento will play only one sample root, and the clicks will disappear.

There's a tradeoff here, since many sounds will change in timbre as these single sample roots are pitch-shifted during the portamento. This will be most noticeable for acoustic instrument sounds, and may not be noticeable at all for single-cycle waveforms like sawtooth waves. Furthermore, some samples will not glide all the way up to the highest notes—there's a limit to the amount of upward pitch-shifting that can be applied to samples. If this doesn't work for you, you can compromise between the number of clicks and the amount of timbre change by further adjusting the KeyTrk parameters on the KEYMAP and PITCH pages.

As long as the combined values of the KeyTrk parameters on both pages add up to 100, you'll have normal semitone intervals between keys. If you set both parameters to values of **50**, for example, the sound will still play normally, and you'll have several sample roots (about half the number of the original sound) stretched evenly across the keyboard, instead of just one. This will give you fewer clicks than in the original sound, but not as much change in timbre as setting the KEYMAP KeyTrk value all the way to **0**. Set the KEYMAP KeyTrk parameter higher to reduce the change in timbre, or set the PITCH KeyTrk value higher to reduce the number of clicks. Just make sure the combined values add up to 100, to preserve the normal intervals between notes.

## Portamento Rate

The setting for Portamento rate determines how fast the current note glides from starting pitch to ending pitch. The value of this parameter tells you how many seconds the note takes to glide one semitone toward the ending pitch. At a setting of **12 keys/second**, for example, the pitch would glide an octave every second. The list of values is nonlinear; that is, the increments get larger as you scroll to higher values.

## Attack Portamento

This parameter toggles between two types of portamento. When set to **On**, the K2661 remembers the starting pitch so you don't have to hold a note on to achieve portamento. The pitch always glides to each new note from the previously triggered note. When set to **Off**, the pitch will glide to the most recently triggered note only when the previous note is still on (in other words, you must use legato fingering).

## Globals

This is another toggle, which affects LFO2, ASR2, FUNs 2 and 4, and program output routing to KDFX. When off, these four control sources are local; they affect each individual note in the layers that use them as a control source. They begin operating each time a note in that layer is triggered.

When the Globals parameter is set to **On**, these control sources become global, that is they affect every note in every layer of the current program, not just the one to which they're applied. When these control sources are global, they begin operating as soon as the program is selected. When Globals are on, LFO2, ASR2, and FUNs 2 and 4 will appear on their respective pages preceded by the letter G to indicate that they're global.

You'll use global control sources when you want to affect each note in a given layer uniformly, and local control sources when you want to affect each layer's note independently. For example, you'd use a global LFO controlling pitch to create a *Leslie effect* on an organ sound, since you want the affect applied to all the notes you play. You'd use a local LFO controlling pitch to create a vibrato for a solo violin, since you want to be able to vary the rate and depth of the vibrato for each note.

## OutPair

When Globals are off, the OUTPUT page for each layer determines the KDFX input to which the layer sends its audio output. This enables you to apply a different effect to each layer. Turning Globals on overrides the settings on each OUTPUT page, causing every layer in the program to be routed to the KDFX input you specify for the Outpair parameter. This is a quick way to apply the same effect to every layer in the program.

## The Amplitude Envelope (AMPENV) Page

Amplitude envelopes have three sections: attack, decay, and release. The attack section determines how long each note takes to reach its assigned amplitude level after you trigger a Note On event. The decay section determines how quickly and how much a sustained sound fades before a Note Off is triggered. The release section determines how quickly a sound fades to silence *after* a Note Off is triggered.

Press the **AMPENV** soft button to reach the Amplitude Envelope page. For many programs, it will look like the diagram below, which tells you that the amplitude for the current layer is the default, "natural" ROM amplitude envelope that's applied to each sample and waveform during its original development process. You'll leave the amplitude envelope in Natural mode when you don't want to change the way the current layer's loudness develops.

```

EditProg:AMPENV          <>Layer:1/1

                          Mode:Natural

<more  AMPENV  ENV2  ENV3  ENVCTL  more>
  
```

If you want to build your own amplitude envelope, just turn the Alpha Wheel a click. The value **Natural** will change to **User**, and a set of AMPENV parameters will appear. The sound will change when you do this, because the default settings for the User envelope, as shown in the diagram below, take effect as soon as you leave Natural mode. Returning to Natural mode applies the original amplitude envelope once again.

Many programs feature User envelopes with appropriate envelope settings. This is usually the case for programs that use samples of acoustic instruments, since it provides a convenient starting point for you to adjust the envelopes.

```

EditProg:AMPENV [1/1] <>Layer:1/1
Att1:Att2:Att3:Dec1:Rel1:Rel2:Rel3:Loop:
0s 0s 0s 0s 0s 0s 0s Off
100% 0% 0% 0% 0% 0% User Inf
  |
  v
<more AMPENV ENV2 ENV3 ENVCTL more>

```

You'll tweak the parameters on the AMPENV page when you want to shape the amplitude characteristics of your sounds. A graphic view of the amplitude envelope will appear on the display to give you a visual sense of the envelope's characteristics. The dots along the envelope graphic indicate the breakpoints between the envelope's various segments. The small horizontal arrow represents the end of the decay section. The small downward-pointing arrow represents the beginning of the release section.

Because the K2661's ROM samples are stored in a compressed format, applying an altered amplitude envelope can change more than just the amplitude of your sound, since it also changes the rate at which the samples are decompressed for playback. When the samples are made to play back with altered envelopes, the timbres can evolve in new and interesting ways.

The AMPENV page's top line gives you the usual location reminder, points out the currently selected layer, and tells you the relative scale of the envelope's graphic view. The envelope graphic shrinks in scale as the segment times get longer. This auto-zoom feature maximizes the available display space. Try lengthening one of the segment times. The envelope graphic will stretch to fill the display from left to right. When it fills the display, it will shrink to half its size, and the top line will indicate that the scale has changed (from [1/1] to [1/2], for example).

Each parameter on this page has two values, as listed below. For the envelope segments, the first (upper) value is the duration of the segment, and the second is the amplitude level at the completion of the segment. For the Loop parameter, the values define how the envelope loops, and how many times the loop cycles.

Parameter Group	Parameter	Range of Values
Attack Segment 1, 2, 3	Time	0 to 60 seconds
	Level	0 to 100%
Decay Segment	Time	0 to 60 seconds
	Level	0 to 100%
Release Segment 1, 2, 3	Time	0 to 60 seconds
	Level	0 to 100%
Loop	Type	Off, Forward, Bidirectional
	# of loops	Infinite, 1 to 31 times

## Attack Segment Times

These indicate how long it takes for the current layer's amplitude to reach its final level from its starting level.

## Attack Segment Levels

These are the final levels that each segment achieves at completion. The levels are expressed as percentages of the maximum possible amplitude for the current layer. Attack segment 1 always starts at zero amplitude, and moves to its assigned level in the time specified by its time value. So the default settings of **0 seconds** and **100%** mean that the first segment of the attack section moves instantly from zero amplitude to 100% amplitude. Increase the time of Attack segment 1 if you want the sound to ramp up more slowly.

Attack segments 2 and 3 affect the sound only when you set a nonzero value for time. They will then move to their assigned levels in the time specified. Their starting levels are equal to the final levels of the preceding segment.

## Decay Segment

The decay section has only one segment. It has values for time and level, just as for the attack section. The decay section begins as soon as the attack section has been completed. It starts at the same amplitude level as the attack segment preceding it, and moves to its assigned level in the time specified. You'll hear a note's decay section only when the attack section is completed before a Note Off message is generated for that note.

To create a sustaining envelope, simply set the Decay segment's level to a nonzero value.

## Release Segments

Like the attack and decay sections, each of the three segments in the release section has values for time and level. Each segment reaches its assigned level in the time specified for that segment. Release segment 1 starts at the Note Off event for each note, at the current amplitude level of that note—whether it's in the attack section or the decay section. It then moves to its assigned level in the time specified. Release segments 2 and 3 start at the final levels of the segments before them. Release segments 1 and 2 can be set to any level from **0** to **100%**. Release segment 3 always has a level of **0%**, so you can't adjust its level. In place of its Level parameter you see a parameter that lets you toggle between User envelopes and the sound's preprogrammed natural envelope.

## Loop Type

There are seven different values for Loop type.

A value of **Off** disables looping for the current layer's amplitude envelope.

Values of **seg1F**, **seg2F**, and **seg3F** are forward loops. In each case, the amplitude envelope plays through the attack and decay sections, then loops back to the beginning of the first, second, or third attack segments, respectively.

Values of **seg1B**, **seg2B**, and **seg3B**, are bidirectional loops. The amplitude envelope plays through the attack and decay sections, then reverses and plays backward to the beginning of the first, second, or third attack segment, respectively. When it reaches the beginning of the assigned attack segment, it reverses again, playing forward to the end of the decay section, and so on.

## Number of Loops

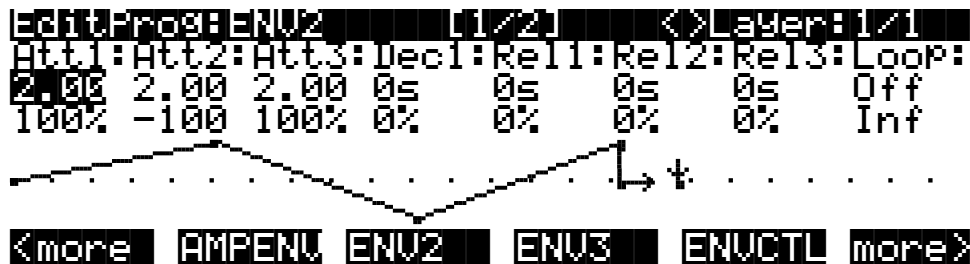
A value of **Inf** makes the amplitude envelope loop until a Note Off is generated. Values of **1** through **31** indicate how many times the loop will repeat after the amplitude envelope has played once through its normal cycle.

Regardless of the loop type and the number of loops, each note goes into its release section as soon as its *Note State* goes off (that is, when a Note Off is generated). The envelope will continue to loop as long as Note State remains on, whether it's held on by a pedal, by the IgnRel parameter (described in the section entitled *The LAYER Page* on page 6-19), or whatever.

## The Envelope 2 (ENV2) and Envelope 3 (ENV3) Pages

The K2661 offers two envelopes in addition to AMPENV. Like AMPENV, ENV2 and ENV3 can be assigned like any other control source. Unlike AMPENV, however, ENV2 and ENV3 can be bipolar. This means that you can set negative values for them. (Obviously, you can't have an amplitude less than zero, so AMPENV is unipolar—the values range from **0** to **100%**.) A bipolar envelope controlling pitch, for example, could modulate the pitch both above and below its original level.

Another difference is that AMPENV *always* controls the amplitude of the layer, so even if you use it as a control source for other functions, it will still affect the layer's amplitude. ENV2 and ENV3 affect only those layers that have them assigned as a control source. Also, AMPENV uses an exponential attack (the amplitude rises much faster at the end of the attack segment than it does at the beginning), while ENV2 and ENV3 use linear attacks (the attack segment increases at the same rate from start to finish).



The pages for Envelopes 2 and 3 are reached with the soft buttons ENV2 and ENV3. When you select these pages, you'll find a display that looks very much like the AMPENV page. The only differences are that you can program an amount for Rel3, and in the envelope graphic, which has a dotted line running horizontally across the display. This is the zero level line; negative level values for the various envelope segments will cause the envelope graphic to dip below this line.



## The Envelope Control (ENVCTL) Page

Envelopes are control sources with outputs that evolve over time without repeating (unless you want them to). You can make the envelopes even more powerful by using envelope control. This gives you realtime control over the rates of each section of the envelopes. Press the **ENVCTL** soft button to reach the ENVCTL page.

```

EditProg:ENVCTL <>Layer:1/1

Adjust:KeyTrk:VelTrk:Source:Depth:
Att: 1.000x 1.000x 1.000x MIDI73 1.000x
Dec: 1.000x 1.000x MIDI72 1.000x
Rel: 1.000x 1.000x MIDI72 1.000x
Imp: 0.0dB 0.00d 0.0dB OFF 0.0dB
<more AMPENV ENV2 ENV3 ENVCTL more>
  
```

The display's top line reminds you of the current layer. The first line of text in the center of the display shows five of the common DSP control parameters: Adjust, Key tracking, Velocity tracking, and Source/Depth.

This page is a table showing the five envelope control parameters, and their values for each of the three sections of the envelopes. Additionally, the line above the soft buttons lets you make use of the Impact feature, which adds an amplitude overshoot to the first 20 milliseconds of a note's attack. It's important to keep in mind that if you set up an envelope control source, it affects Envelopes 2 and 3, as well as the amplitude envelope (Natural or User). Furthermore, the values for the various parameters are cumulative. With the exception of Impact, though, ENVCTL does not affect the attack sections of natural envelopes.

The parameters and values in the following list apply to *each* of the three envelope sections—attack, decay, and release. We'll describe them only once, since their functions are largely the same for each envelope section. The only difference is with velocity tracking, which is hard-wired to control only the attack sections of the envelopes (you can assign attack velocity as the value for the Source parameter in each of the sections, however).

The values of each of these parameters multiply the *rates* of the envelope sections they control. Values greater than **1.000x** make the envelope sections run *faster* (they *increase* the rate), while values less than **1.000x** make the envelope sections run *slower*. Say for example that on the current layer's AMPENV page you had set the Decay section's time at **2.00 seconds**, and its level at **0%**. This sets the layer's amplitude to fade to silence two seconds after the completion of the last attack segment. The decay *time* is two seconds; the decay *rate* is 50% per second. Now if you select the ENVCTL page and set the Decay Adjust parameter to a value of **2.000x**, you've increased the decay *rate* by a factor of two. The rate increases to 100% per second, and the decay time is now one second instead of two.

Parameter Group (Available for each of Att, Dec, Rel, Imp)	Range of Values
Adjust	0.018 to 50.000x (-24.0 to 24.0 dB for Imp)
Key Tracking	0.018 to 50.000x (-2.00 to 2.00 dB for Imp)
Velocity Tracking	0.018 to 50.000x (Not available for Dec or Rel; -24.0 to 24.0 dB for Imp)
Source	Control Source List
Depth	0.018 to 50.000x (-24.0 to 24.0 dB for Imp)

## Adjust

This is the familiar Coarse adjust found on many other pages. Use it here to change the rate of one of the envelope sections without reprogramming the envelope itself. This parameter doesn't give you realtime control over the envelope. It is, however, a good way to adjust the natural envelopes without switching to a User envelope and trying to approximate the Natural envelope.

## Key Tracking

This uses the MIDI note number of each key as the control input for the current layer's corresponding envelope section. When the value of this parameter is greater than **1.000x**, notes above C 4 will make the envelope section run faster, while notes below C 4 will make it run slower. When the value of this parameter is less than **1.000x**, notes above C 4 will make the envelope section run slower, and notes below C 4 will make it run faster. This gives you realtime envelope control right from your MIDI controller. You might use it, for example, to cause an acoustic guitar sound to decay quicker at the high end (set the key tracking to a positive value).

## Velocity Tracking

Use your attack velocity as the control input for the current layer's attack section (this parameter doesn't apply to decay or release). When the value of this parameter is greater than **1.000x**, attack velocities greater than 64 make the attack section run faster, and attack velocities below 64 make it run slower. This gives you realtime attack control over the envelope.

## Source, Depth

These two parameters work together to let you assign a control like the Mod Wheel to affect the current layer's envelopes in realtime. The value of the Source parameter defines which control affects the envelope section, and the value of the Depth parameter defines how much the rate is multiplied when the control is at its maximum.

## Impact

Impact punches the volume during the first 20 milliseconds of the attack of an envelope. Use this feature to get maximum "thump" from your bass and drum sounds.

This feature is incompatible with the K2000. K2000s running operating system version 3.54 or earlier won't even load programs that use Impact. K2000s running operating system version 3.8 or higher *can* load programs that use Impact. The Impact feature won't work, but the programs will play normally otherwise.

## The LFO Page

These are low-frequency oscillators. You'll use the LFO page to define the behavior of the two LFOs available to each layer. LFOs are periodic (repeating) control sources. The basic elements are the rate and shape, which define how frequently the LFO repeats, and the waveform of the modulation signal it generates.

With the K2661, you can set upper and lower limits on each LFO's rate, and assign a control source to change the LFO's rate in realtime, if you wish.

Because of its periodic nature, the LFO is perfect for creating effects like vibrato (cyclic variation in pitch) and tremolo (cyclic variation in amplitude). When you're editing LFOs, or any control source, remember that it must be assigned to control some parameter before you'll hear the effects of your edits.

LFO1 is always local, meaning that it's triggered with each Note On event, and runs independently for each note in the layer. LFO2 is local by default, but can be made global. This is done on the COMMON page, by setting the Globals parameter to **On**, which causes LFO2, ASR2, FUN2 and FUN4 all to become global. Global controls uniformly affect every note in each layer.

```

EditProg:LFO          <>Layer:1/1

LFO1:  MnRate:MxRate:RateCt:Shape: Phase:
      2.00H  0.00H OFF   Sine  Odeg
LFO2:  OFF   0.00H OFF   Sine  Odeg

<more  LFO  ASR  FUN  UTRIG  more>
  
```

The top line of this page gives the usual mode reminder and tells you which layer you're looking at. There are five parameters for each of the LFOs.

Parameter Group (Available for each of LFO1 and LFO2)	Range of Values	Default
Minimum Rate	0 to 24 Hz	2.00 (Off for LFO2)
Maximum Rate	0 to 24 Hz	0.00
Rate Control	Control Source List	Off
LFO Shape	LFO Shape List (Ref. Guide)	Sine
LFO Start Phase	0, 90, 180, 270 Degrees	0

## Minimum Rate

This is the slowest rate at which the LFO runs. When its Rate control is set to **OFF**, or when the control source assigned to it is at its minimum, the LFO runs at its minimum rate.

## Maximum Rate

This is the fastest possible rate for the LFO. When its Rate control is set to **ON**, or when the control source assigned to it is at its maximum, the LFO runs at its maximum rate.

## Rate Control

Assign any control source in the list to modulate the LFO's rate between its minimum and maximum. A continuous control like the Mod Wheel is a natural choice, enabling you to get just about any rate between min and max. But you can use a switch control too, to get just the min or max with nothing in between. Assigning **MPress** as the rate control for an LFO vibrato gives you an easy way to increase the vibrato rate in realtime, as you can on many acoustic instruments.

## LFO Shape

The shape of the LFO waveform determines the nature of its effect on the signal its modulating. There are diagrams of each LFO shape in the *Musician's Reference*; these will give you an idea of how each LFO shape affects the signal. An easy way to check the effects of the different LFO shapes is to set **LFO1** as the value for the Src1 parameter on the PITCH page, and set the Depth

for Src1 to **400 cents** or so. Then go to the LFO page, set the Min and Max rates for LFO1 at **0.00 Hz** and **4.00 Hz** or so, and set the Rate control to **MWheel**. Now play your MIDI controller and you'll hear the LFO's rate change when you move its Mod Wheel. Select different LFO Shapes and check out the effect on the pitch.

LFO Phase

Use this parameter to determine the starting point of the LFO's cycle. One complete cycle of the LFO is 360 degrees. 0 degrees phase corresponds to a control signal value of 0, becoming positive. Each 90-degree increment in the phase represents a quarter-cycle of the LFO.

When an LFO is local, the phase parameter gives you control over the starting point of the LFO for each note (for example, you could make sure every vibrato started below the pitch you played instead of at the pitch you played). The LFO's phase also affects global LFOs, although it's often indistinguishable, since global LFOs start running as soon as the program containing them is selected, even if you don't play any notes.

The ASR Page

ASRs are three-section unipolar envelopes—attack, sustain, and release. The K2661's ASRs can be triggered by a programmable control source, and can be delayed. ASR1 is always a local control. ASR2 is local by default, but becomes global if the Globals parameter on the COMMON page is set to **On**. ASRs are frequently used to ramp the depth of pitch or amplitude in a vibrato or tremolo, enabling delays in those effects. Chapter 20 gives an example of creating a delayed vibrato. The ASR page consists of two rows of five parameters, one row for each of the ASRs.

```

EditProg:ASR          <>Layer:1/1

ASR1:  Trig:  Mode:  Delay:  Attack:Releas:
      ON      Hold  1.00s  1.00s  1.00s
ASR2:  UN      Rept  1.00s  1.00s  1.00s

<more  LFO  ASR  FUN  UTRIG  more>
```

Parameter Group (Available for each of ASR1 and ASR2)	Range of Values	Default Values (for both ASR1 and ASR2)
Trigger	Control Source List	Off
Mode	Normal, Hold, Repeat	Normal
Delay	0 to 60 seconds	0 seconds
Attack	0 to 60 seconds	0 seconds
Release	0 to 60 seconds	0 seconds

Trigger

This defines the control source that starts the current layer's ASRs. The ASR starts when the trigger switches from off to on. If the Trigger parameter is set to **ON**, a global ASR starts running immediately when you select a program that contains it. A *local* ASR starts running as soon as you trigger a note in the layer that contains it. Switch controls are better suited for ASR triggers because of their binary (on/off) nature. A continuous control will trigger the ASRs when its signal value is above its midpoint.

## Mode

This parameter sets the sustain section of the ASR. The ASR's mode determines what the ASR does when it finishes its attack section. If the Mode parameter is set to **Normal**, the ASR will run directly from its attack section to its release section (no sustain). At a setting of **Repeat**, the ASR will cycle through the attack and release sections, then loop forward and cycle through again until the ASR's trigger switches off. If the mode is set to **Hold**, the ASR maintains its position at the end of the attack section until the ASR's trigger switches off. The ASR then goes into its release section. If the ASR's trigger switches off before the attack section is complete, the ASR goes directly to its release section.

## Delay

When the ASR's trigger switches on, the ASR will start immediately if this parameter is set to zero. Nonzero values will cause a corresponding delay between the ASR trigger and the start of the ASR.

## Attack

This defines how long the ASR takes to ramp up from minimum to maximum effect on whatever it's patched to.

## Release

This defines how long the ASR takes to fade to minimum from its maximum. If the ASR's trigger switches off before the ASR has reached maximum, the ASR releases from that level.

# The Function (FUN) Page

FUN is short for function. The K2661's four FUNs greatly extend the flexibility of the control sources. Each FUN accepts input from any two control sources, performs a selectable function on the two input signals, and sends the result as its output, which can be assigned like any other control source. Using the FUNs involves defining them on the FUN page, then assigning one or more of them as control sources. The FUN page looks like this:

```

EditProg:FUN          <>Layer:1/1

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 three parameters for each FUN. Inputs **a** and **b** can be any control source from the Control Source list. The control sources you want to combine are the ones you'll assign as the values for these parameters.

The Function parameter determines what mathematical function is applied to the two inputs. When a FUN has been assigned as a control source, the K2661 reads the values of the two control sources defined as Inputs **a** and **b**. It then processes them according to the setting for the Function parameter, and the resulting value is the FUN's output.

Chapter 17 describes each of these functions, and provides a few diagrams to give you a hint of the immense control (as well as some chaos) that these functions make possible.

## The Velocity Trigger (VTRIG) Page

The velocity triggers base their operation on the attack velocity of each note you play. To use a VTRIG, you simply set its velocity level (threshold), then set it to switch on or off when your attack velocities exceed that threshold. Then assign it as a control source for some other parameter. They’re handy for triggering ASRs, for example.



Parameter Group (Available for each of VTrig1 and VTrig2)	Range of Values	Default
Vel. Trigger Level	ppp to fff	ppp
Vel. Trigger Sense	Normal, Reversed	Normal

The velocity trigger’s level is expressed in terms of the standard dynamic markings of western music—ppp, pp, p, mp, mf, f, ff, and fff. The K2661 converts each attack velocity value it receives into one of these eight levels. When a velocity trigger has been assigned as a control source, the K2661 compares the velocity trigger’s level and sense with the attack velocity values it receives. If the sense is **Normal** and the attack velocity value is greater than the velocity trigger’s level, the trigger switches on. When the velocity trigger’s sense is reversed, the trigger switches on when the attack velocities it receives are lower than the velocity trigger’s level. Keep in mind that you won’t hear the effect of editing the VTRIG page until you’ve assigned a VTRIG as a control source for some other parameter.

## The KDFX Page

This is where you assign a studio to be used with the current program (depending on the settings for the FX Mode and FX Chan parameters—see *FXCtrl* on page 9-4, and *Effects Channel (FX Chan)* on page 9-10).

The KDFX page is the first of four pages containing parameters that enable programs to control their associated studios in real time. For example, you can create (or edit) a program that uses the Mod Wheel or a slider to control the Wet/Dry mix of the signal that goes through FXBus1. The possibilities are almost limitless.

There are 18 sets of these real-time studio-control parameters, which we call FXMods. Each FXMod consists of five parameters. An important point to remember about FXMods is that they are components of a *program* or a *setup*, and they don’t permanently affect the studios they control. They simply enable you to make real-time, temporary changes to the studio settings,

using the physical controller of your choice. FXMods provide a convenient way to gain serious performance flexibility.

```

EditProg*KDFX      H11 Layers
Studio:199 Default Studio

Bus: Param:      Adjust:      Source:      Depth:
Mix  Mix Lvl      0.0dB      OFF          0dB
FX1  Aux Lvl      -12.5dB     MIDI22      11dB
FX2  Aux Lvl      -6.0dB      MIDI22      6dB
<more  KDFX  FXMOD2 FXMOD3 FXMOD4 more>

```

The top line of the display indicates that you're on the KDFX page in the Program Editor, and that any changes you make will affect all layers of the current program. The second line consists of the Studio parameter; this identifies the studio associated with the current program, and enables you to change that studio.

The next line identifies the five parameters that make up each FXMod. Every FXMod uses these same five parameters, although the available values for some parameters depend on the specifications of the current studio, or on the values of other parameters in the FXMod—or both.

- Bus** Indicates which bus the FXMod will affect. Every FXMod is associated with a single KDFX bus. This can be any of the KDFX buses, which include Input A (**InA**) through Input D, FXBus1 (**FX1**) through FXBus4, the Aux bus (**Aux**), and the Mix bus (**Mix**).
- Param** Determines which aspect of the current studio the FXMod will affect. The available values for this parameter depend on both the value of the Bus parameter, and the specifications of the current studio.
- Adjust** Like the Adjust parameter on other editor pages, this enables you to set a “starting point” for your real-time control over the specified aspect of the studio. For example, you might want to zero a Wet/Dry mix before using a slider to vary it, or change a cutoff frequency to enhance the harmonics in a particular sound or effect. The available values for this parameter depend on the value of the Param parameter. The values you set for the Adjust parameter are not cumulative, as they are in other editors; they override the programmed settings for the studio used by the program.
- Source** Specifies which control source you'll use to modify the studio in real time. You can use any global control for the source.
- Depth** The amount that the specified aspect of the studio will be modified by the specified source. This is in addition to the amount specified by the Adjust parameter. The available values for Depth depend on the value of the Param parameter.

On the Program-editor page above, the first FXMod affects the level on the Mix bus, overriding whatever mix level is defined for the studio.

The second FXMod cuts the level of the send from FXBus1 to the Aux bus by 12.5 dB. Slider B (MIDI 22) boosts the Aux level by 11 dB at the top, and returns it to the Adjust value at the bottom.

The third FXMod does pretty much the same thing as the second, except that it does it to FXBus2, cuts the signal less, and boosts it less when you move Slider B up.

## The FXMOD2–FXMOD4 Pages

These pages are a continuation of the KDFX page. Unlike the KDFX page, these pages each list five FXMods, and they don't indicate the current studio. Otherwise, they're identical to the KDFX page, listing the five parameters that make up each FXMod.

```

EditProgram:FXMOD2          H11 Layers
Bus: Param: Adjust: Source: Depth:
FX1 Mix Lvl 5.5dB OFF 0dB
FX2 VibChInOut Out MIDI22 1
FX2 Vib/Chor V2 FXFUN1 5
FX2 Lo Rate -0.50Hz FXASR2 -6.20H
FX2 Hi Rate 0.50Hz FXASR2 6.20H
<more KDFX FXMOD2 FXMOD3 FXMOD4 more>

```

## The FXLFO, FXASR, and FXFUN Pages

If you've read the descriptions of LFOs, ASRs, and FUNs that begin on page 6-40, you already know most of what you need to know about these three pages. They contain the parameters for the two LFOs, two ASRs, and four FUNs that the K2661 provides for KDFX control. With two exceptions, these control sources operate like the "regular" LFOs, ASRs, and FUNs. The differences are that the FX versions are global (they affect all layers in a program instead of individual layers), and they're available only for control of KDFX. Otherwise they're the same as the regular LFOs, ASRs, and FUNs (the regular versions can be used to control KDFX as well). In fact the descriptions for the regular LFOs, ASRs, and FUNs apply to their KDFX counterparts. Please see pages 6-40 through 6-44 if you need more information.

## The ImportKDFX Page

You can import studios and FXMods from existing programs or setups. Press the **ImpFX** soft button to bring up the ImportKDFX page.

```

EditProgram:ImportKDFX
From Program 1 Concert Piano 1
      (Studio 49 Sndbrd Room Hall )
Prog Setup Import Cancel

```

There's just one parameter, which tells you the program or setup from which to import. Use the **Prog** or **Setup** soft button to toggle between the list of programs and the list of setups. In this case, we're about to import the studio and FXMods from Program 1 **Concert Piano 1** (which uses Studio 49 **Sndbrd Room Hall**). As you scroll through the list of programs or setups, the display indicates which studio is assigned to the program or setup.

When you press the **ImpFX** soft button, the K26610 copies the studio and FXMod settings from the specified program or setup, and applies them to the current program. You then return to the page you were on when you pressed the **ImpFX** soft button.



## Function Soft Buttons

The remainder of this chapter describes the soft buttons that perform specific functions, as opposed to selecting programming pages. The descriptions below are arranged in the order in which you would see the soft buttons if you pressed the **more>** button repeatedly. You can always get to these buttons, regardless of which page is currently selected.

### Set Range (SetRng)

The **SetRng** soft button gives you a quick way to set the lowest and highest notes in the currently selected layer. Press this button, and the K2661 will prompt you to trigger the note you want to set as the low note for the layer. Press the **Cancel** soft button if you change your mind. Otherwise, trigger the desired note on the K2661 keyboard or a MIDI controller. When you trigger a note, the K2661 prompts you to trigger the note you want to be the highest in the layer. When you trigger another note, the previously selected page returns, and the notes you triggered will be recorded as the new values for the LoKey and HiKey parameters on the LAYER page. You'll notice that the higher of the two notes you triggered is entered as the HiKey value, regardless of the order in which you triggered the two notes.

### Name

Call up the page that enables you to change the name of the current program. See page 5-3.

### Save

Start the process of saving the current program. See page 5-3.

### Delete

Delete the current program from RAM. You can also delete any other RAM program by scrolling through the list that appears when you press the **Delete** soft button, then pressing Delete again when the desired program is selected. If you attempt to delete a ROM program, the K2661 will say it's deleting the program, but it doesn't actually do it. See page 5-6.

### Dump

Send a MIDI System Exclusive dump of the current program's settings. See the *Musician's Reference* for more information about System Exclusive messages.

### New Layer (NewLyr)

Create a new layer, numbered one above the highest existing layer. The new layer's parameters are those of the single layer in Program 199, called **Default Program**. When you press this button, the K2661 will tell you that it is creating a new layer, then will return to the page you were on. The new layer becomes the current layer, and is the highest-numbered layer in the program. If the current program already has its maximum number of layers, the K2661 will tell you that you can't add any more.

Program 199 makes a good template for programs that you build from the algorithm up. You might want to edit Program 199 to adjust one or more parameters to values you want to use in your template program. If you like the settings of the default layer as they are, however, remember not to make any permanent changes to Program 199.

## Duplicate Layer (DupLyr)

Create a copy of the current layer, duplicating the settings of all its parameters. The copy becomes the current layer, and is the highest-numbered layer in the program.

## Import Layer (ImpLyr)

Copy a specific layer from another program into the current program. This button brings up a dialog that prompts you to select a layer number and a program number. The dialog tells you the currently selected layer, and the total number of layers in the program. Use the **Layer-** or **Layer+** soft buttons (or the **Up/Down** cursor buttons) to change the layer number. If the current program has only one layer, pressing these buttons will have no effect. Use **Prog-** or **Prog+** soft buttons (or the **Left/Right** cursor buttons) to change the program number.

While you are in this dialog, you can listen to the layer you are selecting to import, along with all other layers in the current program. If you want to hear the layer to be imported by itself, you must mute the other layers.

When you have selected the desired layer from the desired program, press the **Import** soft button, and the selected layer will be copied from the selected program, becoming the current layer. Importing layers is a convenient alternative to creating layers from scratch. If you have a favorite string sound, for example, and you want to use it in other programs, just import its layer(s) into the program you're building. This will preserve the envelopes and all the control settings so you don't have to reprogram them.

## Delete Layer (DelLyr)

Delete the current layer. When you press this button, the K2661 asks you if you want to delete the layer; press the **Yes** soft button to start the deletion process, or the **No** soft button to cancel it. This prompt prevents you from accidentally deleting a layer.

## Editing KB3 Programs

You can edit a wide assortment of any KB3 program's parameters. You can also create your own KB3 programs, though you must start with an existing KB3 program to do this. A regular K2500 program cannot be turned into a KB3 program. If you're not sure whether the current program is a KB3 program, check the information box on the left side of the program display; it will indicate "KB3 Program" if that's what the program is.

Enter the KB3 program editor by pressing the Edit button while a KB3 program is selected in program mode. You'll quickly see that the KB3 editor differs from the standard VAST program editor.

## The TONEWL Page

By default, KB3 Mode uses DSP-generated waveforms for the lower half of its tone wheels and samples for the upper half of its tone wheels. Using the parameters on the TONEWL page, you can specify the waveforms and samples you wish to use, the number of tone wheels (which will affect how many other voices are available to you), and other related settings. You can even

switch the tone wheels, so that samples are used for the lower tone wheels and waveforms are used for the upper ones.

```

EditProg: TONEWL
UpperToneWheels:163 Sine Wave
LowerToneWheels:SINE2 LowerXpose:0ST
UpperVolAdjust:-2dB UpperXpose:0ST
NumToneWheels:79 LowestPitch:C 2
Upper/LowerSwap:Off
WheelVolumeMap:Bright OrganMap:Peck's
[more] [TONEWL] [ORGAN] [Setup] [Pitch] [more]
  
```

Parameter	Range of Values
Upper Tone Wheels	Sample List
Lower Tone Wheels	Sine, Sine2, Saw, Square
Upper Volume Adjust	-96 dB to 96 dB
Number of Tone Wheels	24–95
Upper/Lower Swap	Off, On
Wheel Volume Map	Equal, Bright, Mellow, Junky
Lower Transposition	-120 to 127 Semitones (-168 to 79 if Upper/Lower Swap is <b>On</b> )
Upper Transposition	-176 to 79 Semitones (-128 to 127 if Upper/Lower Swap is <b>On</b> )
Lowest Pitch	C 0 to C 7
Organ Map	Equal, Peck's, Bob's, Eric's

## Upper Tone Wheels

Use this parameter to indicate the keymap (and thereby the samples) to use for the upper tone wheels. You can use any ROM or RAM keymap, though you must specify a keymap that uses looped samples for KB3 Mode to work correctly. When in Program mode, the keymap assigned to the program appears in the info box.

## Lower Tone Wheels

Here you can specify the waveform to use for the lower half of the tone wheels. Choose from SINE, SINE2, SAW, and SQUARE. SINE2 is an improved version of SINE, with less distortion.

## Upper Volume Adjust

Since sample volumes can vary, while the volume of DSP-generated waveforms will remain consistent, you may find it necessary to adjust the level of the sample-based tone wheels. This parameter lets you adjust the amplitude of the upper (sample-based) tone wheels relative to amplitude of the waveform-generated tone wheels.

## Number of Tone Wheels

This parameter lets you specify the number of tone wheels used by a KB3 program. The classic tone wheel organs used 91 tone wheels, though the lowest 12 were for the pedals only. Therefore, you may find 79 a good number of tone wheels to specify for realistic organ

emulations. This would leave you eight voices for other programs. You can specify up to 95 tone wheels.

Here's how to do the math to calculate polyphony: the number of K2500 voices used by a KB3 program is  $(\text{number of tone wheels} + 1) / 2$ , rounded to the next highest whole number if the result is a fraction. So, for example, with 79 tone wheels specified you would use 40 voices. Keep in mind that these voices are permanently allocated and running while the KB3 program is selected, and cannot be stolen. The additional voice used by KB3 programs, by the way, is for keyclick.

### Upper/Lower Swap

Use this parameter to swap the upper and lower tone wheel groups (that is, to use sample-based tone wheels for the lower positions and waveform-generated tone wheels in the upper positions). Setting this parameter to **On** changes the available values for upper and lower transposition.

### Wheel Volume Map

The wheel volume map determines the volume level for each tone wheel. We've provided several tone wheel volume maps here, based on measurements we've taken on different organs. **Equal** is a map with all tone wheels at the same volume. It's not based on a real B3. **Bright** is a good normal map, based on a B3 in good condition. **Junky** is based on a B3 with an uneven, rolled-off response. **Mellow** is somewhere between **Bright** and **Junky**.

You can also apply EQ to control wheel volumes base on the frequencies of each tone wheel. See page 6-59.

### Organ Map

The organ map controls the relative amplitude of each key, per drawbar. Like the wheel volume maps, these maps are based on measurements we've made on actual organs. **Equal** uses the same volume for each key and drawbar, and is not base on a real B3. **Peck's** is a good normal map, from a B3 in good condition. **Eric's** is a bit more idealized; it's smoothed out, but less realistic. **Bob's** is more uneven, based on an old B3.

### Lower Transpose / Upper Transpose

These two parameters let you transpose the upper and/or lower tone wheels in semitone steps away from their default tunings. The available ranges of values for these parameters depends on the setting of the Upper/LowerSwap parameter.

### Lowest Pitch

Here you can specify the pitch of the lowest tone wheel, which is typically set to C2. The rest of the tone wheels—as many as you have specified in the NumToneWheels parameter—will be tuned in semitone steps above this pitch.

## The DRAWBR Page

Press the **Drawbr** soft button to view the DRAWBR Page. This page lets you edit KB3's drawbars. Remember that your K2661's sliders function as Drawbars 1-8, while the Mod Wheel is Drawbar 9.

```

EditProg: DRAWBR
Mode: Preset Steps: 0-8
Vol : 0  0  0  0  0  0  0  0  8
Tune: -12 7  0  12 19 24 28 31 36
<more TONEW DRAWBR SetDBR PITCH more>
  
```

### Mode

When you set Mode to **Preset**, the preset drawbar settings on this page will be installed at program selection. The drawbar values will immediately change, however, as soon as you move the corresponding drawbar. Set Mode to **Live** if you want the drawbar volume settings at program selection to be determined by the positions of the drawbar controllers (sliders and Mod Wheel). With either setting, the drawbar controllers will affect drawbar volumes subsequent to program selection.

### Steps

This parameter lets you specify the increments by which drawbar volumes will change. Choose either **0-8**, to approximate the drawbar settings on actual organs, or choose **0-127** for a finer degree of resolution.

### Volume

This parameter only appears only if you've set Mode (see above) to **Preset**. Use the Volume parameter to set the preset volume of each of the nine drawbars. The available values will be **0-8** or **0-127**, depending on the setting of the Steps parameter.

### Tune

This parameter lets you tune each of the nine drawbars up or down in semitone steps. The values for the Tune parameter on the DRAWBR page shown above represent standard drawbar settings on a real B 3, as shown in Table 6-1 on page 6-5.

## The SetDBR Soft Button

Press the **SetDBR** soft button to capture the current position of the drawbars, and use those positions as the preset drawbar positions on the DRAWBR page.

# The PITCH Page

The PITCH page for KB3 programs is much like the PITCH page for VAST programs. The only difference is that for KB3 programs, there are no FineHz, KeyTrk, or VelTrk parameters. For a full description of the PITCH-page parameters, see page 6-27.

```
EditProg:PITCH
Coarse:031 Src1 :OFF
Fine :0ct Depth :0ct
Src2 :OFF
DptCtl:MWheel
MinDpt:0ct
MaxDpt:0ct
<more ALG LAYER KEYMAP PITCH more>
```

# The PERC Page

Percussion is a characteristic feature of tone wheel organs. It’s especially useful while soloing, since percussion adds an extra plink (actually an extra tone at a defined harmonic) to the attack. You can reach the percussion parameters by pressing the Perc and Perc2 soft buttons.

```
EditProg:PERC
Percussion:Off LowHarm :DrawBar4
Volume :Soft HighHarm :DrawBar5
Decay :Slow StealBar :DrawBar9
Harmonic :Low
VelTrack :0%
<more PERC PERC2 KEYCLK AMP more>
```

Parameter	Range of Values
Percussion	Off, On
Volume	Soft, Loud
Decay	Slow, Fast
Harmonic	Low, High
Velocity Tracking	0–100%
Low Harmonic	Drawbar 1–9
High Harmonic	Drawbar 1–9
Steal Bar	None, Drawbar 1–9

## Percussion

This is where you turn the percussion effect on or off. Percussion is created by a decaying envelope applied to one of the nine drawbars. The percussion effect is “single-triggered,” which means that once it’s triggered, it won’t trigger again until all keys (or whatever you’re using to trigger notes) go up. So if no keys are down, and you play a chord, percussion gets applied to all notes in the chord (and in fact, to all notes that are triggered during the short duration of the percussion envelope). Once the envelope runs its course, any notes you play while at least one

key is held down get no percussion. On keyboard models, you can turn percussion on or off by pressing Assignable Controller Button 5 (**Mute** button 5).

## Volume

This parameter switches between loud and soft percussion settings. The actual amplitude is set on the PERC2 page. On keyboard models, you can toggle between loud and soft by pressing Assignable Controller Button 6 (**Mute** button 6).

## Decay

This parameter switches between fast and slow percussion settings. The actual decay rate is set on the PERC2 page. On keyboard models, you can toggle between slow and fast decay by pressing Assignable Controller Button 7 (**Mute** button 7).

## Harmonic

This parameter switches between high and low harmonic percussion settings. The actual pitch is controlled by the LowHarm and HighHarm parameters. On keyboard models, you can toggle between low and high harmonics by pressing Assignable Controller Button 8 (**Mute** button 8).

## VelTrack

Here is where you specify the degree to which key velocity controls percussion volume. A value of zero corresponds to no velocity tracking, which is like a real tone wheel organ. Other values add velocity tracking, so that increased velocity results in louder percussion.

## LowHarm

Controls which drawbar is used as the basis for the percussion when Harmonic is set to **Low**. On an actual tone wheel organ, this is Drawbar 4 (2nd harmonic). The actual pitch obtained depends on the drawbar tuning.

## HighHarm

Controls which drawbar is used as the basis for the percussion when Harmonic is set to **High**. On an actual tone wheel organ, this is Drawbar 5 (3rd harmonic). The actual pitch obtained depends on the drawbar tuning.

## StealBar

Controls which drawbar is disabled (if any) when the percussion effect is turned on. On an unmodified tone wheel organ, the ninth drawbar is the one disabled. Any drawbar can be selected, including **None**.

# The PERC2 Page

```
EditProg:PERC2
PercLevel: DecayTime: OrgLevel:
Loud+Fast: 7.0dB 0.44s -2.0dB
Loud+Slow: 7.0dB 0.60s -2.0dB
Soft+Fast: 0.0dB 0.24s 0.0dB
Soft+Slow: 0.0dB 0.60s 0.0dB
<more PERC PERC2 KEYCLK AMP more>
```

Parameter Group (Available for each combination of the Volume and Decay parameters on the PERC page)	Range of Values
Percussion Level	0 to 24.0 dB
Decay Time	0.10 to 5.10 seconds, in .02-second increments
Organ Volume Level	-12.0 to 12.0 dB

## PercLevel, DecayTime, OrgLevel

With these parameters you can control the amplitude and decay time of the percussion effect for all combinations of the Volume and Decay parameters (on the PERC page). You can also adjust the level of the organ relative to the percussion, for accurate emulation of classic organs.

# The KEYCLK Page

The Key Click feature adds a decaying burst of pitched noise to the attack of notes. Unlike the percussion, the key click is “multi-triggered,” which means that every new note will trigger it. The parameters on this page primarily control the decay, volume, and pitch of the key click.

```
EditProg:KEYCLK
KeyClick:On Random :30%
Volume : -34.0dB RetrigThresh: -31.5dB
Decay : 0.005s
Pitch : -23ST NoteAttack : Normal
VelTrk : 66% NoteRelease : Normal
<more PERC PERC2 KEYCLK AMP more>
```



Parameter	Range of Values
Key Click	Off, On
Volume	-96.0 to 0.0 dB, in .5-dB increments
Decay Time	0.005 to 1.280 seconds, in .005-second increments
Pitch	-129 to 127 Semitones
Velocity Tracking	0–100%
Random	0–100%
Retrigger Threshold	-96.0 to 0.0 dB, in .5-dB increments
Note Attack	Normal, Hard, PercHard
Note Release	Normal, Hard

## KeyClick

This is where you turn Key Click on or off.

## Volume

This parameter sets the level of the keyclick; the noise decays from the level you set here. This level is scaled by the drawbar levels, as well as the expression pedal level.

## Decay

Sets the basic decay time of the noise envelope. Smaller values produce a shorter burst.

## Pitch

Sets the basic pitch of the key click noise, relative to the highest tone wheel's pitch. The pitch is actually controlled by a steep lowpass filter applied to white noise.

## VelTrk

Controls the degree to which key velocity affects the key click volume. A value of zero means that the key velocity has no effect on the key click volume (which is like a real tone wheel organ). Other values add volume as the velocity increases.

## Random

Controls the degree to which a random amount of amplitude variation is added to the key click.

## ReTrigThresh

This parameter lets you set the volume level below which key click must decay before it will be retriggered.

## Note Attack

Controls the attack characteristic of notes. **Normal** provides a smoothed attack, while a setting of **Hard** has an instant attack and will produce an audible click, in addition to any amount of key click specified with the other parameters on this page (you might prefer not to specify any additional key click when you use this setting). **PercHard** sets a hard attack level for percussion only; notes without percussion use a normal attack.

## Note Release

Controls the release characteristic of notes. A setting of **Normal** has a smoothed release, while a setting of **Hard** has an instant release. **Hard** will produce an audible click.

## The AMP Page

Assign amplitude controllers on this page, which is similar to the F4 AMP page for VAST programs (see page 6-28 for a full description). The only difference between the AMP pages for KB3 programs and VAST programs is that the KB3 version doesn't have the KeyTrk, VelTrk, and Pad parameters.

```

EditProg:AMP
Adjust:0dB
Src1 :OFF
Depth :0dB
Src2 :OFF
DptCtl:MWheel
MinDpt:0dB
MaxDpt:0dB
<more PERC PERC2 KEYCLK AMP more>
  
```

## The OUTPUT Page

Use this page to route the programs signal from the sound engine to KDFX. This page is similar to the VAST-program version; the only difference is that it doesn't have the CrossFade and XFadeSense parameters. See page 6-29 for a description of the OUTPUT-page parameters.

```

EditProg:OUTPUT
Pair:KDFX-E
Pan :L * R
Mode:+MIDI
Gain:18dB
<more OUTPUT MISC EQ more>
  
```

## The MISC Page

The MISC page contains an assortment of control parameters, including Leslie speed control and vibrato/chorus selection.

```

EditProg:MISC
PreampResp:On      VolAdjust :21dB
Leakage    :-88.0dB BendRange :0ct
LeakMode   :TypeA   Sustain   :On
SpeedCtl   :Slow    Sostenuto :On
VibChorCtl:On
VibChorSel:Chor3
<more  OUTPUT  MISC  EQ  more>
  
```

Parameter	Range of Values
Preamp/Expression Response	Off, On
Leakage	-96.0 to 0.0 dB, in .5-dB increments
Leak Mode	None, Type A, Type X, Type Y, Type Z
Speed Control	Slow, Fast
Vibrato/Chorus Control	Off, On
Vibrato/Chorus Type Selection	Vib1, Vib2, Vib3, Chor1, Chor2, Chor3
Volume Adjust	-96 to 96 dB
Bend Range	-7200 to 7200 Cents
Sustain	Off, On
Sostenuto	Off, On

### PreampResp

Set this parameter **On** or **Off** to enable or disable the preamp+expression pedal part of the KB3 model. Turning this **On** (the default) makes KB3 programs function like stock organs. The expression pedal in this case is more than a volume pedal; it actually functions like a “loudness control,” varying the frequency response to compensate for the ear’s sensitivity at different volumes. In addition, the preamp provides a deemphasis curve to compensate for the built-in tone wheel volume preemphasis. Turning preamp response **Off** emulates organs that have been modified to have a direct out (before the preamp and expression pedal).

### Leakage

Controls the level of the simulated crosstalk and signal “bleed” of adjacent tone wheels in the model. This is provided to help “dirty up” the sound to make it a bit more realistic. A setting of -96 dB gives the purest tones; other values add more simulated leakage. This level is scaled by the drawbar levels, as well as the expression pedal level.

### LeakMode

Selects between different leakage models, determining which leakage harmonics are emphasized. **TypeA** provides an overall tone wheel leakage, with all tone wheels leaking a small amount. **TypeX**, **TypeY**, and **TypeZ** emulate different degrees of drawbar leakage, where the leakage components correspond to the nine drawbars, instead of all the tone wheels.

## SpeedCtl

Select either **Fast** or **Slow** to choose the speed of the rotary speaker emulation. On keyboard models, you can toggle between fast and slow speed using Assignable Controller Button 1 (**Mute** button 1).

When you select a KB3 program, the K2661 sends several MIDI Controller messages both locally and to the MIDI Out port. One of those messages is Controller number 68, with a value corresponding to the value of SpeedCtl (**Slow** = 0, **Fast** = 127).

## VibChorCtl

Choose **On** or **Off** to turn on or off either vibrato or chorus (as selected with the VibChorSel parameter). On keyboard models, you can toggle between on and off using Assignable Controller Button 2 (**Mute** button 2).

When you select a KB3 program, the K2661 sends several MIDI Controller messages both locally and to the MIDI Out port. One of those messages is Controller number 95, with a value corresponding to the value of VibChorCtl (**Off** = 0, **On** = 127).

## VibChorSel

Choose the vibrato or chorus program (there are three of each) you wish to use with this KB3 program. Note that you must set VibChorCtl (also on the MISC page) to **On** to hear the effect. On keyboard models, you can select the vibrato or chorus you want using Assignable Controller Buttons 3 and 4 (**Mute** buttons 3 and 4).

When you select a KB3 program, the K2661 sends several MIDI Controller messages both locally and to the MIDI Out port. One of those messages is Controller number 93, with a value corresponding to the value of VibChorSel (**Vib1** = 0, **Vib2** = 36, **Vib3** = 58, **Chor1** = 79, **Chor2** = 100, and **Chor3** = 122).

## VolAdjust

Provides an overall volume adjust for the KB3 model. Use this parameter to “normalize” KB3 programs with other programs.

## BendRange

Controls the pitch bend range of this KB3 program.

## Sustain

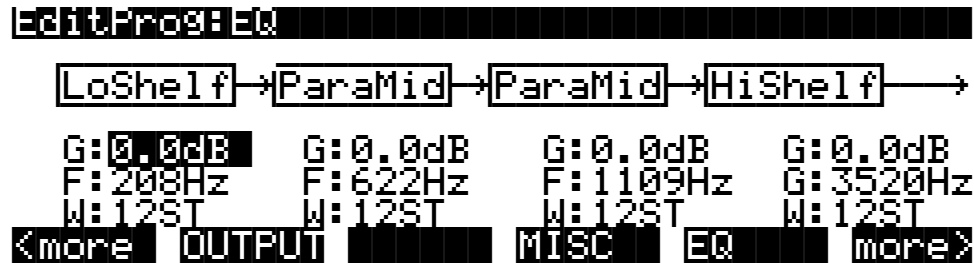
Set **On** or **Off** to enable or disable response to MIDI sustain (MIDI 64).

## Sostenuto

Set **On** or **Off** to enable or disable response to MIDI sostenuto (MIDI 66).

## The EQ Page

When you press the **EQ** soft button, you'll see a page that at first glance looks quite similar to the ALG page for a VAST program. The four blocks on this page, however, represent two shelving bands of equalization and two parametric bands. The KB3 EQ offered here, though, is not implemented as a true EQ section, instead it adjusts the volume of the tone wheels based on frequency. If the tone wheels are based on sine waves, then this acts a lot like a real EQ.



Parameter Group (Available for each EQ Block)	Range of Values
Gain	-24.0 to 24.0 dB, in 0.2-dB increments
Frequency	0 to 25088 Hz, in varying increments
Width	-128 to 128 Semitones, in 2-semitone increments

Each EQ section has Gain (G), Frequency (F), and Width (W) controls. Frequency controls the center frequency of the band. Width controls the bandwidth. Gain controls the amount of boost or cut.

## All the Other Pages

The rest of the pages—LFO, ASR, FUN, etc.— are the same for KB3 programs as they are for VAST programs, so we won't describe them again here. Begin on page 6-40 to find descriptions of these pages.

## Programming Tips

This section provides some starting points for creating your own KB3 programs. Remember that you'll have to start with one of the existing KB3 programs.

As described below, the most prominent difference between organ vintages is the number of tone wheels used. Keep in mind, however, that the sound of an actual tone wheel organ will depend not only on its age, but also on how well it has been maintained.

Octave folding, where an octave (or part of an octave) is repeated at the top or bottom of the keyboard, is handled automatically by KB3 Mode, emulating the folding done on actual tone wheel organs.

**Early Tone Wheel Organs.** Instruments of this period had 91 tone wheels. To get this sound, go to the **TONEWL** page, select 91 tone wheels, and set lowest pitch to C 1. Start with the **Junky** Wheel Volume Map and **Bob's** Organ Map. You may also want to increase the Key Click level, since this tends to become louder on older organs.

**Middle Period Organs.** To model one of these instruments, set 82 tone wheels and a low note of A 1. Use the **Mellow** Wheel Volume Map and **Eric's** Organ Map. Set Key Click to a moderate level.

**The Classic B-3.** For this sound, choose 79 tone wheels and set the low note to be C 2. The best settings here are the **Bright** Wheel Volume Map and **Peck's** Organ Map. You may also want to reduce the Key Click level.

## Using a KB3 Program in a Setup

To get the rotary speaker effect to work properly, you'll need to make a few adjustments. On the Effects-mode page, make sure that FX Mode is **Auto**, and FX Chan is **Current**. Then go into the Studio Editor, and assign a KB3 program to one of the setup's zones. Then import the KDFX settings from a program that uses the rotary effect you want. If the setup has only one zone, you're finished. If the setup has more than one zone, you'll need to check each zone to see where the audio output for its program is going. If you don't want the rotary effect applied to the other programs in the setup, set those programs' outputs to a KDFX input that's different from the output of the KB3 program (you'll need to look at the setup's studio, find which FXBus uses the rotary-effect FX preset, then set the output of each non-KB3 zone so that it goes to a KDFX input that isn't routed to the FXBus using the rotary-effect FX preset).



***Note:** Many of the factory KB3 programs use Studio 160 KB3 V/C -> Rotary, which in turn uses FX Preset 779 KB3 FXBus on FXBus1, and also uses the Aux Bus for the full rotary-speaker effect. Since FX Preset 779 uses 4 PAUs, there are no PAUs free for use on the other FXBuses. Consequently, when you use this studio, there's no way to route other zones through different effects. If you want a setup that uses a KB3 program with rotary-speaker effects, but you also want other programs in the setup that don't use the same effect, use a studio other than Studio 160.*

### Emulating a Two-Manual Organ

You can use Setup Mode to emulate a two-manual keyboard. However, you can only use a KB3 program in one zone of the setup (use a regular K2661 organ program in the other zone). Also, you will have limited polyphony on the non-KB3 channel. The actual number of voices that will be available on the non-KB3 channel depends on the number of tone wheels used by the KB3 program. (You may be able to reduce the number of tone wheels used, depending on the note range of the zone.)

## Programs Using 2PARAM SHAPER

If you are running a KB3 program, you will not be able to simultaneously play K2661 programs that use the 2PARAM SHAPER DSP function on another channel. (Actually, the program will still play, but the 2PARAM SHAPER DSP function will be bypassed.)

### Shaper AMP (!AMP)

Although the !AMP DSP function is not part of the KB3 editor, we have used the !AMP to help "dirty-up" several programs. This allows us to recreate the tube distortion characteristics of classic drawbar organ pre-amps.

## Using the VAST Program Editor on a KB3 Program

KB3 Mode lets you use certain VAST-related parameters—such as the !AMP DSP function mentioned above—that are not found within the KB3 editor. There's a “back door” that takes you to these non-KB3 parameters: if you mark a page (with the Mark button) while editing a VAST program, you can jump to that page (with the Jump button) while you're editing a KB3 program.

When editing a KB3 program, we recommend that you edit only those parameters on the KB3-editing pages. We realize, however, that some power users (you know who you are!) will want to hear the results of applying VAST parameters to KB3 programs. Before you start experimenting, please be aware of the following:

- **Turn your volume down!** In KB3 mode, most resources are being used in much different ways than they in a typical VAST program; the results of applying VAST parameters to KB3 programs can be extremely unpredictable.
- Most changes made to these parameters do not take effect until save the program and reselect it.
- You cannot import a KB3 program layer into a regular VAST program.





# Chapter 7

## Setup Mode and the Setup Editor

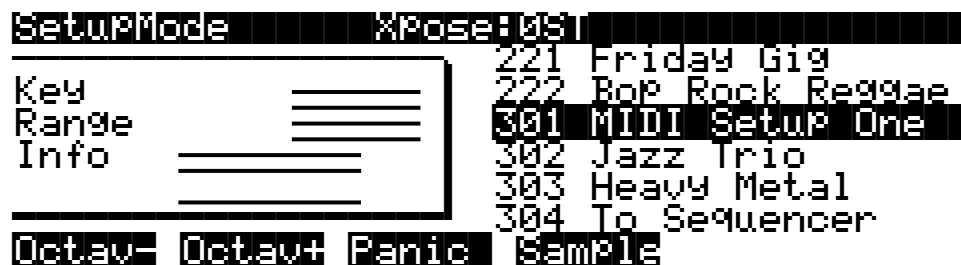
### Setup Mode

In Setup mode, the K2661 can take on the identity of eight distinct instruments and eight distinct MIDI transmitters, each of which can use the setup's physical controller assignments (or any subset of those controller assignments). For example, you can create a setup that is split into eight different keyboard regions (called zones). Each zone can play its own program, while also transmitting on its own MIDI channel.

Selecting setups in Setup mode is much like selecting programs in Program mode—just use one of the normal data entry methods to scroll through the list of setups.

There are some important differences between a program and a setup. A program plays on a single keyboard zone and on a single MIDI channel. A setup enables you to use up to eight keyboard (or MIDI controller) zones, each of which can have its own program, MIDI channel, and control assignments. The parameters you define for each setup affect programs *only while you are in Setup mode*. An exception to this is the control setup, which we discuss on page 7-2.

Press the Setup-mode button to enter Setup mode. You'll see a list of setups, which you can select with any data entry method.



The lines in the info box represent the approximate key range of each zone, and let you know if any zones overlap. In the preceding diagram, the setup has seven active zones (Zone 7 is turned off); Zones 1–4 are at the upper end of the keyboard. Zones 5, 6, and 8, which overlap Zones 1–4, cover the lower two thirds of the keyboard.

For setups containing three or fewer zones, the box displays the MIDI channel and program assignments for each zone, with lines under the Program names to indicate the key range of each zone (as shown in the following diagram). An **L** or an **M** next to the channel number indicates that the zone transmits only locally or via MIDI (the default is Local *and* MIDI).

**Off** indicates that the zone has been turned off completely (when a zone is turned off, no MIDI, program, or key-range information is visible for the zone).

```

SetupMode      Xpose:451
Chan/Program Info
1   36 Cool Traps
2   676 S19 Acoust
3L 122 Izit Jimmy
Octav- Octav+ Panic Sample
222 Bop Rock Reggae
301 MIDI Setup One
302 Jazz Trio
303 Heavy Metal
304 To Sequencer
305 MIDI Setup Two

```

You can transpose the entire setup up or down with the two **Octav** soft buttons. Press them simultaneously to set the transposition back to zero. When you transpose a setup, the split points between zones remain in place; each program is transposed within its respective zone.

The **Panic** soft button sends All Notes Off and Reset All Controllers messages to all zones. The **Sample** soft button provides convenient access to the K2661's sampler. Refer to Chapter 14 for complete information on the sampler.

When you select a setup in Setup mode, the K2661 sends a number of MIDI messages, on each of the MIDI channels used by the setup. Some of these include: Program Change commands, MIDI Bank Select messages, Pan and Volume messages, and entry values for physical controllers (entry values are the values that take effect as soon as you select the setup; there are also controller *exit* values, which are the values of the controllers when you leave the setup—either by selecting another setup or by exiting Setup mode). The values of all these messages depend on the parameters you define in the Setup Editor.

## Loading Older Setup Versions

You can load setups created on the K2000, K2500, or K2600 into the K2661. Setups created on the K2500 or K2600 are the same as those created on the K2661, so they're completely interchangeable. Since setups created on the K2000 have fewer features (like three zones instead of eight), you have two choices for using K2000 setups on the K2661. You can leave the K2000 setups as they are, and they'll work for either the K2000 or the K2661—but they won't have all the K2661 features like eight zones. Or you can edit the K2000 setups to take advantage of the K2661's expanded features. Once you do this, however, the setups will no longer work on the K2000, so if you want to keep using those setups on a K2000, be sure to keep backup copies of the original setups.

If you edit a K2000 setup to use K2661 features, you may need to reassign the programs it uses. If your K2000 setup uses a K2000 ROM program, the setup won't play the same program when you load it into the K2661, because the two instruments have different lists of ROM programs. In this case, you can either select a similar program for the K2661 setup (if one exists), or you can save the programs used by the K2000 setup into the same ROM IDs in the K2661 (this doesn't really replace the K2661 programs; they reappear when you delete the K2000 programs). If your K2000 setup uses K2000 RAM programs, you won't have any problems as long as you load dependent objects when you load the K2000 setup into the K2661.

## The Control Setup

In addition to zone splitting and layering, Setup mode is a powerful way to take advantage of the K2661's programmable sliders, ribbon controllers, and assignable buttons. In order to provide some of the same flexibility for Program mode, we created the control setup, which defines the controller assignments for programs in Program mode.

The default control setup is **97 Control Setup**, but you can choose any control setup you want. To do this, go to the MIDI-mode TRANSMIT page and use any normal data entry method to change the CtlSetup parameter. When you reenter Program mode, all programs will now respond to many of the controller assignments defined in Zone 1 of the control setup (Zones 2–8 are not relevant in Program mode, because a program can occupy only one MIDI channel).

To edit the control setup, press the **Edit** button while the CtlSetup parameter is highlighted on the MIDI-mode TRANSMIT page. This brings you to the Setup Editor, which is described in the following sections. The following table shows which control-setup parameters affect controller assignments in Program mode.

Control Setup–Setup Editor Page, Zone 1	Parameters Affecting Program Mode	Parameters Not Affecting Program Mode
CH/PROG	ZoneArpeg, Destination, MIDIBankMode	LocalPrg, Out, Channel, MIDIBank, MIDIPrg, Status, EntryProgChg
KEY/VEL	VelScale, VelOffset, VelCurve	LoKey/HiKey, Transpose, NoteMap, LoVel/HiVel
PAN/VOL	None	
BEND	All	
COMMON	Sync	
ARPEG	All	
RIBCFG	All	
Continuous Controller assignment pages (SLIDER, SLID/2, CPEDAL, RIBBON, WHEEL, PRESS)	Dest, Scale, Add, Curv	Ent and Exit
Switch Controller assignment pages (FOOTSW, SWITCH)	SwType, Dest, On, Off	Ent and Exit
KDFX, FXMOD2, FXMOD3, FXMOD4, FXLFO, FXASR, FXFUN	None	

**Table 7-1 Control Setup Parameters Affecting Program Mode**

In summary, physical controller destinations, their curves and states, and the Arpeggiator parameters all define controller assignments for programs in Program mode. The other parameters have no effect; this keeps Program mode relatively simple. Program mode lets you change values for transposition, MIDI channels, and programs independently of the control setup.

Once you save changes to the control setup, those changes will affect all programs when you are in Program mode. For example, programming the Large Ribbon in the control setup to have three sections will mean that in every program in Program mode, you will have a three-section Large Ribbon.

You may want to program several different control setups, and switch among them for different applications. Suppose, for example, that you're recording a song, but you don't want to record aftertouch. You can create a setup with pressure turned off in Zone 1 (on the PRESS page in the

Setup Editor, set the value of the Press parameter to **Off** for Zone 1). Now whenever you want to record without aftertouch, just select this setup as the control setup.

## Zone-status LEDs in Setup Mode

Take a minute to scroll through some of the factory setups. As you change setups, you'll notice that the LEDs in the eight buttons above the programmable sliders go on and off and change color. These LEDs indicate the status of each of the zones in the setup. You may also see the **Solo** button go on. This means that the setup is configured to have only one zone playing when you select it.

While you're in Setup mode, each of the eight zone-status LEDs will always be in one of four states:

- |        |                                                                                                                                                                                                                                                                                                                                                   |
|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Off    | Empty zone—that is, a zone that has no program or MIDI channel associated with it. For example, if you select a setup and only four status LEDs light up (regardless of their color), the setup contains just four zones. Whenever you're in Setup mode, the number of lines in the info box matches the number of zone-status LEDs that are lit. |
| Red    | Soloed zone. As you might have guessed, only one zone can be soloed at a time. When a zone is soloed, <i>only</i> that zone plays notes and generates controller information. Other zones, if they're not turned off, still generate program changes and entry/exit controller values.                                                            |
| Green  | Active zone. As long as no other zone is soloed, an active zone plays notes— and generates controller information, program changes, and entry/exit controller values. If another zone is soloed, an active zone is “backgrounded”—it's status LED remains green, but it doesn't play notes or generate controller information.                    |
| Orange | Muted zone. Muted zones don't play notes or generate controller information, but they do generate program changes and entry/exit controller values.                                                                                                                                                                                               |

In performance situations, the zone-status buttons provide a convenient way to temporarily change the status of one or more zones. This can be very effective for bringing voices and/or controller configurations into and out of your performance. The best way to get familiar with this technique is to play with the buttons, as the next few paragraphs describe.

Select a setup (look for one with lots of active zones that cover the whole keyboard), and play a few bars. You'll hear sounds corresponding to each of the active zones (green LEDs). If you see any muted zones (orange LEDs), press their zone-status buttons, and they'll become active. Play around a bit. Try muting all the zones, then bringing them back one by one until all the zones in the setup are active.

Now press the **Solo** button. The **Solo**-button LED lights (it's always red), and one of the zone-status LEDs (never more than one) turns red. You'll now hear only that zone as you play. All the active zones are now backgrounded—they'll still generate program changes and entry/exit controller values, but you won't hear anything from them.

Now press one of the zone-status buttons. Its LED turns red, and it becomes the soloed zone. The previously-soloed zone returns to its programmed status. Solo each zone in turn, using the info box in the display to find the range covered by the zone. Note that you can solo a zone even if it's muted.

Press the **Solo** button again, and its LED goes out. The previously-soloed zone returns to its programmed status, and you can hear all the active zones again.

Table 7-2 gives you a quick visual reminder of how zones behave depending on their status.

LED Color	Zone Status	Data Generated by Zone			
		Notes	Controllers	Program Number	Entry and Exit Values
Red	Soloed	✓	✓	✓	✓
Green (no others are red)	Active	✓	✓	✓	✓
Green (another is red)	Backgrounded			✓	✓
Orange	Muted			✓	✓
(Off)	Empty				

**Table 7-2 Zone Status in Setup Mode**

Remember that any changes you make to zone status in Setup mode are temporary; as soon as you select another setup, that setup's programmed zone status takes over. To change a setup's zone status permanently, use the Setup Editor (see *Status* on page 7-8).

## The Setup Editor

From Setup mode, press the **Edit** button to enter the Setup Editor, where you can make changes to the currently selected setup. Use the soft buttons to select the various Setup-editor pages. The upper line of each page displays the usual mode reminder, as well as the current Setup-editor page, and the current zone. Use the **Chan/Bank** buttons to select one of up to eight different zones, each having its own set of Setup-editor pages.

The parameters on the Setup-editor pages define what each of a setup's zones sends—both to internal programs and to the MIDI Out port. They also determine how the K2661 responds to MIDI signals received from a MIDI controller connected to the K2661's MIDI In port (when the Local Keyboard Channel matches the transmit channel of your MIDI controller).

The display diagrams you see in this chapter show the default values for setup **97 ControlSetup**.

# The Channel/Program (CH/PROG) Page

This is the first page you see when you enter the Setup Editor. Here, you can select programs, MIDI channels, and MIDI Bank numbers for each of the setup’s eight zones. You can also solo or mute each zone, and assign zones to be controlled by the K2661’s Arpeggiator.

```

EditSetup:CH/PROG          <>Zone1/1

LocalPrg:1 Concert Piano 1  Out:Prog
Channel :1      Destination :Local+MIDI
MIDIBank:0      MIDIBankMode:Ctl 32
MIDIProg:1      EntryProgChg:On
Status :Active  ZoneArpeg :On
<more CH/PRG KEYVEL PANVOL BEND more>

```

Parameter	Range of Values	Default
Local Program	Program list	1 Concert Piano 1
Channel	1-16	1
MIDI Bank	Variable, dependent on MIDI Bank Mode	0
MIDI Program	Variable, dependent on MIDI Bank Mode	1
Status	Status list	Active
Output assignment (Out)	Prog, KDFX-A to KDFX-D	Prog
Destination	Destination list	Local+midi
MIDI Bank Mode	MIDI Bank Mode list	Ctl 32
Entry Program Change	On/Off	On
Zone Arpeggiation (ZoneArpeg)	On/Off	On

## Local Program (LocalPrg)

This selects an internal program to play on each zone. As you change the value of Local Program, notice that MIDI Program and MIDI Bank match the local program and bank numbers (Program IDs 0–99 correspond to MIDI Bank 0, 100–199 are MIDI Bank 1, and so on). If you want to transmit different program and bank numbers over MIDI, highlight either MIDI Program or MIDI Bank and select a new value. Note that changing the Local Program parameter again will reset both the MIDI Program and MIDI Bank parameters to match the local program and bank numbers.

## Channel

The Channel parameter defines the MIDI transmit channel for the currently selected zone. You can set it to any of the 16 MIDI channels. Normally, you will want each zone on a separate MIDI channel. This is necessary if you want to combine different programs in the setup.

If two zones have the same MIDI channel (and destination), but they have different program settings, there will be conflicts: no MIDI device, including the K2661, can respond correctly to two different simultaneous Program Change commands on one channel. The result will be that only one Program Change will be recognized, and every note played will sound double (if Note Maps are on). This can create odd and unpredictable timing effects, and will reduce your polyphony by 50%.

Nevertheless, there will be occasions when “stacking” zones on the same MIDI channel might come in handy. Suppose you want a physical controller on the K2661 to send data for two *different* numbered MIDI Controllers on the *same channel*. In this case, you must create two zones assigned to the same channel, but with different controller assignments.

Here’s one example: if a receiving synth is using Controller #1 for modulation depth and Controller #13 for modulation speed, you can increase both the depth and the speed with Slider A. Start by assigning Slider A in Zone 1 to **MWheel** and in Zone 2 to **MIDI 13**; then assign both zones to the same MIDI channel. (You may want to make sure you aren’t sending doubled notes. Use the Note Map parameter on the KEY/VEL page to set one zone’s Note Map to **Linear** and the other zone’s Note Map to **Off**.)

Another example: create two or more zones that are identical except for their transposition settings. Now you can play parallel intervals (or chords) with single keystrokes.

## MIDI Bank

The K2661’s programs are divided into ten MIDI banks, numbered 0-9. Program 99 in Bank 1, for example, is **199 Default Program**. The MIDI Bank parameter displays which bank the current program is assigned to, and automatically changes to match the Local Program value you set.

You can send Bank Select messages to external MIDI devices as well, by setting the Destination parameter to a value of **MIDI** or **Local + MIDI**, then changing MIDI Bank. Some instruments may have more than ten banks; the MIDI Specification says a device can have up to 16,384 banks, and the K2661 gives you access to all of them. Bank switching via MIDI makes it easy for the K2661 user to select sounds on external instruments, no matter how many banks they might have.

Different MIDI banks are accessible depending on the value of the MIDIBankMode parameter:

Value of MIDIBankMode	Available MIDI Banks
Ctl 0 or Ctl 32	0–127
Ctl 0/32	0–16383
K2000 or K1000	0–9
None	None

When you change the value of the LocalPrg parameter, the value of MIDIBank automatically changes correspondingly. If you want to transmit a MIDI bank number different from the one corresponding to the local program, select the local program first, then change the MIDI bank.

If you select an empty bank (like Bank 16, 383), the zone will still produce sound on the K2661, provided that Destination is set to **Local** or **Local + MIDI**. The LocalPrg parameter will display whatever internal program you set, but the bank number transmitted over the MIDI Out port will be different from the internal program’s bank number.

## MIDI Program (MIDIProg)

MIDI Program defines which program number is transmitted out the MIDI Out port on the current zone's MIDI channel.

When you change the value of the LocalPrg parameter, the value of MIDIProg automatically changes correspondingly. If you want to transmit a MIDI program change number different from the one corresponding to the local program, select the local program first, then change the MIDI program.

## Status

This parameter determines what the current zone does when you select the setup in Setup mode. If the value is **Active**, the zone sends and receives normally via MIDI. **Muted** means that the zone sends and receives program changes and entry/exit controller values, but doesn't play notes. **Soloed** causes only the current zone to play, "backgrounding" all other zones (backgrounded zones send and receive program changes and entry/exit controller values, but don't play notes). With a value of **Solo/M**, the current zone is both soloed and muted; when the zone isn't soloed, it won't produce sound).

See the section about zone-status LEDs (page 7-4) for more information about muting and soloing zones.

For setups containing four or more zones, pressing **Mute 1** mutes the current zone; the zone becomes unmuted when you mute another zone. Pressing **Mute 2** or **Mute 3** solos the current zone. You can then press the **Chan/Bank** buttons to solo each zone.

## Output Assignment (Out)

This determines the signal routing for each zone in the setup—not the final audio output, but the KDFX input to which the zone's signal gets sent. You can route the zone to KDFX A through D, or you can use a value of **Prog**, which uses the output assignments of the program that's assigned to that zone.



## Destination

This parameter determines whether the currently selected zone transmits only to the K2661 (**Local**), gets sent only via MIDI (**MIDI**), controls both the K2661 and connected MIDI instruments (**Local+MIDI**), or is disabled (**Off**). Setting Destination to **Off** turns the zone off completely; no MIDI data will be sent and no local program will sound.

## MIDI Bank Mode

The MIDI Bank mode you choose determines how bank numbers will be sent over MIDI when the setup is selected, and in what format. It also affects how many MIDI banks and programs you can choose.

**None** means no bank number is sent, just the program number. **Ctl 0** means that the bank number is sent as a MIDI Controller #0 message. **Ctl 32** means it is sent as MIDI Controller #32. **Ctl 0/32** means it is sent as a dual-controller (two-byte) message, with the most-significant byte (MSB) of the bank number sent as Controller #0 and the least-significant byte (LSB) as Controller #32. Single-byte Bank Select messages (either 0 or 32) allow you to specify banks numbered 0-127. Two-byte messages allow you to specify banks numbered 0-16,383. With 128 programs per bank, this allows you to access 2,097,152 different programs on one instrument.

The MIDI Specification is a little ambiguous when it comes to Bank Select messages, as to whether they should be only Controller 0, only Controller 32, or both Controllers sent as a pair. Different manufacturers design their instruments to respond to different schemes, and if you send Bank Select in a form an instrument doesn't like, it may ignore it or interpret it incorrectly. The MIDI Bank Mode parameter is designed to allow the greatest flexibility in addressing other MIDI instruments. Usually you can look on the MIDI Implementation chart in the user's manual of an instrument to determine how it's designed to receive Bank Select messages, and then set MIDI Bank Mode for each zone to suit the instrument that is receiving data from it. The default setting, which works with the largest number of other instruments, is **Ctl 32**.

There are two other values for this parameter, which will be of special interest to owners of other Kurzweil instruments. A value of **K2000** is intended for use with the K2000, K2500, K2600, or K2661. The Bank Select message is sent as Controller 32, with a value between 0 and 9. Remember, the K2000, K2500, K2600, and K2661 support only 10 banks, with 99 programs per bank, so Program Changes 100 or higher are sent as Bank Select 1, followed by the last two digits as a Program Change. For example, if Program 124 is assigned to the zone, this will be sent out the MIDI Out port as Bank Select (Controller 32) 1, and then Program Change 24.

**K1000** is intended for use with any of the 1200-series keyboards or modules, or any of the 1000-series instruments that have version 5 software installed. Those instruments predate the adoption of standard Bank Select messages; instead, they use Program Changes 100–109 as Bank Selects. If you select Bank 5: Program 42 for a K2661 zone, for example, it will send out Program Change 105 followed by Program Change 42. K2661 program numbers over 99 are not sent.

## Entry Program Change (EntryProgChg)

This enables or disables bank and program change commands sent to internal programs or to the MIDI Out jack when you select setups. If it's set to **On**, the program numbers for the programs in the eight zones will be sent via MIDI when a setup is selected. By setting this parameter to **Off**, you can select a setup on the K2661 without changing the internal programs or those on MIDI devices receiving from the K2661. This is useful if you want to send only controller data to the K2661 or to MIDI devices, without changing program assignments.

## Zone Arpeggiation (ZoneArpeg)

Zone Arpeggiation determines whether the Arpeggiator will affect notes played in the current zone. The Arpeggiator affects only those zones that have this parameter set to a value of **On**.

For any given zone, the Arpeggiator plays notes only within that zone's Key Range. If the Arpeggiator, for example, tries to play a C<sup>#</sup>4 in a zone, but that zone's Key Range ends at C4, the note will not sound. However, another zone whose Key Range ends at C5 *will* be able to play the C<sup>#</sup>4 from the Arpeggiator. Therefore, setting a zone's Key Range can be important in deciding how it will respond to the Arpeggiator. (A separate set of range parameters, found on the ARPEG page, determines whether the notes you play get arpeggiated; see *Low Key (LoKey)* and *High Key (HiKey)* on page 7-36.)

## The Key/Velocity (KEY/VEL) Page

The Key / Velocity page allows you to set key range, velocity range, transposition, and Note Maps for each zone.

```
editsetup:KEY/VEL <>zone1/1
LoKey:C -1 Transpose:0ST
HiKey:G 9 Notemap :Linear
VelScale :100%
LoVel:1 VelOffset:0
HiVel:127 VelCurve :Linear
<more> CH/PRE KEYVEL PANVOL BEND <more>
```

Parameter	Range of Values	Default
Low Key	C -1 to G9	C -1
High Key	C -1 to G9	G9
Transpose	-128 to +127 Semitones	0 semitones
Note Map	Note Map list	Linear
Low Velocity	1-127	1
High Velocity	1-127	127
Velocity Scale	±300%	100%
Velocity Offset	-128 to +127	0
Velocity Curve	Velocity Curve list	Linear

## Low Key (LoKey), High Key (HiKey)

The LoKey and HiKey parameters define the note range of the currently selected zone. The easiest way to change these values is to press the **SetRng** soft button, which you can find by pressing either of the **more** soft buttons. You'll be prompted to trigger the notes you want to be the lowest and highest notes for the zone. When you do, you'll return to the Setup-editor page, and the notes you triggered will be reflected in the values for LoKey and HiKey. They'll also be represented by the lines beneath the program names in the box at the left of the Setup-mode page. You can set these values with normal data entry methods as well.

You can create "negative" ranges as well. To do this, select the HiKey parameter and set its limit *lower* than the LoKey limit. This results in the zone being active at the top and bottom of the

keyboard, but being silent in the range between the two limits. This lets you create a layer with a “hole” in the middle, which you can then fill with a different sound on another zone.

Note that using the **SetRng** soft button won’t allow you to create a negative range, since it always defines the higher keystroke as the HiKey value. If you want to set HiKey lower than LoKey (or vice versa), use a normal data entry method. Note: Intuitive Entry doesn’t work for setting values for LoKey and HiKey.

The limits of MIDI are C-1 to G9. The untransposed 88-key range is A0 to C8. The untransposed 76-key range is E1 to G7.

## Transpose

This changes the pitch of the zone, without changing its position on the keyboard. It changes the MIDI note numbers generated by the keys in the zone, without physically shifting the zone. The range is  $\pm 127$  semitones. Since there are 12 semitones (or half steps) to an octave, you can transpose up or down over ten octaves. If you transpose out of the range of the active voice, however, no notes will sound; MIDI note numbers will transmit, but notes will not.

## Note Map

Note Map lets you change the way notes are sent from the K2661. The default setting is **Linear**: all notes go out as played. Pressing the **Minus** button takes you to **Off**; no notes are sent, but controllers and other non-note data are.

Next comes **Inverse**, which turns the keyboard upside-down, with the highest key being A 0 and the lowest C 9. If you set Note Map to **Constant**, all of the keys on the keyboard will play the same note. The note defaults to C4, but you can change this with the **Transpose** parameter. This works well when you want the sound from a particular key to play with every note of another layer—for example, playing a ride cymbal with every note in a bass line.

Next are the alternating Note Maps, which let you divide the keyboard in some unique ways. If you are using two or more MIDI devices (including the K2661), you can expand polyphony by assigning each zone to a different alternating Note Map. For example, if you have two K2661s, you can assign two zones to each play the same program on a different K2661, thereby doubling polyphony.

To split a zone into one of two alternating Note Maps, set Note Map to **1 of 2**; now the zone plays on every second key, starting on C, but won’t play on any other keys. Set another zone to **2 of 2**, and this zone will play on every second key, starting on C<sup>#</sup>, thus covering the remaining keys.

Three- and four-zone alternating Note Maps work the same way, but cause each zone to play only on every third and every fourth key, respectively.

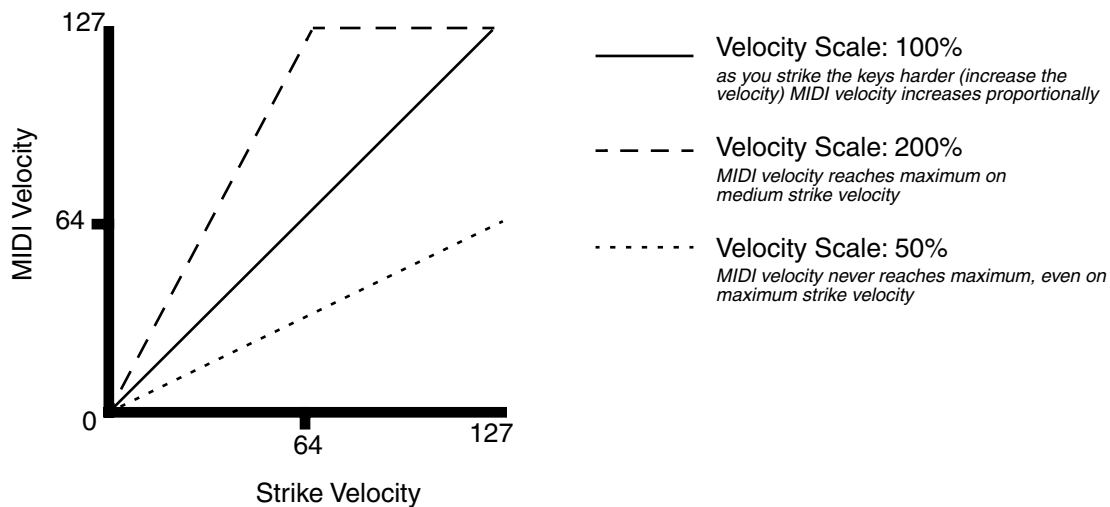
**White** and **Black** allow you to compensate for the differing velocity responses of white versus black keys on some keyboards. By choosing White for one zone and Black for another, you can modify velocity curves and limits independently for the white and black keys.

Note maps are also used to create drum patterns with the arpeggiator. Several of the ROM setups use this feature.

## Velocity Scale (VelScale)

This lets you amplify or diminish velocity response. Normal response is **100%**. Higher values make the keyboard more sensitive (you don't need to play as hard to get higher MIDI velocities) while lower values make it less sensitive (playing harder doesn't change MIDI velocity as much). You can also set the scale to a negative number, in which case the velocity response is turned upside-down: playing harder produces a softer sound and vice versa. This is useful for creating velocity-based crossfades between zones. See the following section on Velocity Offset for ideas about negative scaling.

The following illustration shows what happens when you change Velocity Scale. Note that Velocity Scale is the only parameter changed in this example; the other parameters are set to their defaults (offset = **0**, curve = **linear**, min = **1**, max = **127**).

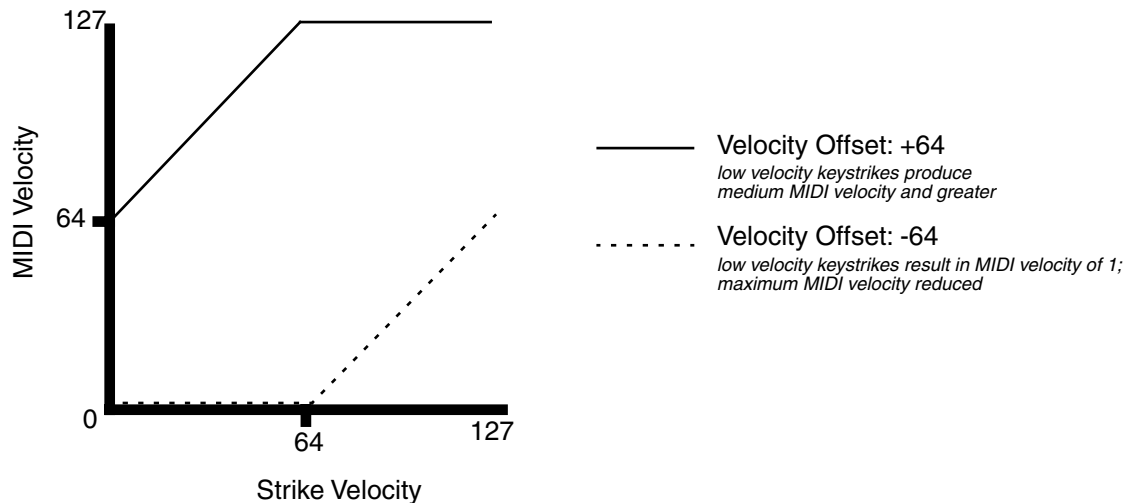


## Velocity Offset

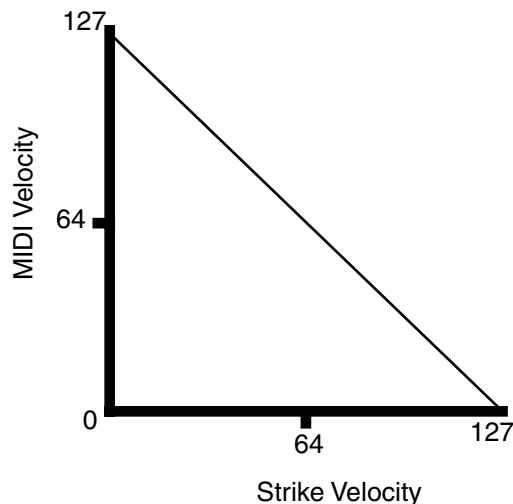
**VelOffset** also changes the response, but in a more direct way, by adding or subtracting a constant to the key velocity. For example, if this is set to **25** (assuming a scale of 100%), then 25 is added to the velocity of every keystroke, usually making the sound that much louder. The softest possible keystroke will have a value of 25, while a keystroke with velocity of 102 will produce the same sound as a note with velocity 127 ( $102+25=127$ ). Negative values diminish the response: a setting of **-25** means the loudest velocity available will be 102, while any keystroke 25 or below will produce a velocity of 1 (a velocity value of zero has a special meaning in MIDI and cannot be used for Note Ons).

You can think of Scale as being a proportional change to the velocity, while Offset is a linear change. The maximum values for Offset are  $\pm 127$ . The following illustration shows the effects of

Velocity Offset. Note that Velocity Offset is the only parameter changed in this example; the other parameters are set to their defaults (scale = **100%**, curve = **linear**, min = **1**, max = **127**).



Offset and Scale work together. If scaling takes the velocity out of the ballpark — for example, you want to set it to **300%** but that puts *all* of your notes at maximum velocity — using a negative offset, say around **-60**, can make it possible to still play at different volumes, although your curve will still be a lot steeper than normal. If you use a negative scaling, then you must use an offset: otherwise all of your velocities will end up as zeroes (well, ones actually, since a MIDI note-on with velocity zero is something else). So to get true inverse scaling (that is, minus 100%), you must set an offset of **127** to get the full range of velocities. Setting the offset to **127** and the scale to **-100%** produces a slope like this (which is the same as the reverse linear curve):



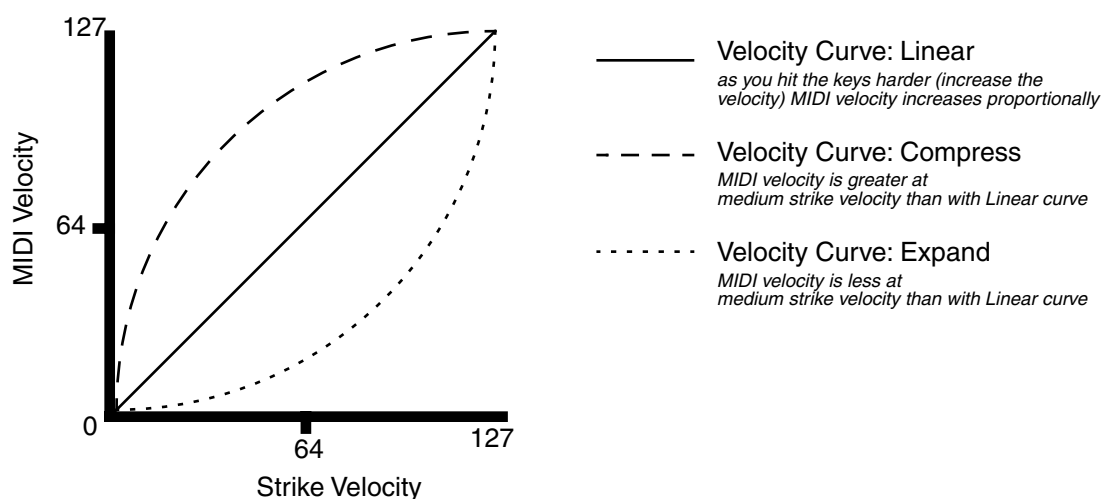
Note that Offset and Scale affect only *MIDI* velocities; that is, these parameters don't change Velocity Tracking in the programs themselves. Therefore, some programs (such as organ sounds, which often have low VelTrk values) may respond only subtly to Offset and Scale, or not at all.

## Velocity Curve (VelCurve)

VelCurve lets you taper the velocity response. The default setting is **Linear**, which means that the output velocity changes directly proportionally to the played velocity.

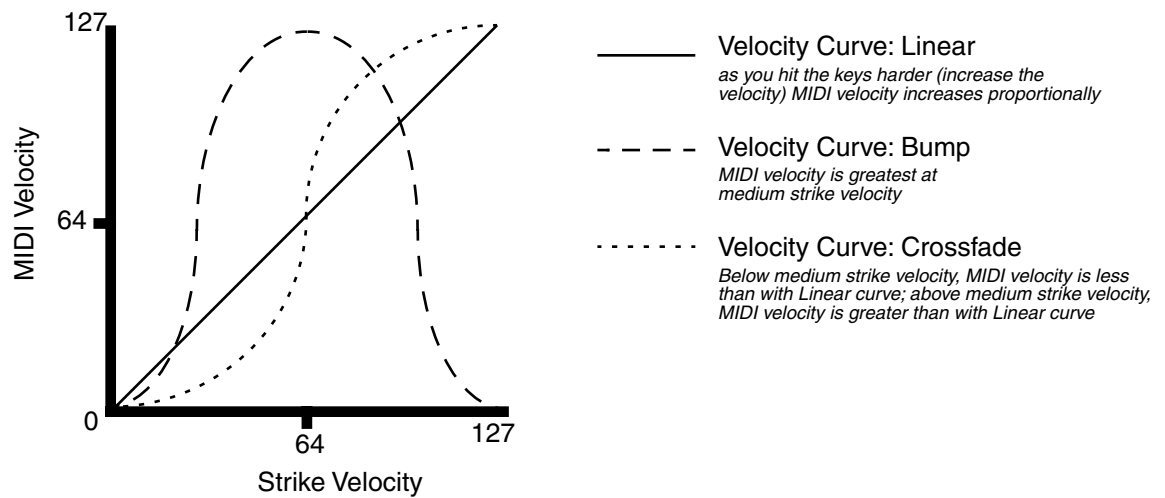
**Expand** produces a curve that is less steep than the linear curve at keystrike velocities below 64, and steeper than the linear curve at keystrike velocities above 64. In other words, when you're playing softly, you'll notice velocity differences less than with a linear curve, while when you're playing hard, you'll notice velocity differences more.

**Compress** produces a velocity curve that is the opposite of the expanded curve—that is, you'll notice velocity differences more when you're playing softly than when you're playing hard.

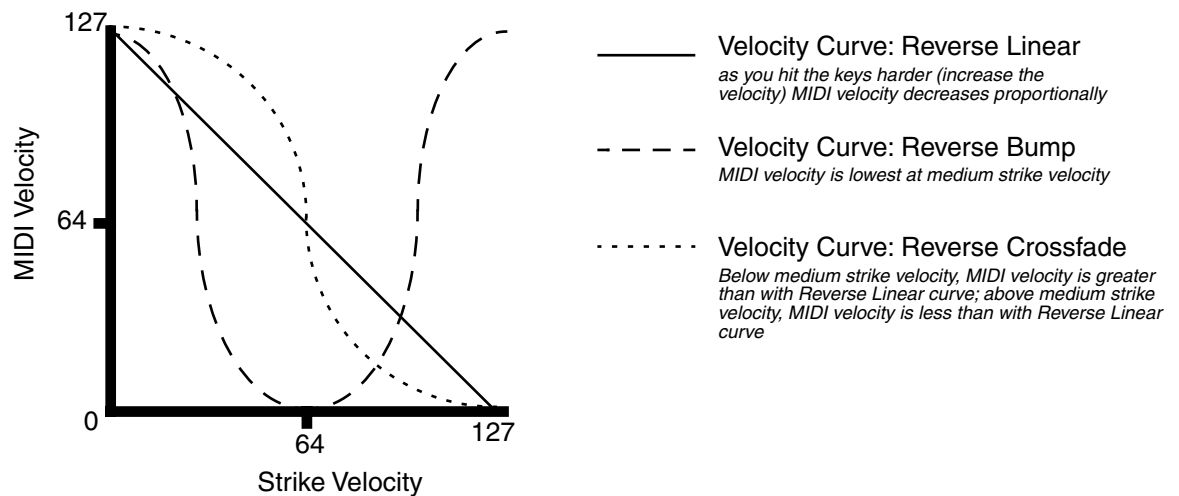


Crossfade (**Xfade**) is designed to be used in tandem with the Reverse Crossfade curve, enabling you to perform smooth crossfades between different programs.

**Bump** tapers velocity response to resemble a bell curve, so that notes are loudest when your keystrike velocity is 64. Notes get softer as the keystrike velocity approaches 0 or 127.



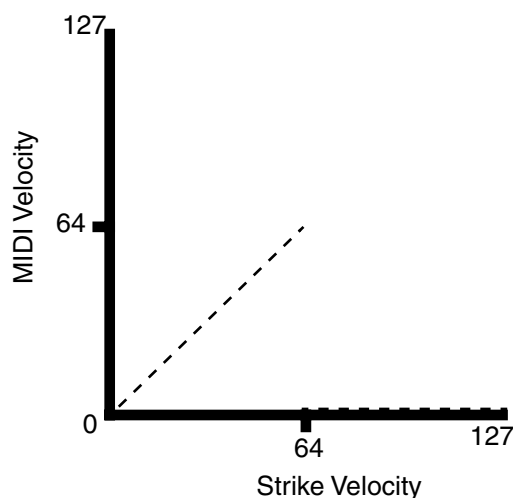
The next five velocity curves are Reverse Linear (**revrsLin**), Reverse Expand (**revrsExp**), Reverse Compress (**revrsCmp**), Reverse Crossfade (**revrsXfd**), and Reverse Bump (**revrsBmp**). These taper velocity in reverse of the five curves we just covered. For example, Reverse Linear's response is such that striking a key harder will produce a lower volume, striking it softer will produce a higher volume, and so on. This provides a convenient way to achieve negative scaling, by letting you set one parameter instead of two.



## Low Velocity (LoVel), HighVelocity (HiVel)

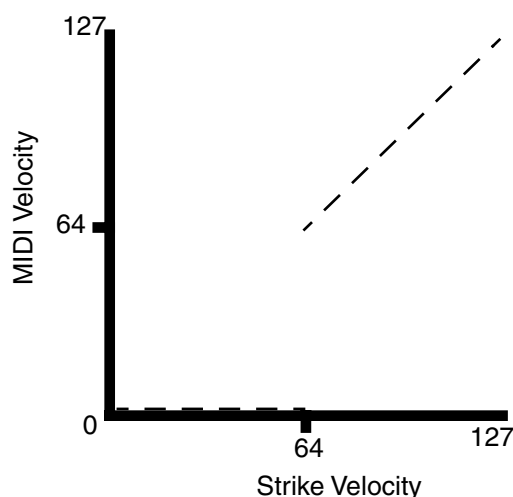
LoVel and HiVel set the minimum and maximum velocity limits that the current zone transmits. A keystroke in the current zone whose velocity — *after* it has been scaled and offset — is below the minimum does not generate a Note On. Neither does a keystroke whose velocity after processing is above the maximum. These parameters are useful for “velocity switching”—having a key play different sounds depending on how hard you strike it.

The values can be anywhere from **1** to **127**. As with other parameters, zones can overlap or be totally discrete, or be identical. Usually, LoVel will have a smaller value than HiVel, but as with LoKey and HiKey, you may also create a gap in velocity response, by setting negative ranges for velocity.



----- Velocity Min: 1, Max: 64

*No MIDI Note Ons are transmitted when you strike keys with medium velocity or greater*



- - - - - Velocity Min: 64, Max: 127

*No MIDI Note Ons are transmitted unless you strike keys with velocity of medium or greater*



## The Pan/Volume (PAN/VOL) Page

By changing the parameters on this page, you can define how each zone sends MIDI volume and pan messages.

```
EditSetup*PAN/VOL <>Zone1/1
```

```
EntryVolume:None EntryPan:None
ExitVolume :None ExitPan :None
```

```
<more CH/PRE KEYVEL PANVOL BEND more>
```

Parameter	Range of Values	Default
Entry Volume	None, 0-127	None
Exit Volume	None, 0-127	None
Entry Pan	None, 0-127	None
Exit Pan	None, 0-127	None

### Entry Volume, Exit Volume

Entry Volume enables you to control the initial MIDI volume setting for each zone of the current setup. When you select a setup in Setup mode, the K2661 sends MIDI volume control (MIDI Controller 07) messages on each of the setup's MIDI channels, according to the value of the Entry Volume parameter. This sets the starting volume level for each zone, for any value other than **None**. Subsequent MIDI volume control signals sent to the setup's MIDI channels affect the volume normally.

When you exit the current setup, Exit Volume sends another MIDI Controller 07 message.

The setting of the Volume Lock parameter on the MIDI-mode CHANNELS page (see Chapter 10) determines whether this parameter has any effect.

### Entry Pan, Exit Pan

You can set entry and exit values for Pan as well. When you select a setup, the K2661 sends a MIDI pan control (MIDI Controller 10) message on each MIDI Channel in each zone; another MIDI pan control message is sent when you exit the setup. The Entry and Exit values for Pan are the same as those for Volume. There is also a Pan Lock parameter on the MIDI-mode CHANNELS page, which overrides the Setup Editor's Pan settings.

If you are trying to set the Pan and the program doesn't seem to be responding, you should check the Mode parameter on the OUTPUT page in the Program Editor. If it is set to **Fixed**, then the K2661 is ignoring the MIDI Pan message; setting Mode to **+MIDI** allows you to control the program's panning from the Setup Editor.

Most programs respond to pan messages on the next keystroke. This means that if you hold a note and change the pan, the current note will stay at its current position until you strike it again. However, a K2661 program that uses the PANNER algorithm will respond to real-time pan adjustments as well.

# The BEND Page

The parameters on the BEND page define the bend ranges for each of the three types of pitch bend messages the K2661 can respond to.

```
editSetup: BEND <>zone1/1
BendRange(ST): Prog AuxBend1Up : 12ST
BendRange(ct): Oct  AuxBend1Dwn: 12ST
                        AuxBend2Rng: 2ST

<more CH/PRG KEYVEL PANVOL BEND more>
```

Parameter	Range of Values	Default
Bend Range (semitones)	Prog, 0–60 semitones	Prog
Bend Range (cents)	±100 cents	0 cents
Aux Bend 1 Up	0–60 semitones	12 semitones
Aux Bend 1 Down	0–60 semitones	12 semitones
Aux Bend 2 Range	0–60 semitones	2 semitones

## Bend Range (Semitones) and Bend Range (Cents)

Bend Range (semitones) sends a bend range message to an internal program or a MIDI device, telling it how to define subsequent pitch bend messages. Some programs may behave strangely when you change the Bend Range value, because they use FUNs (see Chapter 17) or DSP Functions (see Chapter 16) to affect the pitch wheel. In that case, you should either set the value of Bend Range to **Prog**, or edit the program itself.

Bend Range (cents) lets you fine tune the value for Bend Range (semitones). **100 cents** equals one semitone, or one half-step; you can set this parameter anywhere between **±100 cents**.

Bend Range, in both semitones and cents, affects all controllers that are set to **BendUp** (in the default control setup, the PWHL parameter on the WHEEL page is assigned to **BendUp**). Physical controllers assigned to **BendDwn** are also affected by Bend Range, but they bend notes in the opposite direction from controllers assigned to **BendUp**. Any physical controller that uses the MIDI Control Source list can be assigned to **BendUp** or **BendDwn**. To keep things simple though, you will normally want to use **BendUp** as a Pitch Wheel destination and use **Aux Bend 1** and **Aux Bend 2** for other controllers, such as the Large Ribbon.

Keep in mind that not all MIDI devices respond to Bend Range messages. K2600 series instruments, the K2500, PC2, and the PC88 support these messages, but with many older MIDI instruments (like the K2000 and K1000), you must set bend ranges on the devices themselves.

Changing programs sends a Bend Range message with the current program’s values. So does pressing **Panic**, which is a quick way to reset your K2661 or MIDI slaves if you’ve used a controller to modulate the bend range.

## Aux Bend 1Up and Aux Bend 1 Down

Like Bend Range, Aux Bend 1 defines the range for Pitch Bend messages, but does so for those physical controllers assigned to **MIDI 21**. There are two parameters related to AuxBend 1: an upward value (Aux Bend 1 Up) and a downward value (Aux Bend 1 Down). This means that you can set different values for upward and downward pitch-shifting. For example, you could get the Pitch Wheel to give you both vibrato and whammy-bar effects for the guitar program in a setup. In the zone that contains the guitar program, set AuxBend1Up to **2ST** and AuxBend1Dwn to **-12ST** (on the BEND page), then assign the PWhl parameter on the WHEEL page to a value of **AuxBend 1**. Now moving the Pitch Wheel up gives you a whole tone of upward bend, while moving it down gives you a full octave of downward bend.

In most factory setups, **AuxBend 1** is the assignment for the Large Ribbon.

## Aux Bend 2 Range

The K2661 allows you to specify a third pitch bend range; this is called Aux Bend 2, and it defines the range for MIDI Controller 15 messages. For AuxBend 2, you can set only one range for both upward and downward pitch bending.

# Controllers

Controller editing is one of the strongest aspects of the K2661's usefulness as the main controller for a sophisticated MIDI studio. In this section, we'll talk about two different types of "controllers" as they apply to the K2661. One is the *physical* controllers: the wheels, buttons, pedals, etc. that you move with your fingers or feet. The other is *MIDI* Controllers, which are MIDI commands sent by the K2661. For our purposes, "MIDI Controllers" includes the complete set of Controllers defined by the MIDI Specification, as well as pitchbend, aftertouch, and a few other useful MIDI commands. To fend off confusion, we'll refer to the K2661's physical controllers with a lower-case c, and MIDI Controllers with an upper-case C.

Any *MIDI* Controller can be used as the assignment for any *physical* controller (and for multiple physical controllers, as well). Or in other words, any physical controller like the Mod Wheel can be programmed to send any MIDI control signal. In addition, each controller in each setup zone can be tweaked just like keyboard velocity (or any other Setup-mode parameter). Although controller editing on the K2661 can be somewhat complex, it can also be very rewarding.

To get an idea of the expressive capabilities of Setup mode, explore the factory setups that come with the K2661.

The K2661's physical controllers include the following:

- The eight sliders (A, B, C, D, E, F, G, H) in the Assignable Controllers section
- The two Continuous Control Pedal jacks (CCPedal 1 and CC Pedal 2)
- The Large Ribbon, which you can define as a one- or three-section controller
- The Pitch Wheel
- The Modulation Wheel (Mod Wheel)
- Keyboard aftertouch, or Mono Pressure (MPress)
- The Breath Controller jack (which is connected to Continuous Controller Pedal 2)
- The two Panel Switches, located above the Pitch and Mod Wheels
- The four Footswitch pedal jacks (1, 2, 3, and 4)

The following tables and illustration provide an overview of the physical controllers and their parameters.

## Continuous Controllers

Physical Controller	Setup Editor Page	Parameter:Values
Sliders A, B, C, D, E, F, G, H	SLID, SLID/2	Destination: MIDI Control Source list
		Scale: -300% to 300%
		Add: -128 to 127
		Curve: Linear, Expand, Compress, Crossfade, Bump, Reverse Linear, Reverse Expand, Reverse Compress, Reverse Crossfade, Reverse Bump
		Entry Value: None, 0 - 127
		Exit Value: None, 0 - 127
CPedals 1 & 2 / Breath	CPEDAL	(same as Sliders)
Large Ribbon	RIBBON	(same as Sliders)
Pitch Wheel and Mod Wheel	WHEEL	(same as Sliders)
MPressure	PRESS	(same as Sliders)

**Table 7-3 Continuous Controllers**

## Switch Controllers

Physical Controller	Setup Editor Page	Parameter:Values
Footswitches 1, 2, 3, and 4	FOOTSW	SwType: Toggle, Momentary, Note Toggle, Note Momentary
		Destination: MIDI Control Source list
		On Value: None, 0 - 127
		Off Value: None, 0 - 127
		Entry Value: None, Off, On
		Exit Value: None, Off, On
Panel Switches 1 and 2	SWITCH	(same as Footswitches)

**Table 7-4 Switch Controllers**

After you've selected which zone and which physical controller to work with, use the Destination parameter to choose what this controller will do. Some of the controllers have default settings that are preprogrammed in all of the factory setups (of course, you can change them). You can select from the entire list of numbered MIDI controllers, as well as other MIDI commands and some "special functions." To move through the list, you can use the Alpha wheel, or Intuitive Entry with any continuous controller, or call up the Controller's number with the numeric buttonpad.

## The MIDI Control Source List

Here are the available values for the Destination parameter for each controller, in scrolling order:

MIDI Controller Number	Corresponding Destination Name	Description
0	Off	Turns physical controller off
1	MWheel	Default Assignment for Mod Wheel
2	Breath	Default Assignment for CC Pedal 2
3	MIDI 03	
4	Foot	Default Assignment for CC Pedal 1
5	PortTim	Monophonic K2661 programs respond to this controller if portamento is turned on
6	Data	Almost all K2661 programs respond to this controller with DSP sweeps and effects
7	Volume	MIDI Volume
8	Balance	MIDI Balance
9	MIDI 09	

**Table 7-5 MIDI Control Source List**

## Setup Mode and the Setup Editor

### Controllers

MIDI Controller Number	Corresponding Destination Name	Description
10	Pan	MIDI Pan—programs which use the PANNER algorithm will respond to real-time pan adjustments; all other programs will respond on the next note start
11	Express	MIDI Expression—an attenuator for fading in and out. It scales between minimum (0) and the current value of Volume
12–14	MIDI 12–14	MIDI Controllers 12–14
15	Aux Bend 2	
16	Ctl A	
17	Ctl B	
18	Ctl C	
19	Ctl D	
20	MIDI 20	
21	Aux Bend 1	Default destination for Large Ribbon
22–31	MIDI 22–31	MIDI Controllers 22-31. Almost all K2661 programs respond to MIDI 22–29 in one way or another. In the default control setup (97 ControlSetup) and in most other factory setups, Sliders B–H send MIDI 22–28, and Panel Switch 2 sends MIDI 29.
<b>Special Function Controllers</b> When you enter numbers 32–46 on the numeric buttonpad, you select one of the K2661's Special Functions. Entering 47–63 sets a physical controller's Destination to <b>Off</b> . Note that these are not MIDI Controller numbers; they're the K2661's internal global control numbers		
32	MPress	Default destination for Press
33	BendUp	Default destination for Pitch Wheel
34	BendDwn	Physical controllers bend in opposite direction of BendUp
35	Tempo	MIDI Clock tempo
36	MuteZn	Mute current zone
37	KeyNum	Key Number—sends a MIDI note with a velocity defined by the last value for KeyVel
38	KeyVel	Key Velocity
39	TapTempo	Enables you to tap a switch controller (like FtSw1–4 or PSw1–2) to change the K2661's internal tempo setting
40	ArpOrder	Change the order of arpeggiated notes
41	ArpBeats	Change the number of arpeggiated notes per beat
42	ArpShift	Change the amount of arpeggiator shift
43	ArpLimit	Change the arpeggiator shift limit
44	ArpLmtOp	Change the arpeggiator limit option
45	ArpVel	Change the attack velocity of arpeggiated notes
46	ArpDur	Change the duration of arpeggiated notes
47–63	Off	Reserved for future use
64	Sustain	Default destination for Footswitch 1

**Table 7-5 MIDI Control Source List (Continued)**

MIDI Controller Number	Corresponding Destination Name	Description
65	PortSw	Monophonic K2661 programs respond to this controller if portamento is turned on
66	SostPd	Default destination for Footswitch 2—holds notes that are currently down, but not notes played subsequently
67	SoftPd	Default destination for Footswitch 3— lowers the volume by a preset amount and may soften the timbre as well
68	LegatoSw	Forces mono playback
69	FrezPd	Envelopes freeze at current state
70–79	MIDI 70–79	MIDI Controllers 70–79
80	Ctl E	
81	Ctl F	
82	Ctl G	
83	Ctl H	
84–90	MIDI 84–90	MIDI Controllers 84–90
91	FX Depth	Controls wet/dry mix of effects
92–95	MIDI 92–95	MIDI Controllers 92–95
96	DataInc	Data Increment
97	DataDec	Data Decrement
98–101	MIDI 98–101	MIDI Controllers 98–101
102	Play/Stp	Alternately starts and stops sequencer playback
103	Rec/Stop	Alternately starts and stops sequencer recording
104–115	MIDI 104–115	MIDI Controllers 104–115
116	ArpSw	Arpeggiator On/Off
117	ArpVel	Arpeggiator Velocity
118	Latch2	Arpeggiator Latch 2
119	ArpLatch	Arpeggiator Latch
120	Panic2	Sends All Notes Off message
121	MIDI 121	Resets all controllers
122	Off	Turns physical controller off
123	Panic	Sends All Notes Off message
124, 125	MIDI 124, 125	MIDI Controllers 124 and 125
126	Mono On	
127	Poly On	

Table 7-5 MIDI Control Source List (Continued)

## Continuous Controller Parameters

The continuous (physical) controllers are those that have a *range* of values: the two wheels, the ribbon, eight sliders, two Continuous Control pedals, and mono pressure (aftertouch). As the

table on page 7-20 shows, all of them use the same parameters. Each parameter's function is described below.

## Destination (Dest)

Use this parameter to select a destination from the MIDI Control Source list (see page 7-21).

## Scale

After you've selected a continuous physical controller, you can modify the controller's response similarly to the ways you can modify velocity response. Refer to the graphs beginning on page 7-12 for illustrations of the velocity scaling parameters.

Scale lets you amplify or diminish the action of the controller. Full scale is 100%. Higher values will make the controller more sensitive, and lower values will make it less so. Setting the scale to a negative number makes the controller action work in reverse. As with velocity, you can use a controller to crossfade between two zones by setting the scaling for one zone positive and the other negative. Maximum scale values are **+300%** and **-300%**.

## Offset (Add)

This adds or subtracts a constant to the controller, and at the same time sets minimum or maximum values (there's no need for separate Max and Min parameters). If the offset is **25**, the minimum value of the controller will be 25. If it is **-25** (and scale is **100%**) the first one-fifth of the controller's movement ( $25 / 127 = \text{about } 1 / 5$ ) will send value of 0, and the maximum value of the controller will be 102 ( $= 127 - 25$ ). As with velocity, Scale is a proportional change to the controller, while Offset is a linear change. The maximum values for Offset are **±127**.

## Curve (Curv)

This lets you taper the controller response. The default setting is **Linear**, which means that the response follows a straight line as you move the controller.

**Expand** produces a curve that is less steep than the linear curve at keystroke velocities below 64, and steeper than the linear curve at keystroke velocities above 64. In other words, when you're playing softly, you'll notice velocity differences less than with a linear curve, while when you're playing hard, you'll notice velocity differences more.

**Compress** produces a velocity curve that is the opposite of the expanded curve—that is, you'll notice velocity differences more when you're playing softly than when you're playing hard.

Crossfade (**Xfade**) is designed to be used in tandem with the Reverse Crossfade curve, enabling you to perform smooth crossfades between different programs.

**Bump** tapers velocity response to resemble a bell-curve, making its response greatest at medium strike velocity, and progressively weaker as strike velocity increases or diminishes. If you start playing softly, then progressively louder, the response will increase from 1 to 64; if you keep playing harder, the sound will get softer.

The next five curves are Reverse Linear (**rLin**), Reverse Expand (**rExp**), Reverse Compress (**rCmp**), Reverse Crossfade (**rXfd**), and Reverse Bump (**rBmp**). These taper controller response in reverse of the first five curves. For example, Reverse Linear's response is such that moving the pitch wheel forward decrease pitch, moving it back will increase pitch, and so on. As with velocity, the reverse curves offer you a quick way to achieve negative scaling of physical controller response.



To get an idea of how these curves affect controller response, refer to the Velocity Curve charts, which begin on page 7-12.

## Entry (Ent) and Exit Values

Entry value allows you to specify an initial value for a controller in a setup that will be sent whenever you select that setup. For example, if you want to make sure that all of the modulation in a zone is turned off when you select a setup, assign a physical controller to a destination of MIDI 01 (MWheel) and set Entry Value to 0.

Entry values ignore the current position of the physical controller when the setup is selected. In fact, if the physical controller is above or below the entry value when the setup is selected (which it often is), moving the controller will have no effect until it is past its entry value. In the modulation example above, moving the assigned controller won't turn on any modulation until it's pushed all the way *down*, and then up again.

An entry value of **None** is quite different from a value of 0. **None** means that there will be no initial controller command when the setup is selected, and any subsequent movement of the physical controller will be effective.

Exit Value tells the K2661 to send a value for that controller whenever you leave the setup, either by selecting another setup or by selecting a different mode altogether. It can be very useful when a controller is doing something to the sound, and you don't want that effect to continue after you leave the setup. For example, if you want to make sure a zone's pitch returns to normal whenever you leave a setup, you would set Exit Value to 64 for any controller whose Destination parameter is set to **BendUp**. Again, **None** means no command is sent.

## The SLIDER and SLID/2 Pages

You can assign each of the K2661's eight programmable sliders to a destination on each of the eight zones. Or, you can assign any combination of sliders to the same zone, allowing you tremendous flexibility. For example, you can assign Sliders A and B to modulate pitch and volume on Zone 1, then assign Slider C to control panning on Zones 2, 3, and 4.

The **SLIDER** button gives you access to Sliders A–D. Press the **SLID/2** button to program Sliders E–H.

Take a look at the parameters. Since all the continuous controllers work in a similar manner, you'll find these same parameters on the CPEDAL, RIBBON, WHEEL, and PRESS pages.

```

EditSetup:SLIDER                                <>Zone1/1

SlidA:  Dest:      Scale: Add: Curv: Ent:  Exit:
SlidA:  Data       100%  0    Lin   None None
SlidB:  MIDI22     100%  0    Lin   None None
SlidC:  MIDI23     100%  0    Lin   None None
SlidD:  MIDI24     100%  0    Lin   None None
<more> SLIDER SLID/2 FOOTSW CPEDAL <more>
  
```

```

EditSetup:SLID/2                                <>Zone1/1

Slide:  Dest:      Scale: Add: Curv: Ent:  Exit:
Slide:  MIDI25     100%  0    Lin   None None
Slide:  MIDI26     100%  0    Lin   None None
Slide:  MIDI27     100%  0    Lin   None None
Slide:  MIDI28     100%  0    Lin   None None
<more> SLIDER SLID/2 FOOTSW CPEDAL <more>
  
```

Parameter	Range of Values	Default
Destination (Slider A)	MIDI Control Source list	Data
Destination (Slider B)	MIDI Control Source list	MIDI 22
Destination (Slider C)	MIDI Control Source list	MIDI 23
Destination (Slider D)	MIDI Control Source list	MIDI 24
Destination (Slider E)	MIDI Control Source list	MIDI 25
Destination (Slider F)	MIDI Control Source list	MIDI 26
Destination (Slider G)	MIDI Control Source list	MIDI 27
Destination (Slider H)	MIDI Control Source list	MIDI 28
Scale	±300%	100%
Add	-128 to +127	0
Curve	Curve list (see text)	Lin
Entry Value	None, 0–127	None
Exit Value	None, 0–127	None



**NOTE:** The FOOTSW page, and all the switch-controller pages, are described following the section called Switch Controller Parameters, which begins on page 7-30.

## The Continuous Control Pedal (CPEDAL) Page

If you look at the back of the instrument, you will see that there are jacks for plugging in two CC (Continuous Control) pedals. There is also a jack labeled Breath Controller. This jack is wired in parallel with CC Pedal 2; in other words, you can send MIDI 02 Breath (or any other MIDI or global Controller) with both CC Pedal 2 and an breath controller.

```

editsetup:CPEDAL <>zone1/1

Dest:  Scale: Add: Curv: Ent: Exit:
CPed1: Foot 100% 0 Lin None None
CPed2: Breath 100% 0 Lin None None

<more SLIDER SLID/2 FOOTSW CPEDAL more>

```

Parameter	Range of Values	Default
Destination (CPed1)	MIDI Control Source list	Foot
Destination (CPed2)	MIDI Control Source list	Breath
Scale	±300%	100%
Add	-128 to +127	0
Curve	Curve list (see text)	Lin
Entry Value	None, 0–127	None
Exit Value	None, 0–127	None

## The RIBBON Page

The RIBBON page lets you define controller assignments for the K2661's ribbon controller. Each ribbon senses movement when you press on it and move your finger left or right; this creates numerous possibilities for controlling pitch, volume, panning, crossfades between zones, or any other uses you might imagine.

The Large Ribbon can be used as a single long controller, or it can be divided into three separate sections, each with its own controller assignments (this is done on the RIBCFG page). The two small arrows above the strip indicate the boundaries of the three sections. The large arrow above the ribbon points to the center of the ribbon, for when the ribbon is configured in one section. The K2500 and K2600 had a small ribbon physical controller as well. Small Ribbon parameters are still included, for compatibility with K2600 and K2500 setups.

To modify other ribbon parameters, go to the RIBCFG page, which is described on page 7-42.

The first of the following diagrams shows how the RIBBON page looks when the Large Ribbon is configured in one sections. The second shows the RIBBON page for a three-section Large Ribbon.

```

EditSetup*RIBBON          <>zone1/1
Dest:      Scale: Add: Curv: Ent:  Exit:
SmRib:  AuxBend2 100% 0   Lin  None  None
SmPrs:  MPress  100% 0   Lin  None  None
LgRib:  AuxBend1 100% 0   Lin  None  None

```

```

<more  RIBBON  WHEEL  SWITCH  PRESS  more>

```

```

EditSetup*RIBBON          <>zone1/1
Dest:      Scale: Add: Curv: Ent:  Exit:
SmRib:  AuxBend2 100% 0   Lin  None  None
SmPrs:  MPress  100% 0   Lin  None  None
Sect1:  AuxBend1 100% 0   Lin  None  None
Sect2:  Ct1D    100% 0   Lin  None  None
Sect3:  MIDI20  100% 0   Lin  None  None
<more  RIBBON  WHEEL  SWITCH  PRESS  more>

```

Parameter	Range of Values	Default
Destination (SmRib)	MIDI Control Source list	AuxBend2
Destination (SmPrs)	MIDI Control Source list	MPress
Destination (LgRib)	MIDI Control Source list	AuxBend1
Scale	±300%	100%
Add	-128 to +127	0
Curve	Curve list (see text)	Lin
Entry Value	None, 0–127	None
Exit Value	None, 0–127	None

## The WHEEL Page

The two wheels are typical of what is found on many keyboards. The left one is normally used for pitch bend and springs back to center, while the right wheel is normally used as a standard Mod Wheel.

```
EditSetup:WHEEL <>Zone1/1
```

```

      Test:   Scale: Add: Curv: Ent: Exit:
PWhl : BendUp 100% 0   Lin  None None
MWhl : MWheel 100% 0   Lin  None None

```

```
<more RIBBON WHEEL SWITCH PRESS more>
```

Parameter	Range of Values	Default
Destination (PWhl)	MIDI Control Source list	BendUp
Destination (MWhl)	MIDI Control Source list	MWheel
Scale	±300%	100%
Add	-128 to +127	0
Curve	Curve list	Lin
Entry Value	None, 0–127	None
Exit Value	None, 0–127	None

## The Pressure (PRESS) Page

The K2661 features mono pressure, commonly called aftertouch on other keyboards.

A word about pressure: Key Range in a zone does *not* define which notes will generate pressure in that zone. If pressure is enabled in a zone, playing with aftertouch *anywhere* on the keyboard will produce data. For example, if Zone 1's Key Range is C3-C5 and you play C2 and push down on the note, pressure messages will be sent from Zone 1. As with any other physical controller, however, you can disable pressure in any zone, or scale it or offset it differently in the various zones. It might help to think of pressure as an extra wheel—wheels operate in a zone regardless of Key Range, and so does pressure.

```
EditSetup:PRESS <>Zone1/1
```

```

      Test:   Scale: Add: Curv: Ent: Exit:
Press: MPress 100% 0   Lin  None None

```

```
<more RIBBON WHEEL SWITCH PRESS more>
```

Parameter	Range of Values	Default
Destination	MIDI Control Source list	MPress
Scale	±300%	100%
Add	-128 to +127	0
Curve	Curve list (see text)	Lin
Entry Value	None, 0–127	None
Exit Value	None, 0–127	None

## Switch Controller Parameters

Switch (physical) controllers have only two states: on and off. The K2661 switch controllers are:

- Panel Switches 1 and 2 (PSw1 and PSw2)
- Footswitch Pedals 1, 2, 3, and 4 (FtSw1, FtSw2, FtSw3, and FtSw4)

Note that buttons 1–8 above the sliders are dedicated to zone status and muting, as well as sequencer muting, and are *not* assignable controllers.

### Switch Type (SwType)

The parameters for Switch controllers are slightly different from those for continuous controllers. The first parameter is Switch Type (SwType). The choices available are **Momentary** in which a switch's action lasts only as long as you are pushing it, and **Toggle**, in which the switch's action lasts until you press it again. You can also assign the switch controller to send a note, using **Note Momentary** and **Note Toggle**.

Momentary mode is used for functions like sustain or portamento, while Toggle mode is used for functions such as arpeggiator on/off. The buttons show which mode they are in by the behavior of their lights: if a button is in Momentary mode, its light glows only as long as you are holding it, while if it is in Toggle mode, the light stays on until you press it again. Bear in mind that button assignments are independent per zone, and since there's just a single light per button, the light shows the state of the button only for the current zone. When you press the button, however, it executes its assignments for all zones that use that button.

### Destination

Destination determines what MIDI Controller or other message will be sent when the switch is on—that is, either pressed and held or toggled from the off position. The list of available controllers on page 7-21 is the same as for the continuous controllers, and can be accessed the same way.

If you set SwType to **Note Momentary** or **Note Toggle**, the values for Destination change to display MIDI note values. This can be pretty interesting if you want to do some unusual things with the switch controllers. For example, choose a drum program for the current zone, then set the destination of Footswitch 1 to **Note Momentary**. Now set the value of Destination to a key with a kick drum or a closed hi-hat sound, and impress your drummer friends. Or, you could set SwType to **Note Toggle** and use the Footswitch to start and stop a sampled groove, freeing up your hands to play a different program.

## On Value

On Value sets the value of the Controller when the switch is on. In the case of conventionally-switched functions, such as sustain, the On Value will be **127**. (For example, the default for Switch Pedal 1 (FtSw1) is Controller 64 — **Sustain** — with an On Value of **127**.) However, you might want to use a button or pedal as a “soft” switch, in which case you might set Destination to 7 (Volume) and On Value to **50**. Destination can also be set to **Off**, so that turning on the switch has no effect at all in this zone. This can be useful when you are using one switch for multiple functions in different zones.

If Switch Type is set to **Note Momentary** or **Note Toggle**, the On Value will define the velocity of the note message that gets sent.

## Off Value

Off Value is the value of the Controller when the switch is off. The default value is **0**. You might want to change this, as in the “soft switch” example above: in order to bring the zone up to full volume when you release the pedal, set Off Value to **127**.

If Switch Type is set to **Note Momentary** or **Note Toggle**, be sure to leave the Off value at **0**; this will send a note off message when you turn the switch off, preventing stuck notes from occurring.

## Entry (Ent) and Exit States

Entry State determines whether an initial setting for the switch will be sent when the setup is selected. There are three choices: **None** (no change), **Off** (the Off value), and **On** (the On value). With a Panel Switch button, if the Entry State is **On**, the button will light as soon as you select the setup.

Exit State similarly determines whether a setting for the switch will be sent when you leave the setup, either for another setup or for Program mode. The same three choices (**On**, **Off**, and **None**) are available. This is very useful for turning off sustains when changing setups.

# The Footswitch (FOOTSW) Page

On the back of the instrument, there are four jacks for Footswitch pedals.

All the foot switches in the default control setup are set to a switch type that doesn't generate a note, and consequently the range of values for their respective Destination parameters is the

MIDI Control Source list. If you set a footswitch's switch type to Note Momentary or Note Toggle, the range of values for its Destination parameter is C -1-G 9.

```

EditSetup:FOOTSW          <>Zone1/1

FtSw1: Moment Sustain 127 0  None None
FtSw2: Moment SostPd  127 0  None None
FtSw3: Moment SoftPd  127 0  None None
FtSw4: Moment TapTempo 127 0  Off  Off
<more SLIDER SLID/2 FOOTSW CPEDAL more>

```

Parameter	Range of Values	Default
Switch Type	Momentary, Toggle, Note Momentary, Note Toggle	FtSw1: Moment
		FtSw2: Moment
		FtSw3: Moment
		FtSw4: Moment
Destination	MIDI Control Source list when SwType is Momentary and Toggle); C -1 to G9 when SwType is Note Momentary or Note Toggle	FtSw1: Sustain
		FtSw2: SostPd
		FtSw3: SoftPd
		FtSw4: TapTempo
On	0-127	127
Off	0-127	0
Entry Value	None, 0-127	None
Exit Value	None, 0-127	None

The SWITCH Page

The K2661 keyboard offers two Panel switches, located above the pitch and mod wheels. Each of these functions exactly like the Footswitches; you can choose between momentary and toggle switches, or you can use each one to trigger a note.

```

EditSetup*SWITCH          <>Zone1/6

PSw1 : Toggle Off 127 0  Off Off
PSw2 : Toggle Off 100% 127 0  Off Off

<more RIBBON WHEEL SWITCH PRESS more>

```



Parameter	Range of Values	Default
Switch Type	Momentary, Toggle, Note Momentary, Note Toggle	PSw1: Toggle
		PSw2: Toggle
Destination	MIDI Control Source list or C -1 to G9	PSw1: ArpSw
		PSw2: MIDI 29
On	0-127	127
Off	0-127	0
Entry Value	None, 0-127	PSw1: None
		PSw2: Off
Exit Value	None, 0-127	Off

## The KDFX and FXMOD Pages

These are the same seven effects-control pages that we discussed in Chapter 6 (beginning on page 6-44). They enable you to define the studio and FXMods for all zones in the setup. As long as the FX Mode parameter is set to **Auto** or **Setup**, the values you set on these pages apply to all zones in the setup. If FX Mode is **Master**, all programs and setups use the studio specified on the Effects-mode page.

## The COMMON Page

The COMMON page determines what song, if any, becomes current when you select a setup, and how song playback is synchronized. It also sets the operational mode for the buttons above the programmable sliders.

```

EditSetup:COMMON      H11 Zones

```

```

Song  :0 None          Sync :Off
SngCtl:Setup
Mutes :Zone Mutes

```

```

<more  COMMON ARPEG RIBCFG more>

```

Parameter	Range of Values	Default
Song	List of existing songs	0 None
SngCtl	Song, Setup	Setup
Sync:	Off, On	Off
Mutes	Zone Mutes, KB3 Control	Zone Mutes

## Song

You can link a song to the current setup, using the Song parameter. Select any existing song (or **0 None** if you don't want a song associated with the setup). The song **1 New Song** is a blank template that contains no notes (unless you've saved **1 New Song** without renaming it).

Press **Edit** while the Song parameter is highlighted to use the Song Editor. See Chapter 12 for more information on the Song Editor's functions.

When the current setup has a song linked to it, the **Stop**, **Play/Pause**, or **Record** buttons enable you to start and stop both Recording and Playback of the current song while you're in Setup mode. For rack-mount model owners, pressing the **Up/Down** cursor buttons toggles between Play and Pause, without resetting the Locate point. Pressing the **Left/Right** buttons toggles between Start and Stop, resetting Locate to 1:1. These double-button presses work in any mode.

If you are planning to use a particular song with a setup, there are some important things to keep in mind. First, every track of your song and every setup zone should be on different MIDI channels. Secondly, set the FX Mode to **Auto** and the FX Channel to **Current** on the Effects-mode page; this way, when you play a song from within Setup mode, you'll hear the effect assigned to that song.

Once you've chosen a song on the COMMON page (and worked out any MIDI conflicts), you can use Setup mode to do some pretty cool things. If you want to start the song with a Footswitch, for example, you can assign the Destination parameter of any Footswitch to **Play/Stp** (MIDI 102). Because of the nature of this special function controller, set SwType to **Momentary** rather than **Toggle**.

Here's another great feature: by linking songs with setups, you can select songs from Quick Access mode. By assigning different songs to different setups, and then assigning those setups in one Quick Access bank, you can select one of up to ten songs with a single button press.

## Song Control

This parameter has two possible values: Song or Setup. Song Control works in conjunction with the Song parameter on the COMMON page. If a Song is called up from within the Setup, and the Song Control parameter is set to "Song", the song plays back exactly as recorded. If Song Control is set to "Setup", all events except for Notes are filtered out when the Song is played.

This parameter has been added for use with the preset Groove Setups (1–30) in the K2661; the Groove Setups make extensive use of song files for drum patterns. Song Control lets you choose either the Song or the Setup to be in control of selecting the programs and other controller info.

## Sync

Setting this parameter to a value of **On** enables you to do two things during song playback:

- When you change the current song, it finishes before the newly-selected song starts
- When using Note Ons (keystrokes or MIDI input) to retrigger and transpose steps in an arrangement, you can trigger a Note On at any time during the current bar, to retrigger the step at the beginning of the next bar

When Sync is **Off**, changing a song during playback immediately starts the new song, and Note Ons immediately retrigger or transpose.

## Mutes

The Mutes parameter gives you manual control over the behavior of the buttons above the programmable sliders. There's really only one case in which you'd need to worry about this parameter: when you have a setup that contains both VAST programs and KB3 programs, and you want the buttons to control KB3 features.

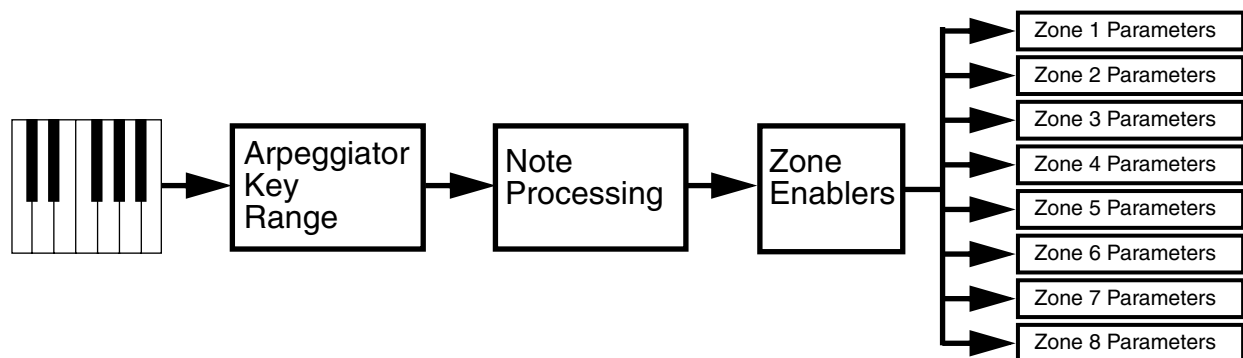
By default, the K2661 uses the **Mute** buttons (the buttons above the sliders on keyboard models) to control the muting and unmuting of zones—which means you don't have real-time control over any KB3 features. Change the Mutes parameter to KB3 Control to use the **Mute** buttons as real-time KB3 controllers. The light grey labeling above the buttons describes their functions.

## The Arpeggiator (ARPEG) Page

The Arpeggiator takes input from the K2661 keyboard (or via MIDI) and turns it into a constant rhythmic pattern. You can control the speed and nature of the pattern in real time. The Arpeggiator resembles what were called “sequencers” on old analog synths—playing a finite series of notes repeatedly, with changes in the series controlled by the notes you play. The Arpeggiator can affect both the K2661 and external MIDI instruments. The notes produced by the Arpeggiator in a given zone go to all of that zone’s destinations: local, MIDI, or both.

The concept behind the K2661’s Arpeggiator is fairly simple, although the options are extensive. You might think of it as a “note processor,” generating complex output from relatively modest input. You can select any number of notes for the input, and tell the Arpeggiator to recognize and remember them. This is called “latching” the notes. The Arpeggiator then processes them by playing them repeatedly, and/or transposing them up and down the keyboard. You have control over several processing parameters: tempo, velocity, order, duration, transposition, orchestration, and whether the intervals between notes are filled chromatically. You can also tell the Arpeggiator how to deal with new information coming from the keyboard. The settings you define on the ARPEG page apply to all zones for which arpeggiation is activated; you can program each zone individually to respond to or ignore the setup’s arpeggiation values.

Figure 7-1 shows the processing order the K2661 uses to create arpeggios from your input:



**Figure 7-1** Arpeggiator Processing Sequence

```

EditSetup*ARPEG All Zones
Active:Off Sync:Off NoteShift :0ST
LoKey:C 0 HiKey:G#8 ShiftLimit:0
Latch:Auto LimitOption:Reset
Order:Played Glissando :Off
Beats:1/16 Velocity :Played
Tempo:120BPM Duration :75%
<more COMMON ARPEG RIBCFC more>

```

Parameter	Range of Values	Default
Active	On/Off	Off
Sync	On/Off	Off
Low Key	C -1 to G9	C0
High Key	C -1 to G9	C#8
Latch	Latch list (Auto, Keys, etc.)	Auto
Order	Order list (Played, Up, etc.)	Played
Beats	1/1 to 1/384	1/16
Tempo	1 to 255 beats/minute	120 beats/minute
Note Shift	±88 semitones	0 semitones
Shift Limit	0 to 88	0
Limit Option	Limit List	Reset
Glissando	On/Off	Off
Velocity	List	Played
Duration	1% to 100%	75%

Active

The first parameter on the Arpeggiator menu is Active — is the Arpeggiator on or off for this setup? This can be switched from here, or for convenience in live performance, it can be switched using MIDI Controller number 116. This can either be assigned as the destination of a K2661 physical controller, or it can come from an external MIDI source.

Turning Active on affects zones whose ZoneArpeg values are also set to **On**. By setting the ZoneArpeg parameter (on the CH/PRG page) to **Off** or **On** in the individual zones of a setup, you can choose which zones will be controlled by the Arpeggiator when it is on.

Sync

When Sync is set to **On**, the arpeggiator plays latched notes exactly on the next scheduled beat, which is a function of the Beats parameter and the Song-mode clock. For example, if Beats is 1/8, the arpeggiator plays eighth notes, exactly on each beat and half beat of the current song. When Sync is set to **Off**, latched notes still play at the same *tempo* as the song, but they don't play exactly on the beat.

Low Key (LoKey) and High Key (HiKey)

The Arpeggiator processes notes within the range of these parameters. Notes outside the specified range play normally, and do not become part of the arpeggiation sequence. Set the LoKey and HiKey parameters using the data entry wheel or buttons.

## Latch

Latch determines how the Arpeggiator responds to notes when they are triggered.

**Keys** means that the Arpeggiator plays only while you are holding one or more keys down (or note triggers on). As you play different notes, they get added to the Arpeggiator, and as you release notes, they get taken out. If you play notes faster than the Arpeggiator's current tempo, each subsequent note will be added to the arpeggiation at the next division of a beat. This can cause a lag between the time you play the note and the time you hear it in the arpeggiation.

In the next three modes, the Arpeggiator latches notes only when MIDI Controller 119 (ArpLatch) sends a value of On (64 or higher). An easy way to experiment with these modes is to assign the Mod Wheel to send MIDI 119 (go to the WHEEL page, and set the value of MWhl to **ArpLatch** by pressing **1, 1, 9, Enter** on the alphanumeric buttonpad).

In **Overplay** mode, the Arpeggiator latches any notes that are being held when ArpLatch goes on, and continues playing them, even after you let them go, until ArpLatch off. Any notes that you play after ArpLatch is already on do not get arpeggiated, even if they're in the arpeggiation range.

**Arpeggiation** is similar: any notes held when ArpLatch goes on are latched and arpeggiated, and keep going until ArpLatch goes off. Any notes you play outside the arpeggiation range play normally. Notes that you play inside the arpeggiation range do not play normally, but if you hold them on, they become part of the arpeggiation. They drop out of the arpeggiation as soon as you release them.

Like Overplay and Arpeggiation, **Add** means that all notes being held when ArpLatch goes on get latched, and keep playing until ArpLatch goes off (even if you've released the notes). Any notes you play after ArpLatch is already on also get latched.

**Auto** is independent of ArpLatch; every note you play is automatically latched, and the Arpeggiator runs as long as you hold at least one arpeggiated note. As long as you keep holding on at least one note (it doesn't have to be the same note the whole time), every note you play in the arpeggiation range gets latched.

**Pedals** is sort of a combination of Keys, Add, and Overplay modes. It relies on both ArpLatch (MIDI 119) and Latch2 (MIDI 118). If neither latch controller is on, notes will arpeggiate only while you are holding down keys (similar to Keys mode). If you activate Controller 119, the keys currently held down will latch, and any additional keys played while Controller 119 is on will also latch (similar to Add mode). When Controller 119 is off, any keys that are not currently held down will be removed from the arpeggiation. If you activate Controller 118, keys currently held down will latch, and any additional keys played while Controller 118 is on will play normally (similar to Overplay mode). This mode is called Pedals mode because you might want to assign Switch Pedal 1 to **Controller 119** (Latch 1) and Switch Pedal 2 to **Controller 118** (Latch 2) to make the pedals function similarly to sustain and sostenuto pedals.

**Autohold** is similar to Auto. Holding at least one arpeggiated note on and playing other notes latches those notes. Unlike in Auto mode, if you stop holding at least one arpeggiated note on, the arpeggiation continues playing (although you can't latch any more notes). In this case, if you strike another key within the setup's arpeggiation range, you start a new arpeggiation sequence.

**Autohold** is useful for arpeggiating chords: when you play a chord, it gets latched, and continues arpeggiating after you release the chord. When you play another chord, the previous chord gets unlatched, and the new one gets latched.

You can use the **Panic** soft button or the **Stop** button to stop arpeggiation at any time.

## Order

This parameter determines the order in which the K2661 plays arpeggiated notes. **Played** causes them to play back in the chronological order in which you played and latched them. **Up** means that notes play in ascending pitch order, regardless of their chronological order. **Down** means descending pitch order. **Up/Down** causes notes to play from lowest pitch to highest, then from highest pitch to lowest, repeating the cycle until you stop the arpeggiation. The notes at the very top and very bottom only play once. **Up/Down Repeat** is similar to **Up/Down**, except that the notes at the top and bottom play *twice* (repeat) when the Arpeggiator reverses direction.

**Random** plays the currently-latched notes in completely random order. **Shuffle** plays them at random, but keeps track of the notes so that no note repeats until all of the others have played. **Walk** is a “random walk” order: each successive note is either the next or previous note (in chronological order). For example, suppose you’ve latched four notes—G 4, B 4, D 5, and F 5—in that order. The first note the Arpeggiator plays is the G 4. The second note will be either B 4 (the next note chronologically), or F 5 (the “previous” note chronologically—that is, the last latched note). If the second note is B 4, the third note will be either D 5 or G 4. If the second note is F 5, the third note will be either G 4 or D 5.

**Simultaneous** means that each note you play repeats in time with the Tempo value, sort of like a digital delay with no decay. If you play a C and hold it while you play an E and a G, the Arpeggiator will play all three notes at the same time and at the same tempo. This is the only Order setting that allows two or more notes to sound simultaneously as intervals or chords. Note that **Simultaneous** works well with Note Shift and Shift limit.

## Beats

The Beats parameter sets the number of notes per beat. The tempo is based on quarter notes. Therefore, if you set it to  $1/4$ , you will get one note per beat of the clock. At  $1/16$ , you will get 4 notes per beat. You can go all the way to 96 notes per beat ( $1/384$ ), but at most tempos, divisions smaller than  $1/64$  will sound pretty much the same.

## Tempo

This sets the tempo, in beats per minute (bpm), at which the Arpeggiator plays when activated. Tempo is tied to the K2661’s internal MIDI clock, as is song playback. Consequently, if a song is playing when you activate the Arpeggiator, the song’s tempo defines the Arpeggiator’s tempo, overriding the Arpeggiator’s settings. Inversely, if you activate the Arpeggiator, then start a song, the Arpeggiator’s Tempo setting defines the song’s tempo.

You can change Tempo in real time by changing the assignment of any of the K2661’s physical controllers in the setup to **Tempo**. The range is **1** to **255 bpm**.

## Note Shift

You can tell the Arpeggiator to transpose all of the currently-latched notes each time it plays through them. Note Shift determines how much transposition will occur for each cycle of notes. For example, if you have latched C4 and F4, and you assign a Note Shift of 2, the Arpeggiator will play C4, F4, D4, G4, E4, A4, and so on until it reaches the Shift Limit. The values can range from -88 to 88, with 0 (the default) being no transposition.

## Shift Limit

Shift Limit determines how far up or down the Arpeggiator shifts from the original note. The minimum value is 0, and the maximum is 88. When the Arpeggiator reaches the limit, the Arpeggiator responds according to the setting for the Limit Option parameter.

## Limit Option

This parameter determines what the Arpeggiator does when it has shifted the currently latched notes up (or down) to the shift limit. **Stop** causes the Arpeggiator to stop when it reaches the shift limit. **Reset** causes the Arpeggiator to return to its original pitch and repeat the latched cycle of notes, transposing each cycle according to the settings for Note Shift and Shift Limit. If the limit allows the notes to go out of MIDI range (for example, if you set Shift to 12, set the limit to 80, and play C4), then those “ghost” notes don’t sound, but they take up rhythmic space: the Arpeggiator waits for the cycle to play itself out before starting over.

**Unipolar** means that after playing up to the shift limit, the Arpeggiator begins shifting notes in the opposite direction, until it reaches the original pitch, where it reverses again. To determine the next note when it reaches the shift limit, the Arpeggiator calculates the interval between the shift limit and what the next note would be if the shift limit weren’t there. It then plays the note that is the calculated interval lower than the last note before the shift limit. The same thing happens in reverse when the arpeggiated notes get back down to the original pitch. The following table makes this easier to visualize by showing the result of arpeggiating one note (C4) in Unipolar mode, with Note Shift set to 3 ST and various values for Shift Limit.

Shift Limit	Resulting Arpeggiation (When LimitOption is Unipolar)			Comment
	Up	Down	Up	
6 ST (F#4)	C4, D#4, F#4,	D#4, C4	D#4, ...	Same notes play in both directions when Shift Limit is a multiple of Note Shift
7 ST (G4)	C4, D#4, F#4,	E4, C#4,	D#4, ...	Last upward note before shift limit is F#4, next upward note would be A4, which is 2 ST from shift limit (G4); therefore first downward note is E4 (2 ST below last upward note)
8 ST (G#4)	C4, D#4, F#4,	F4, D4,	D#4, ...	A4 is 1 ST from shift limit, therefore first downward note is F4 (1 ST lower than last upward note)
9 ST (A4)	C4, D#4, F#4, A4	F#4, D#4, C4,	D#4, ...	All symmetrical again; now A4 is within shift limit
10 ST (A#4)	C4, D#4, F#4, A4,	G4, E4, C#4,	D#4, ...	Next upward note would be C5, which is 2 ST from shift limit
11 ST (B4)	C4, D#4, F#4, A4,	G#4, F4, D4,	D#4, ...	C5 is 1 ST from shift limit
12 ST (C5)	C4, D#4, F#4, A4, C5,	A4, F#4, D#4, C4,	D#4, ...	Symmetrical again, including C5

**Bipolar** starts out the same way as **Unipolar**, but during downward note shifting, it continues past the original pitch until it hits the shift limit in the *opposite* direction, where it reverses again.

**Flt Reset** adds a bit of apparent randomness to the process. **Flt** stands for “Float,” and it means that when the Arpeggiator reaches the shift limit, it resets—but not to its original pitch as with plain Reset. Like Unipolar and Bipolar, it looks at the first note that would exceed the shift limit, and calculates the interval between that note and the shift limit. It then restarts the cycle of latched notes, transposing the entire cycle by the interval it just calculated, then shifting each subsequent cycle by the value of Note Shift, until it reaches the shift limit again.

Here’s a very simple example. Suppose that the only note in the Arpeggiator cycle is C4, Note Shift is 4 (a third), and Shift Limit is 7 (so notes won’t get shifted above G4). The Arpeggiator plays C4, then E4. The next note should be G#4, but that’s above the shift limit—so the K2661 calculates the difference between that G#4 and the shift limit (G4): one semitone. It adds that

difference to the original starting note (C4) and plays that note next—C<sup>#</sup>4. The next note (F4) is within the shift limit, but the next note (A4) isn't, so it gets translated into D4—and so on.

**Flt Uni** uses the same concept and applies it to Unipolar mode: when the Arpeggiator reaches the shift limit, it calculates the difference between the next note and the limit, and transposes the next cycle of notes down by that interval, then shifts each subsequent cycle down until it reaches the original pitch. **Flt Bipl** is similar to **Flt Uni**, but the downward shift limit isn't the original pitch, it's the negative of the Shift Limit value.

The Arpeggiator can be a lot of fun, even if you don't always understand exactly what it's doing. Keep in mind that the stranger the algorithm you set up, the more unlikely the notes will stay close to one key, so if you want to create something that's going to sound at all diatonic, keep it simple.

## Glissando

When the Glissando parameter is **On**, the Arpeggiator chromatically fills between latched notes. When Glissando is on, the Arpeggiator ignores the Note Shift, Shift Limit, and Limit Option parameters.

You must latch at least two notes to get a result. When Glissando is on, all notes played in the arpeggiation range get latched, although you won't necessarily get meaningful results from all latched notes. In general, try to get each subsequent note you latch to be a change in direction. For example, try latching the following sequence of notes: C4, C5, G4, G5, C5, C6, G4, G5. The "glissando" changes direction around each change in direction of the latched notes.

## Velocity

Velocity sets the attack velocity of the played notes. **Played** means each note repeats with the same velocity you played it at. **Last** means all notes play at the velocity of the most-recently played note. **Pressure** means the velocities are controlled by keyboard pressure: as you push down on any key, the velocities get higher, and as you ease up they get lower. **Ctrl 117** means the velocity is controlled by MIDI Controller number 117, which can be assigned as the destination of any K2661 physical controller or can come from an external MIDI source.

There's another element that affects arpeggiator velocity: **ArpVel**, whose global control number is 45 (not MIDI 45). Input from any physical controller assigned to send **ArpVel** (or any entry value for a controller assigned to send **ArpVel**) overrides the programmed value of the Velocity parameter, disabling it until you select a setup (or in Program mode, until you select a control setup on the MIDI-mode TRANSMIT page). The override occurs even if the Velocity parameter is set to **Ctrl117**.

There is also a range of fixed values, from **Vel 1** to **Vel 127**. Selecting one of these determines a fixed velocity setting, which you can use to set a constant volume for the arpeggiated portion of all zones. The value you choose from this range defines the velocity level for the Arpeggiator, and will take effect no matter how hard you strike a key.

## Duration

Duration determines how long each arpeggiated note plays. 100% means that a note sustains until the next one sounds—very legato. 50% means that the note fills half the space between itself and the next note. The lowest value is 1%—*staccatissimo*. This parameter has no effect on percussion sounds or other sounds whose duration is fixed.



## Real-time Control of Arpeggiator Parameters

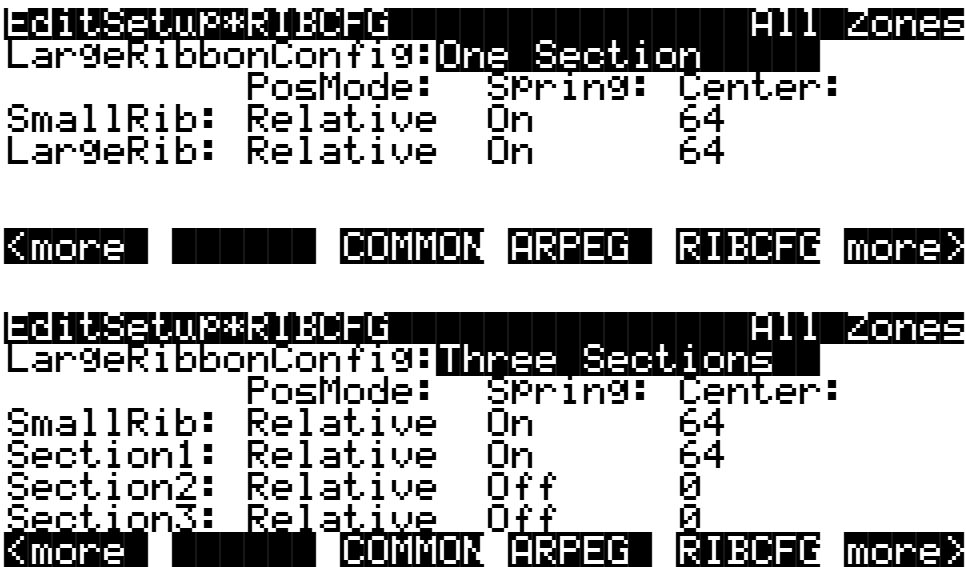
You can have real-time control over several arpeggiator parameters, by assigning physical controllers to special arpeggiator Controller Destinations. Any input (or entry value) from a physical controller assigned to an arpeggiator Controller Destination overrides the programmed values for the parameters. The override remains in effect until you select a setup (or in Program mode, until you select a control setup on the MIDI-mode TRANSMIT page).

There are countless possibilities for configuring physical controllers for real-time control over arpeggiator parameters. The following table covers just one possible configuration: Sliders A–G controlling each of the seven parameters that allow real-time control. We’ve included some suggested settings for the slider parameters, to make each slider at least reasonably useful for arpeggiator control. These settings are based on an arpeggiator range from C 6 to C 8 (as set with the LoKey and HiKey parameters on the ARPEG page).

Slider	Assignment	Suggested Slider Settings
A	40 ArpOrder	
B	41 ArpBeats	Scale = <b>10%</b> ; Add = <b>5</b>
C	42 ArpShift	Scale = <b>7%</b>
D	43 ArpLimit	Scale = <b>28%</b>
E	44 ArpLmtOp	
F	45 ArpVel	
G	46 ArpDur	

# The Ribbon Configuration (RIBCFG) Page

Once you’ve selected a destination for the Ribbon, you can use the parameters on the RIBCFG page to define how the ribbon responds to finger position.



Parameter	Range of Values	Default
Large Ribbon Configuration	One Section/Three Sections	One Section
Position Mode	Relative/Absolute	Relative
Spring	On/Off	On
Center	0-127	64

## Large Ribbon Configuration

You can use the Large Ribbon as one controller, or divide it up into three smaller sections, each with its own controller assignments. Choose a value of **One Section** or **Three Sections** for the LargeRibbonConfig parameter.

Notice the three white arrows directly above the Large Ribbon. The largest arrow (in the middle) points to the Large Ribbon’s center. The two smaller arrows on either side are the section boundaries for three-section configurations.

## Position Mode (PosMode)

When you touch the ribbon, the K2661 responds in one of two ways, depending on the setting of the PosMode parameter. **Relative** means that wherever you touch the ribbon becomes the “zero point” for whatever the ribbon is controlling; you won’t notice any change in the sound until you slide your finger. Relative mode tends to be the most natural for performance—just wiggle your finger anywhere on the Ribbon to get vibrato in many factory programs and setups. You get the same effect no matter where you do the wiggling.

**Absolute** means that the zero point for the ribbon is always at exactly the same physical location on the ribbon. By default this is the center of the ribbon, although you can use the Center

parameter to put the zero point elsewhere. In Absolute mode, just touching the ribbon affects the sound (unless you touch it at exactly the zero point). Every movement you make along the ribbon sends control values based on how far you are from the zero point.

## Spring

When Spring is **On**, the Controller to which you've assigned the ribbon "springs" back to its zero point automatically when you lift your finger off the ribbon. This is generally the behavior you want.

When Spring is **Off**, the Controller to which you've assigned the ribbon sticks at its current value when you lift your finger off the ribbon. While this can be useful, you should be careful with it. When Spring is **Off**, if the ribbon is doing something when you exit the setup, it'll continue doing that when you return to the setup. This can be good or bad. If you want to set Spring to **Off** in a setup, but want to be sure of the initial sound of the setup, go to the RIBBON page in the setup, and for each ribbon for which you've set Spring to **Off**, set the entry value (Ent) parameter to a value of 0 (or whatever you like).

## Center

This defines the zero point for each section on the ribbon—the point at which the Controller to which the ribbon is assigned has no effect on the sound. You can choose any point between 0 and 127. A value of 64 puts the zero point in the physical center of the ribbon. Values of 0 and 127 place the zero point at the section's extreme left and extreme right. Note that setting the Spring parameter to **Off** disables the Center parameter for that ribbon.

The value you choose for this parameter can have considerable effect. In many factory setups, for example, the Ribbon affects pitch. In these setups, changing the value of the Center parameter would transpose the setup.

# The Utility Soft Buttons

In addition to the Setup Editor's pages, there are basic library and editing soft buttons. Their functions are described below.

## Name

This enables you to rename the current setup. Use any data entry method to do this, including the letters on the alphanumeric buttonpad.

## Save

Pressing **Save** calls up the standard Save Dialog. Simultaneously pressing the **Plus** and **Minus** data entry buttons toggles between saving the setup to the first available empty location, or replacing a currently existing setup.

## Delete

This erases a setup from RAM, freeing up space to store setups in other locations. (You can check the free memory in the K2661 at any time, on the top line of the Master-mode page.) Press **Delete**, and use a data entry method to choose which setup you want deleted. Press **Delete** again, and an "Are You Sure?" message will appear (unless you have set a value of **No** for the Confirm parameter on the Master-mode page). Press **Yes** to delete the setup, or **No** to cancel.

As with programs, setups can be saved to and deleted from RAM only. The names of all setups in RAM have an asterisk (\*) next to them. If you try delete a setup from ROM, the K2661 will ignore the delete command, and the setup will remain in memory.

## Dump

This sends a MIDI System Exclusive dump of the current setup's settings. There is also a function for dumping *all* setups. To dump all setups, go to Master mode. Select the OBJECT page, and press the **Dump** soft button to select which objects to dump via SysEx.

For more on Object Utilities, see Chapter 11. For more about System Exclusive messages, see the *Musician's Reference*.

## New Zone (NewZn)

Press **NewZn** to create a new zone with default parameters. The K2661 imports this zone from Zone 1 of **99 Default Setup**. If there are parameters or entire pages you use often, you can create your own Default Setup and save it to location 99; pressing **NewZn** will then import zones from your custom Default Setup.

## Duplicate Zone (DupZn)

This adds a new zone with the same parameters as the current zone.

## Import Zone (ImpZn)

You can import, or "bring in," any zone from any setup in memory. Press **ImpZn**, and use any data entry method to choose a setup to import from. Then use the **Chan/Bank** buttons to select one of that setup's zones. As you do this, you can audition each zone to hear how it will sound in the current setup. Now press **Import**, and the zone you selected will be added to the current setup.

Note: If you are using all eight zones in a setup and you try to add, duplicate, or import a zone, a "No More Zones" message appears. You must delete an existing zone before you can add, duplicate, or import any new ones.

## Delete Zone (DelZn)

This deletes the current zone from the setup. Use **DelZn** to free up zones so you can add or import new ones.

## Copy and Paste

These functions are handy if you've spent some time tweaking a page or line of parameters, and you want to copy them quickly to other zones. Start by highlighting a parameter's value and pressing the **Copy** soft button. You can copy more than one value. Then use the **Chan/Bank** buttons to select the zone to which you want to copy those values. Press the **Paste** soft button, and all the values you copied replace the corresponding values in the current zone.

You can also paste the value into any zone of any other setup. Just copy a parameter's value, press **Exit** to leave the current setup, and select a different setup. Then press **Edit**, select the same page or physical controller type as in the last setup, and paste the new value. The values you copy each remain in one of nine paste buffers, so you can keep pasting these values to different zones in any setup. When you press **Copy** on a new page or line of parameters, you replace the contents of the current paste buffer with the new values.

The table below shows where you can paste copied parameters. Note that the first seven paste buffers each store an entire page, while the last two each store only one line of parameters.

Copied Parameters	Available Destinations in Any Setup Zone
CH/PROG page	Corresponding page
KEY/VEL page	Corresponding page
PAN/VOL page	Corresponding page
BEND page	Corresponding page
COMMON page	Corresponding page
ARPEG page	Corresponding page
RIBCFG page	Corresponding page
Parameters for one continuous controller (on SLIDER, SLID/2, CPEDAL, RIBBON, WHEEL, or PRESS pages)	Assignment for any continuous controller
Parameters for one switch controller (on FOOTSW and SWITCH pages)	Assignment for any switch controller

## Clear

Pressing **Clear** resets the selected page or line of parameters to its default value, as defined in **99 Default Setup**. You can edit the Default Setup, and alter the values that are called up for each page when you press **Clear**. The only exceptions to this are on continuous and switch controller assignment pages, where pressing **Clear** resets the current line to a factory-defined set of values.

You can't undo **Clear** without exiting the setup entirely, so be sure you really want to clear the selected parameters.

## Set Range (SetRng)

As we mentioned earlier in this chapter, **SetRng** gives you a quick way to set the values for the LoKey and HiKey parameters in the currently selected zone. Press this button, and the K2661 prompts you to strike the low and high keys. When you've done so, the Setup-editor page returns, and the notes you triggered are the new values for the LoKey and HiKey parameters. Notice that the higher of the two notes you played is the HiKey value, regardless of the order in which you triggered the two notes. The ranges you set are now represented by lines in the box at the left on the Setup-mode page.

# Editing Hints and Suggestions

## Do These Parameters Always Mean Something?

Be careful not to set up parameters that do crazy things. When in doubt, leave things off, or at the factory default settings. On some of the higher-numbered MIDI controllers and Special Function controllers, the **On** and/or **Off** values don't have any meaning, because the controllers have very limited functionality. The following list shows these. **>=64** means greater than or

equal to 64, and <64 means less than 64. >0 means you can use any value that's greater than zero. x means you can use any value.

MIDI Controller Number	MIDI Controller	On Value	Off Value
102	Play/Stop	>=64	<64
103	Record/Stop	>=64	<64
120	All Sound Off	>=64	<64
121	Reset All Controllers	>=64	<64
122	Local Control Off	>=64	<64
123	All Notes Off	>=64	<64
124	Omni mode Off	>=64	<64
125	Omni mode On	>=64	<64
126	Mono mode On (the On value, in some synths, specifies the number of MIDI channels to respond to, with one voice on each)	>=64	<64
127	127 Poly mode On	>0	x

**Table 7-6 Little-used MIDI Control Sources**

## Do I Need All Those Pedals?

It's important to realize that you don't have to have an actual pedal plugged into every Pedal and Switch Pedal jack in order to take advantage of all of them. Entry and exit values can be assigned to a zone and to a physical controller's destination even if the controller isn't there, and they will behave as if the controller were in fact plugged in. So if you want a specific group of Controller commands to be sent out whenever you choose a setup, and you know they're not going to change while you're playing, you can assign them as the destinations of physical controllers that aren't in use.

## Multiple Controllers

Sometimes you'll want to assign the same Controller number as the destination of more than one physical controller. For example, you might want to be able to control modulation depth both with Wheel 2 and with pressure. The K2661 has no restrictions on assigning multiple physical controllers to a particular MIDI Controller.

## Editing Programs

Remember, except for control-setup parameters, everything you define in a setup affects programs *only while you are in Setup mode*. To edit the programs themselves while you are in the Setup Editor, go to the CH/PRG page and press **Edit**. This gives you access to the other editors nested within the Program Editor as well. When you have finished editing your program, press **Exit** to save it and return to the Setup Editor.

## Chapter 8

# Quick Access Mode and the Quick Access Editor

In Quick Access mode, you can select programs or setups with a single press of an alphanumeric button (or with the other data entry methods). For example, in the illustration below, you would simply press 5 on the alphanumeric pad to choose **FM Harmonica**. Notice that your selection becomes highlighted in the list, as well as appearing on the line just above the soft-button labels.

```
QuickAccessMode  <>Bank:1 For Show 2
Stage Piano      Marimba      Voice+String9
Honky-Tonk       FM Harmonica  Eat
Memorymoog 4    Jazz Trio 2   Fretless Bas
Velveteen

Xpose:0ST      FM Harmonica      Chan:1
Octav- Octav+  Panic Sample Chan-  Chan+
```

Using Quick Access mode involves selecting Quick Access banks from the list of factory preset or user-programmed banks. You can use the bank selection shortcut to do this: press the +/- or **Clear** button on the alphanumeric pad, and you'll be prompted to enter a bank number. Type the desired number on the alphanumeric pad, then press **Enter**. The bank is selected, and you return to the Quick Access-mode page. Or use the **Chan/Bank** buttons to scroll through the QA banks.

Each bank contains ten memory slots, or entries, where you can store programs or setups in any combination. Any program or setup in the currently selected bank can be selected with the numeric buttons **0** through **9**.

The ROM (factory preset) QA banks are organized into useful groupings of sounds that we think you'll find convenient.

You can store 20 Quick Access banks in each memory bank (except the Zeros bank, which can store 75). See *Storing Objects in the Memory Banks* on page 13-34 for a breakdown of the Quick Access-bank IDs that belong in each memory bank. Press both **Chan/Bank** buttons simultaneously to quickly move between memory banks.

The MIDI Program Change commands that the K2661 sends when in Quick Access mode can differ from those in Program or Setup mode. This depends on the setting you have for the PChgType parameter in MIDI mode. If the setting is **Extended** or **Kurzweil**, the Program Change commands sent are the same as in Program or Setup mode. If the setting is **QA Extended** or **QA Kurzweil**, the K2661 sends Program Change commands that correspond to the current Quick Access bank and the entry you select, not the actual program number of the entry. See *Program Change Formats* on page 10-11.

Everything you need to know about using Quick Access mode for performance is covered in Chapter 2, in the section called *Playing the Presets*, so we'll move on to the Quick Access Editor, which you'll use to create your own Quick Access banks.

The first step in editing Quick Access banks is to select Quick Access mode. Then use the **Chan/Bank** buttons to select the bank you wish to edit. The currently selected bank is shown in the top line of the Quick Access-mode page. Press the **Edit** button, and you enter the editor, where you can examine each entry in the bank you selected. The Quick Access Editor page looks like this:

```

EditQuickAccess          <>Entry:9
                        998 Blues in CMOS
Entry:  Type:          999 Default Program
9       Program       1 Righteous Piano
                        2 Mondo Bass
                        3 Killer Drums
                        4 Elvis Again
Name  Save  Delete  Dump

```

The top line gives you the usual mode reminder, and shows you which of the ten entries you're looking at. The cursor is highlighting the object (program or setup) that's stored in that entry.

The easiest way to edit the bank is to use the **Chan/Bank** buttons to scroll through the ten entries. The entry number changes both at the top of the page, and at the left of the page. As the entry number changes, the highlighted objects at the center of the page change as well, showing you what's stored in each entry. On the page above, for example, entry 9 is the current entry. The Type parameter tells you that the object stored at entry 9 is a program. The cursor highlights the program's ID and name.

In this example, you could select a different program with your favorite data entry method. If you wanted to store a setup in that entry instead of a program, you would move the cursor to the Type parameter and change its value to **Setup**. The list of objects would change from the program list to the setup list, and you could move the cursor back to the setup list and select another setup. When you select the Entry or Type parameter, the list of objects at the right disappears, leaving only the currently selected object. This makes it easier to see when it's not highlighted by the cursor.

When you've filled each entry with the object you want, press the **Name** soft button if you want to rename the bank, or press the **Save** soft button to begin the save procedure. Press the **Dump** soft button to dump the bank via MIDI System Exclusive.



# Chapter 9

## Basic Effects Mode

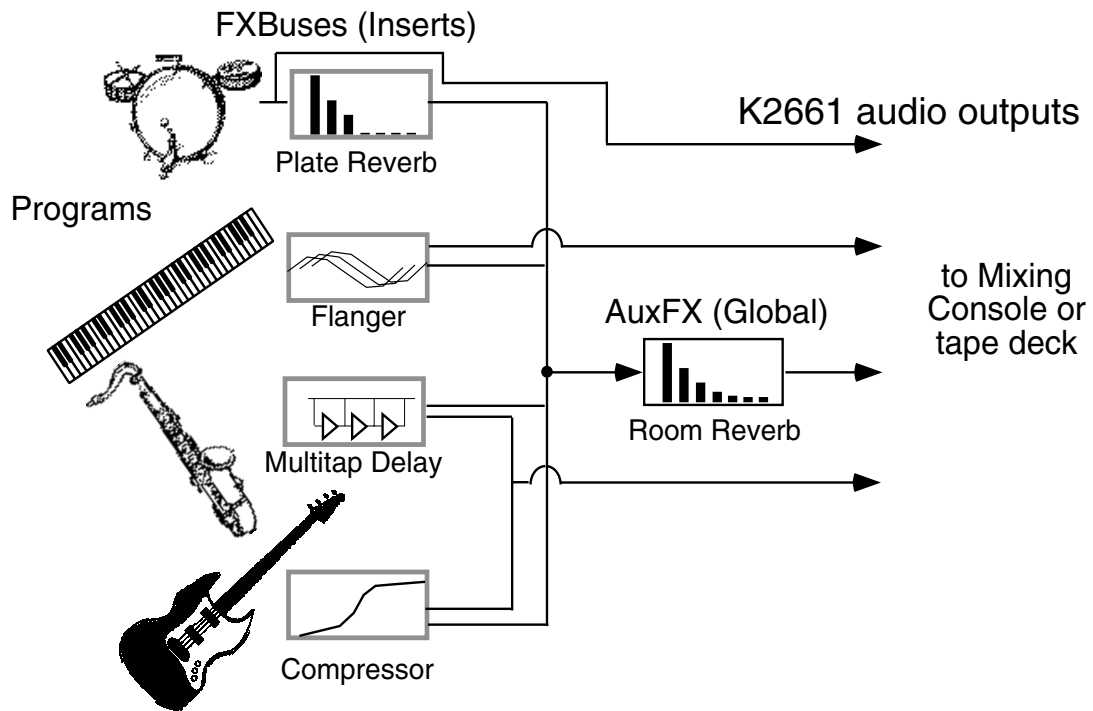
### Introduction

Effects mode puts the power of an entire studio—equalizers, signal processors, and mixers—inside your K2661. It allows you to define processing functions, signal paths, and balances, on sounds created by the K2661 and, in Live mode, sounds from *any* source.

When you're in Program, Setup, or Quick Access mode, you can take advantage of FX Modulation routings (FXMods, which are preset overrides for studio parameters) for real-time control over any aspect of your studio—either from the buttons, sliders, wheels, and pedals on your K2661, or from any MIDI source at all, like a sequencer or a dedicated controller.

Effects mode provides up to five individual effects processors. Four of these are designed to operate on their own individual buses, and the fifth, the “Auxiliary” processor, is designed to be global.

When the K2661 is being used multi-timbrally, Effects mode can provide separate processors for each of several instruments. For example, let's say four K2661 channels are in use, under the control of a sequencer. Each instrument on each channel can have its own processing: flanger for the piano, delay for the sax, compression for the bass, and tight reverb for the drums. Then all of these signals could go through a global Aux processor, where they could be given a room reverb. At the outputs of the K2661, the reverb could show up on one pair, while the other instruments' dry or processed (prereverb) signals could show up at the others, either in mono or stereo. Or the entire mix, carefully balanced and panned, could appear at a single pair, ready to be recorded or played through a PA.



**Figure 9-1 A typical KDFX Studio**

Alternatively, in a live performance setting, you can use the KDFX in Setup mode as an entire orchestra, with different instruments layered on top of each other and mapped to different parts of the keyboard, all with their own signal processing—and that signal processing can change completely in an instant when you call up a new setup.

These are just two of the huge variety of scenarios possible with KDFX. The structure of KDFX is so flexible that you should never run out of ideas for how it can be used.

## Terminology

Before starting to work with Effects mode, take a minute to familiarize yourself with the relevant terms. Understanding the words will help you get the concepts more quickly. Read this chapter first, and refer to it later if something isn't clear.

### Studio

A studio is the entire Effects-mode environment, consisting of inputs, FXBuses, an Aux Bus, FX presets, parameter settings, overrides, and outputs. One studio is current at a time in Effects mode. Studios are objects, like K2661 programs, and have a number. They can be linked to programs or setups using the FXCtrl setting. The K2661 comes with a large number of factory studios. You can modify them or create new ones and store them in RAM.

## FXBus (also called Insert FXBus)

A signal path with some kind of processing—a delay, reverb, flanger, etc.—on it. It has stereo inputs and outputs, and contains an FX preset that determines the kind of processing to be applied. A studio has four insert FXBuses, each with its own FX preset, as well as a fifth FXBus, known as the Aux bus.

## Aux FX Bus (Aux Bus)

A separate bus from the insert FXBuses, which is placed in the signal path after the output of the insert FXBuses, so that it can act as a global processor.

## FX Preset

Determines the type of signal processing that is present on an FXBus or the Aux Bus. FX presets consist of an algorithm, plus the settings of the parameters associated with that algorithm. Any number of FX presets can use the same algorithm. The K2661 comes with a large number of FX presets, and you can modify them or create new ones and store them in RAM. Algorithms, however, are stored in ROM, so you can't change them. Each FXBus and the Aux Bus gets its own FX preset, so up to five FX presets can be active in a studio at a time.

## Algorithm

A specific type of signal processing, like a hall reverb, plate reverb, chorus, flanger, pitcher, compressor, or rotary speaker, which is at the core of an FX preset. KDFX comes with a large variety of algorithms, which are stored in ROM, and are not user-definable. Additional algorithms, supplied by Kurzweil, can be loaded in from disk or SmartMedia when they become available. Algorithm parameters (RT<sub>60</sub>, delay feedback, pitch change, etc.) are user-definable; the K2661 stores the values of those parameters in RAM as part of an FX preset.

## Size/PAUs

How big an algorithm is, in terms of how much processing power it needs to operate. Size is measured in Processor Allocation Units, or PAUs. The simplest algorithms require only 1 PAU, while more complex ones require 2, 3, or even 4. The total number of PAUs available to the four insert FXBuses is 4. PAUs can be allocated automatically as FX presets are assigned to FXBuses, or manually by the user. The Aux Bus has its own set of 3 PAUs, which are not shared with the insert FXBuses.

## Parameters

Refers to the user-controlled settings for the different characteristics of an algorithm. For example, the user can set the reverb time (RT<sub>60</sub>) of an algorithm to **3.5s**, or the delay feedback of a delay line to **90%**, or the dynamic ratio of a compressor to **10:1**. Parameters are stored, along with the algorithm they modify, as part of an FX preset. In certain cases, the parameter settings within a studio can be overridden, either using FXMods or bus overrides.

## FXMods

FXMods give you real-time control over many of the parameters within a studio. Parameters on any of the pages of the Studio Editor can be controlled by MIDI commands—such as physical controller input or sequencer data—or by internal K2661 functions like LFOs, envelopes, clocks, or key states. FXMods are not part of a studio or FX preset; they're part of a program or setup.

A program or setup can have up to 18 FXMods, as well as two dedicated LFOs, ASRs, and FUNs for further effects control. The FXMods and the LFOs, ASRs, and FUNs are defined on seven FX pages in the Program and Setup Editors.

## Bus Overrides (or BusMods)

Bus overrides allow parameters within an FX preset to be controlled on the FXBus Editor page in the Studio Editor, outside the FX Preset Editor page. Any two parameters in an FX preset can be brought out to its FXBus page. Bus overrides are stored as part of the studio, not as part of the FX preset. They are useful when an existing FX preset is close to what you want, so that you can tweak it without having to create a new FX preset.

## FXCtrl

The effects mode is determined by the value of the FXCtrl parameter, which is found on the Effects-mode page. It determines how the K2661 selects studios when you change programs or setups, and determines whether you have real-time control over studio parameters—in other words, whether FXMods are active.

If the value of FXCtrl is **Program** or **Auto**, then as you change programs in Program mode, the K2661 also loads the studio linked with that program. This activates all the FXMods defined within the program.

If the value of FXCtrl is **Setup** or **Auto**, then as you change setups in Setup mode, the K2661 also loads the studio linked with that setup. This activates all the FXMods defined within the setup.

If the value of FXCtrl is **Auto**, and the value of FX Channel is **Current**, then when you're in Program, Setup, Quick Access, or Song mode, programs, setups and songs automatically use their corresponding studios. In Program, Setup, and Quick Access modes, the studio corresponds to the current program or setup. In Song mode, the studio corresponds to the program on the song's assigned effects channel (which is determined by the value of the EffectChan parameter on the COMMON page in the Song Editor).

If the value of FXCtrl is **Master**, changing programs or setups does not load an associated studio; the current studio is defined by the Studio parameter on the Effects-mode page. Any FXMods defined in the current program or setup are inactive.

## Studio Editor

To get to the Studio Editor, go to the Effects-mode page, highlight the current studio, and press **Edit**. The Studio Editor contains the following pages:

The **INPUT** page, where signals coming from the K2661's four stereo output buses are routed to one or more FXBuses.

The **FXBUS** page, for assigning FX presets to the four FXBuses, and for defining bus overrides.

The **AUXFX** page, for assigning an FX preset to the Aux Bus, and for defining bus overrides.

The **OUTPUT** page, where the signals coming from the FXBuses and Aux bus are routed to the K2661's eight physical outputs (analog and digital).

The FX Preset Editor, which is nested within the Studio Editor, and which contains three pages. The FX Preset Editor is where you select algorithms and set parameters for the FX presets, as well as doing administrative things like naming and saving. You can get to the FX Preset Editor from the either the FXBUS page or the AUXFX page, by highlighting the current FX preset and pressing **Edit**.

**Name**, **Save**, **Delete**, and **Dump**, for doing file management on your collection of studios.

You can also get to the Studio Editor from the Program and Setup Editors, by pressing **Edit** when the Studio parameter is selected (this is the most convenient method for getting into the Studio Editor). In both the Program and Setup Editors, the Studio parameter is on the KDFX page.

## MAIN Page

The KDFXMode:MAIN page gives you a summary view of the current effects configuration, including the current studio, the FX Presets assigned to each of the five effects buses, and the bypass status of each bus.

```
KDFXMode:MAIN    FXCtrl:Auto    <>Enable
Studio:113 PltEnvFI4T Plate    Free:0
FX1    43 Plebe Chamber    -    Size:1
FX2    902 Synth Env Filter    B    Size:2
FX3    735 Bap ba-da-dap    -    Size:1
FX4    0 None    B    Size:0
Aux    103 BigPredelayPlate    B    Size:3
MAIN    CTRL    EQBYP    FXBYP    BUSMUT    Enable
```

**Figure 9-1** Effects mode: the KDFXMode:MAIN page

As with every other page, the top line of the KDFXMode:MAIN page identifies the page you're on. It also shows you two other important features of Effects mode:

**FXCtrl:** this is a parameter on the Ctrl page, which is accessible with the **CTRL** soft button.

**Enable state:** Shows whether KDFX is currently enabled or if any part of KDFX is bypassed or muted.

The second line of the display shows the ID and name of the current studio. When you enter KDFX Mode directly (i.e., not through another one of the K2661's editors) you can scroll through the displayed list of studios. This allows you to choose a different studio on the KDFXMode:MAIN page. When FXCTRL is set to Master (see page 9-9), you can also do this, even when you have entered KDFX Mode from within another editor.

If you select the studio then press the **Edit** button, you'll go to EditStudio:FXBUS page, where you can make changes to each bus within the studio.

The second line also shows the number of PAUs available for the current studio ("Free:" on the right-hand side). This number will be 0–4, since in each studio four PAUs are available for the four insert FXBuses (the AuxFX bus has its own fixed set of three PAUs).

The next five lines show the IDs and names of the FX Presets assigned to the five effects buses (insert FXBuses 1–4 and the AuxFX bus). You can't change these assignments on the KDFXMode:MAIN page; to do that you would highlight the Studio name (line two of this page) then press **Edit**. This takes you to the Studio Editor, on the appropriate FXBUS page for the first bus. Use the Chan/Bank buttons to move between buses.

Each of these five lines also indicates the bypass status for the five buses, as well as the number of PAUs used by each FXBus. A dash (–) indicates active/enabled, and **B** indicates bypassed/disabled. You can change the bypass status for a bus by moving the cursor to this field and changing it with either the alpha wheel or pressing one of the increment/decrement buttons.

The size of each FX Preset is measured in PAUs (processor allocation units). FXBuses 1–4 can all use up to four PAUs, but the studio can use a maximum of four total PAUs. The AuxFX bus can use up to three PAUs independent of the insert FXBuses.

## Soft Buttons in Effects Mode

The **MAIN** button takes you to the KDFXMode:MAIN page, where you can view the current studio and the FX Presets assigned to the five KDFX buses.

The **CTRL** button takes you to the KDFXMode:CTRL page, which contains parameters that determine which studio gets selected when you select a program, setup, or song.

### Soft Buttons: Configuring Bypasses

You can individually bypass any of the EQ and effects inputs, and also mute any of the FXBuses (the four insert FXBuses and the AuxFX bus). In the enabled state, nothing is muted or bypassed. The K2661 always starts up in the enabled state.

Use these soft buttons to perform bypasses and muting:

<b>EQBYP</b>	Displays EQ Bypass page, where you can bypass the EQ on each individual input bus.
<b>FXBYP</b>	Displays FX Bypass page, where you can bypass the effects on individual FXBuses.
<b>BUSMUT</b>	Displays the BusMute page, where you can mute the output of individual FXBuses.
<b>BypAll/Enable</b>	Toggles between enabled state and default bypass state (all buses bypassed, none muted). If you have created a custom bypass scene, <b>BypAll</b> resets it to the default bypass state. See page 9-12 for information on creating a custom bypass scene.

You may also use either of the **Chan/Bank** buttons to toggle between enabled and bypassed states. This will often be preferable, since **Chan/Bank**, unlike **BypAll**, does not reset the bypass state to the default (all buses bypassed, none muted). Instead, **Chan/Bank** toggles between the enabled state and any custom bypass scene you may have created, allowing you to audition a studio with and without bypasses.

## Effects Bus Editor

The FXBus Editor lets you create effects Preset chains on any of the four stereo effects buses. See page 9-15 for more information about chaining effects.

```

EditStudio:FXBUS Size:3 Free:0 <>FXBUS:1
FX1 → Rvrb →
FX: 1 NiceLittleBooth
Wet/Dry :42%wet
Out Gain :0.0dB Alloc:Auto
<more> INPUT FXBUS Add Remove <more>

```

Figure 9-2 Effects Bus Editor display -- single effect

```

EditStudio:FXBUS Size:1 Free:0 <>FXBUS:1
FX1 → Rvrb → Chor → Dly → Flng →
FX: 1 NiceLittleBooth
Wet/Dry :42%wet
Out Gain :0.0dB Alloc:Auto
<more> INPUT FXBUS Add Remove <more>

```

Figure 9-3 Effects Bus Editor display -- four chained effects

The **Add** and **Remove** buttons allow you to define your own chains of effects using up to four FX Presets. The **Add** button creates an effects block (shown as a box) to the right of the current cursor position in the effects chain. You can use a total of four effects in any studio, so if you create a four-block effects chain on a bus then you won't be able to use any effects on the other buses in that studio. Your K2661 keeps track of effects usage for you, and won't let you add an effects block to a bus if you're already max'ed out.

The **Remove** button deletes the effects block that the FXBUS editor cursor is on. Adding and deleting effects blocks may cause audio glitches in any signal path and should not be done during critical listening.

Each FX Preset in an effect chain has two "override" parameters (BusMods) that are displayed when that FX block is selected. By selecting the name of an override parameter (e.g., Wet/Dry), you can scroll to choose from any other available parameter.

Each effect also has its full complement of real time modulators as defined and displayed in the Program and Setup editors.

## Effects Send Page

The FXSEND page lets you send the output of each stereo effects bus to the stereo mixdown and auxiliary buses.



Figure 9-4 Effects Bus Send display



## The CTRL Page

The CTRL page is where you set a variety of important parameters for KDFX, including the Effects Control Mode (FXCtrl) for the current studio.

The CTRL page is also where you set the FX Channel for the current studio.

```
KDFXMode:CTRL    FXCtrl:Auto    <>Enable
Studio:49 Snoboard Room Hall
FXCtrl:Auto
FXChan:Current
```

```
MAIN CTRL EQBYP FXBYP BUSMUT Enable
```

**Figure 9-5**      **Effects Control page**

### Effects Control Mode (FXCtrl)

FXCtrl determines how the K2661 selects studios as you change programs or setups, and determines whether you have real-time control over studio parameters—in other words, whether FXMods are active.

If the value of FXCtrl is **Program** or **Auto**, then as you change programs in Program mode, the K2661 also loads the studio linked with that program. This activates all the FXMods defined within the program. If the value of FXCtrl is **Setup** or **Auto**, then as you change setups in Setup mode, the K2661 also loads the studio linked with that setup. This activates all the FXMods defined within the setup. If the value of FXCtrl is **Auto**, and the value of FX Chan is **Current**, then when you're in Program, Setup, Quick Access, or Song mode, programs, setups and songs automatically use their corresponding studios. In Program, Setup, and Quick Access modes, the studio corresponds to the current program or setup. In Song mode, the studio corresponds to the program on the song's assigned effects channel (which is determined by the value of the EffectChan parameter on the COMMON page in the Song Editor).

If the value of FXCtrl is **Master**, changing programs or setups does not load an associated studio; the current studio is defined by the Studio parameter on the Effects Mode page. Any FXMods defined in the current program or setup are inactive.

### Effects Control in Embedded Editors

In the parlance of V.A.S.T., an embedded editor is an editor that you enter while you are already in another editor. An example of this would be entering the KDFX Studio Editor while you are already in the Program Editor. In this sort of situation, an editor may function differently than if you had entered it directly from a performance mode.

When you enter the KDFX Studio Editor from within another editor (for example, you are already in the Program Editor when you press the **Effects** button), KDFX will revert FXCtrl to Auto and FXChan to Current if you attempt to perform an operation that the software doesn't support. For example, you cannot change a program's assigned studio by pressing the **Effects** button to enter the KDFX Editor while you are already within the Program Editor. If you set FXCtrl to Master you will be able to audition different studios, but the software will not let you change a studio. The correct way to change the studio used by a program is to press the **KDFX** soft button from within the Program Editor.

## Effects Channel (FX Chan)

This parameter is closely linked to the FXCtrl parameter—the values available for FX Chan depend on the setting for FXCtrl. FX Chan affects both studio selection and real-time control of KDFX—again, depending on the value of FXCtrl.

### When FXCtrl is Master

The effects channel is irrelevant when FXCtrl is **Master**. KDFX is under the control of the studio you select with the Studio parameter. No FXMods apply.

### When FXCtrl is Program

The available values for FX Chan are **Current**, and **1–16**. When FX Chan is **Current**, studio selection is determined by whatever program is assigned to the current K2661 channel (as shown in the top line of the display in Program mode or Effects mode). That program's FXMods are also enabled, and they respond to MIDI Controller messages received on the current channel.

When FX Chan is any value from **1** to **16**, studio selection is determined as follows: in Program mode it's determined by the program assigned to that channel; in Setup mode, it's determined by the program in the zone assigned to that channel. FXMods respond to MIDI Controller messages as follows: in Program mode, the FXMods of the program on the channel specified by FX Chan respond to messages on that channel; in Setup mode, the FXMods of the program in the zone using the channel specified by FX Chan respond to messages on that channel.

The Studio parameter disappears, since studio selection is under program control.

### When FXCtrl is Setup

The only available value for FX Chan is **None**, since studio selection and FXMods are determined by the program in Zone 1 of the current setup. The FXMods of the program in the zone using the channel specified by FX Chan respond to MIDI Controller messages on that channel. The Studio parameter disappears, since studio selection is under setup control.

### When FXCtrl is Auto

The available values for FX Chan are **Current**, and **1–16**. When FX Chan is **Current**, studio selection and FXMods are determined as follows: in Program mode, it's determined by the program assigned to the current K2661 MIDI channel; in Setup mode, it's determined by the program in Zone 1 of the current setup; in Song mode (or during song playback regardless of mode), it's determined by the program on the channel designated as the song's effects channel—as specified by the EffectChan parameter on the COMMON page in the Song Editor. FXMods respond to MIDI Controller messages as follows: in Program mode, the FXMods of the program on the K2661's current channel respond to messages on that channel; in Setup mode, the FXMods of the program in Zone 1 of the current setup respond to messages on the channel used by Zone 1.

When FX Chan is any value from **1** to **16**, studio selection is determined as follows: in Program mode it's determined by the program assigned to that channel; in Setup mode, it's determined by the program in Zone 1 of the setup (ignoring the FX Chan setting). FXMods respond to MIDI Controller messages as follows: in Program mode, the FXMods of the program on the channel specified by FX Chan respond to messages on that channel; in Setup mode, the FXMods of the program in Zone 1 of the setup respond to messages on that channel (again, ignoring FX Chan).

## Studio

When FXCtrl is **Master**, the Studio parameter selects the studio for the entire K2661. When FXCtrl is **Program** or **Setup**, this parameter is unavailable, since setup selection and FXMods are determined by program or setup selection. When FXCtrl is **Auto**, setup selection and FXMods are under program or setup control, but the Studio parameter is still available; its value reflects the studio assignment for the program or setup that was current before you entered Effects mode. You can change the value of the Studio parameter while in Effects mode, so you can hear how different studios affect the current program or setup, without having to enter the Program or Setup Editor. Any changes you make here revert to their previous settings when you exit Effects mode.

## Bypass and Mute pages

You can bypass effects buses, inside or outside of an editor, by pressing the **Effects** button followed by either the **BypAll** soft button or one of the **Chan/Bank** buttons. Although similar, the two methods are slightly different:

- The **BypAll** soft button globally bypasses all effects buses, and also resets the default bypass state to bypass all buses. When you press this soft button it changes into the **Enable** soft button, allowing you to toggle between the state where all buses are bypassed and the state where all are enabled.
- Either **Chan/Bank** button toggles between the enabled state and the current bypass state. The current bypass state is either the default (all buses bypassed) or the custom bypass scene you have created. See the section that follows for information about creating a bypass scene.

Pressing the **Effects** button again, or pressing **Exit**, puts you back where you were.

### Creating a Custom Bypass Scene

You create a custom bypass “scene” (e.g., effects bypassed on one bus, but not on the other three) by using the soft buttons on the EQBYP, FXBYP, and BUSMUT pages to isolate sounds or effects. You can then toggle between an all-enabled state and your custom scene by pressing either of the **Chan/Bank** buttons (to the left of the display) while in KDFX Mode.

The system indicates whether anything at all is bypassed or muted by showing “Bypass” at the far right of the top line on the display; if nothing is bypassed, this field shows “Enable.” Any settings from the FXBYP page are also indicated on the KDFXMode:MAIN page as either a “B” (bypassed) or a “-” (enabled). EQ Bypass and Bus Mute settings, however, are not indicated on the KDFXMode:MAIN page. If you exit this mode with anything bypassed, the **Effects** button's red LED stays lit to remind you that something is not active.

## The EQBYP Page

```

KDFXMode:EQBYP   FXCtrl:Auto   <>Enable
Studio:113 PltEnvFI4T Plate

EQ A   LoShelf-HiShelf           :In
EQ B   LoShelf-HiShelf           :Out
EQ C   LoPass1-HiShelf           :In
EQ D   HiPass1-LowPass1          :In
MAIN EQ A EQ B EQ C EQ D

```

Figure 9-6 EQ Bypass Page

The **MAIN** soft button takes you to the KDFXMode:MAIN page. The soft buttons **EQ A**, **EQ B**, **EQ C**, and **EQ D** toggle the bypass/active status for the EQ on the corresponding input buses.

The EQBYP page looks a little different when there are mono inputs to the studio. In this case, press the **L/R** soft button to toggle between left and right mono inputs for a bus.

```

KDFXMode:EQBYP   FXCtrl:Auto   <>Enable
Studio:113*PltEnvFI4T Plate

EQ A/L LoShelf-HiShelf           :In
EQ B   LoShelf-HiShelf           :Out
EQ C   LoPass1-HiShelf           :In
EQ D   HiPass1-LowPass1          :In
MAIN EQ A/L EQ B EQ C EQ D L/R

```

Figure 9-7 EQ Bypass Page with Mono Inputs

## The FX Bypass Page

```

KDFXMode:FXBYP   FXCtrl:Auto   <>Enable
Studio:113 PltEnvFI4T Plate
FX1    43 Plebe Chamber           :Active
FX2    158 Soft Chorus             :ByPass
FX3    2 Stereo Echoes             :Active

Aux    31 Platey Room              :Active
MAIN FXBus1 FXBus2 FXBus3 AuxFX

```

Figure 9-8 FX Bypass Page

The **MAIN** soft button takes you to the MAIN page. The **FX1–FX4** and **AuxFX** soft buttons toggle Bypass/Active status for the effect on the corresponding bus.

### The Bus Mute Page

```
KDFXMode:BUSMUT FXCtrl:Auto <>enable
Studio:113 PltEnvFI41 Plate
FXBus1 43 Plebe Chamber :Active
FXBus2 158 Soft Chorus :Muted
FXBus3 2 Stereo Echoes :Muted
FXBus4 --- :Active
AuxFX 31 Platey Room :Active
MAIN FXBus1 FXBus2 FXBus3 FXBus4 AuxFX
```

**Figure 9-9** Bus Mute Page

The **MAIN** soft button takes you to the KDFXMode:MAIN page. The soft buttons **FXBus1–FXBus4** and **AuxFX** toggle the mute/active status for the corresponding input buses.

## Chaining Effects

One of the most powerful features in KDFX is effects chaining, which allows you to send a signal through four consecutive KDFX effects. The screen below shows an example of this:

```

EditStudio:FXBUS Size:1 Free:0 <>FXBus:1
FX1 →Rvrb→Chor→Dly→Flng→
FX: 1 NiceLittleBooth
Wet/Dry :42%wet
Out Gain :0.0dB Alloc:Auto
<more> INPUT FXBUS Add Remove <more>

```

**Figure 9-10** Effects Bus Editor display -- four chained effects

Effects chaining allows the 4 PAUs of processing shared among Buses 1-4 of a Studio to be used in series. You can chain one FX Preset into another, into another, up to four in a row, until you run out of PAUs. This is done by removing processing “blocks” from one bus, and adding them to another. As no effect is less than 1 PAU, and only 4 PAUs are available across Buses 1-4, any Studio may have a maximum of 4 blocks, arranged however you please, in which to select Presets (not counting the Aux bus which is unaffected by chaining).

The FXBUS page shows chained effects at the top of the display (underneath the top menu line). As an example, start from Program Mode, press the **Effects** button, then select Studio 700 Flanger Trio:

```

KDFXMode:MAIN FXCtrl:Auto <>Enable
Studio:700 Flanger Trio Free:0
FX1a 180 Ned Flangers - Size:1
1b 172 Sweet Flange - Size:1
1c 181 WisPy Flange - Size:1
1d 40 SmallDrumChamber - Size:1
Aux 108 Roomitizer - Size:2
MAIN CTRL EQBYP FXBWP BUSMUT Enable

```

You can see that this studio has three flange effects, followed by a reverb. The effects are numbered 1a through 1d to indicate that they are all part of FXBUS 1, instead of four separate effects buses. Now press **Edit** to go into the Studio Editor. The top of the display shows the four effects chained together. Each block contains an abbreviation based on the algorithm used by the Preset:

```

EditStudio:FXBUS Size:1 Free:0 <>FXBus:1
FX1 →Flng→Flng→Flng→Rvrb→
FX: 180 Ned Flangers
Wet/Dry :42%wet
Out Gain :0.0dB Alloc:Auto
<more> INPUT FXBUS Add Remove <more>

```

The name of the FX Preset for the currently highlighted block is now shown underneath the signal path graphics. In this example, you will see the FX Preset Ned Flangers if the first block is highlighted. You still have 2 Bus Overrides (or Bus Mods) per block, which appear just below the name of the Preset.

Use the left and right cursor buttons to select each block. When a block is selected, move the alpha wheel or press the + or - buttons to select a different FX Preset (you can also change the Preset by cursoring to the full name of the Preset after the FX: label).

Notice that the unhighlighted blocks have a box around them. This shows they are active. Since this studio has 4 blocks, each block can use only 1 PAU. If you select an effect that uses more than one PAU, one of the blocks will become inactive and the box surrounding that block will disappear. For example, if you change the first block to FX Preset 183 NarrowResFlange, the box around block 4 disappears. The top line of the display shows you this FX Preset uses 2 PAUs. As in the past, if the Allocation parameter is set to Auto, the lower number blocks have precedence, so block 4 is the one that becomes inactive. If you highlight block 4 at this point, you will see the FX Preset shown in parenthesis, again showing it is not active.

The **Chan/Bank** buttons move you through the four FXBUSes. Since no effects are available in this case, you will see a line with no blocks on them if you look at any bus except FXBUS 1. You can still use a bus to send another signal to the AUX without the chain, by the way, since KDFX has been designed to offer you maximum flexibility.

### Gain Staging in Effects Chains

When chaining Presets together, it is sometimes necessary to adjust the levels between blocks, most often to pad the level going into the next block to prevent unwanted clipping. While most algorithms have both an In Gain and an Out Gain parameter, In Gain is not selectable as a Bus Mod. In fact, any Preset beyond the first in a chain cannot use In Gain, and will display the value inside the Preset in parentheses. We suggest, when necessary, choosing Out Gain as a Bus Mod to adjust the output level of an effect, instead of trying to pad the input of the following effect. Of course, you can always edit FX Presets directly and customize them for your chain.

### Checking Out Some Chains

For examples of studios with chains, check out studios 700-719. By setting the FX Ctrl parameter (KDFXMode:CTRL page) to Master, and the OutPair parameter (MIDI Mode:Channels page) to KDFX-A, you can scroll through Programs on a given MIDI channel and audition these studios as they were intended to be heard, with a variety of input source material.

## The Structure of Kurzweil Digital Effects (KDFX)

### The Studio

All Effects-mode functions are contained in a studio. Only one studio is active at a time.

Each studio has four stereo or eight mono inputs. The sources for these inputs are the four output pairs available in two places: the Pair parameter on the OUTPUT page in the Program Editor, and the Out parameter on the CN/PRG page in the Setup Editor. In both cases, the available values are KDFX-A, KDFX-B, KDFX-C, and KDFX-D. Instead of going directly to the K2661's physical outputs, these program outputs go to the inputs of a studio. Only after going through the studio does the audio output appear at the physical outputs. The inputs to the studio can come from one multi-layered program, or from several programs, or from the K2661's own sample input when the instrument is in Live mode.

The input signals pass through an input routing system, as specified by the settings on the INPUT page in the Studio Editor. Here, the signals go through individual two-band equalizers, or EQs. Then each input, mono or stereo, is routed to any two of the four FXBuses.



Each of the FXBuses contains its own signal-processing program, called an FX preset. Each FX preset has a set of parameters: for example the  $RT_{60}$  value on a reverb, or the Feedback level on a delay line. These parameters can be fixed as part of the FX preset, or they can be externally controlled, from one of two different places. The four FXBuses are also called Insert FX, because in a conventional studio, that's where they would be found: in the insert loop of a mixer, between the channel input and the Mix bus.

There is also a fifth FX preset, which is located on the Auxiliary FX Bus (the Aux bus). The Aux bus follows the four main FX presets in the signal path, and is normally configured as a global processor.

Each FX preset consists of an algorithm, modified by user-definable parameters. An algorithm is a processing function, like a reverb, flanger, or compressor; or a combination of processing functions in a particular order, like a flanger followed by a delay followed by a reverb. The algorithms themselves are fixed in ROM, like Kurzweil ROM samples, but you can change the values of their operating parameters.

Each algorithm requires a certain amount of processing power, which is expressed in Processing Allocation Units (or PAUs). Simple algorithms require 1 PAU, while more complex algorithms require up to 4 PAUs. The amount of processing power available in each FX preset is set by its Allocation parameter. When you are selecting an FX preset for an insert bus, the number of PAUs its algorithm requires appears on the display, so you can keep track of how many PAUs are in use.

PAUs are shared among the four insert buses. There is a limit to the total number of PAUs that the insert buses can use, and that limit is four. PAUs can be manually preassigned to specific FXBuses, or using "Auto" mode they can be assigned automatically as FX presets are assigned to the buses. The Aux bus has a separate set of PAUs—three of them—which are *not* shared with the insert buses.

Finally, the outputs of the FX presets are passed through to an output routing system—as specified by the settings on the OUTPUT page in the Studio Editor—where they are sent to the physical outputs of the K2661.

The following equation summarizes studio structure:

$$\text{Studio} = \text{EQs} + \text{Input-page settings} + \text{FX presets} + \text{Output Editor settings}$$

The next page shows a schematic overview of studio structure.

## Basic Effects Mode

### The Structure of Kurzweil Digital Effects (KDFX)

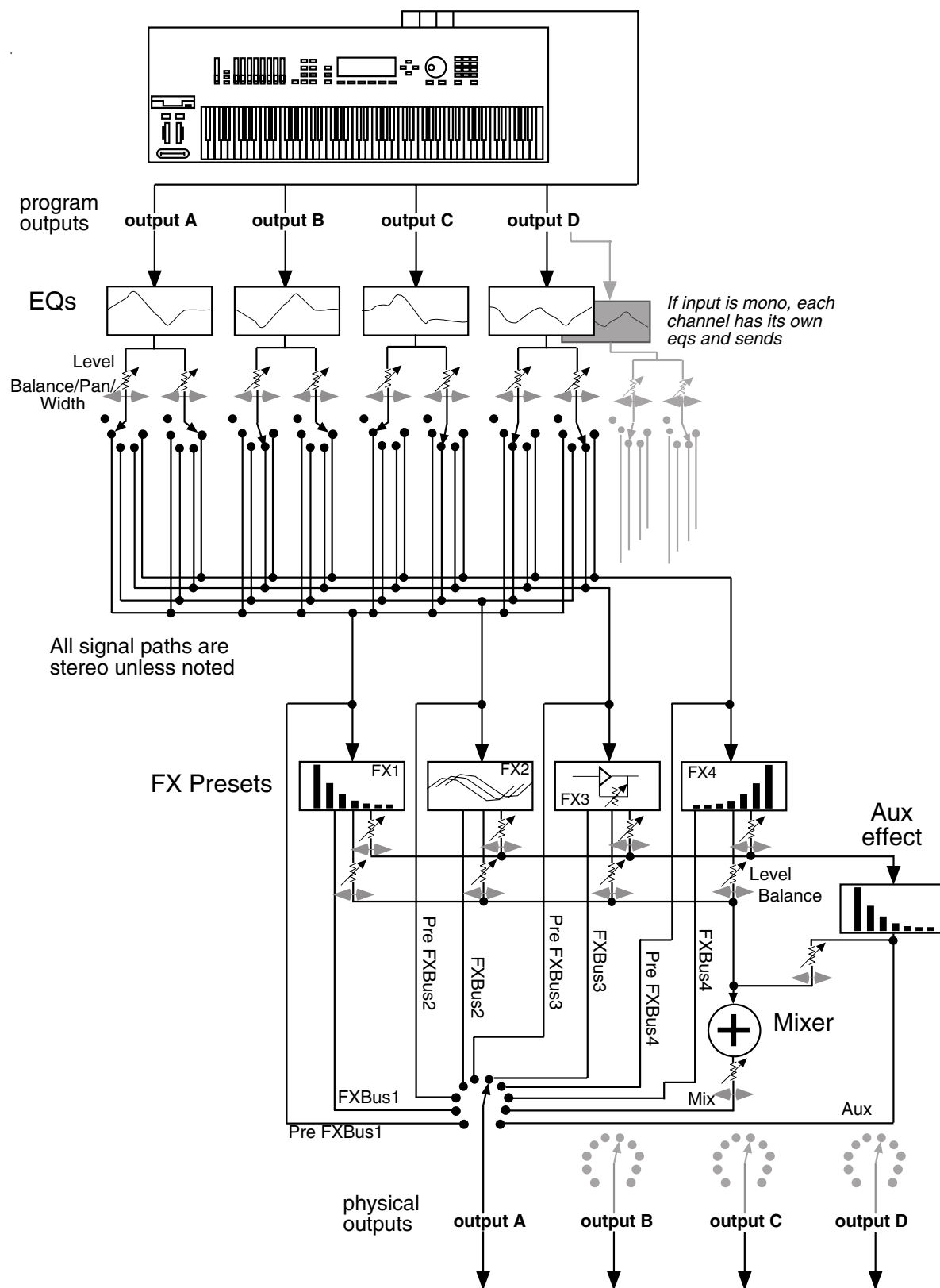


Figure 9-2 KDFX structure

## Normal Studio Structure

Many of the studios provided in ROM follow an overall organizational plan, which uses Effects mode's resources efficiently and clearly. While you're by no means required to follow this structure when creating your own studios, it's a good idea to get familiar with it, to see how Kurzweil's own engineers have approached the issue of studio organization.

Inside the ROM programs and setups, the outputs KDFX-A through KDFX-D are assigned based on the type of effects processing that would most likely be appropriate for that sound.

- Input A / FxBus1 contains a relatively simple reverb with a low Size requirement.
- Input B / FxBus2 contains an effect that does not increase the "length" of the sound (that is, no reverb or delay), something like chorus, flange, distortion, pitcher, or EQ.
- Input C / FxBus3 contains effects that take up lots of time, such as delays, and delays with reverb.
- Input D / FxBus4 is dry.
- The Aux bus contains a larger reverb (Size:3), a compressor, or a graphic EQ.

## Software Organization

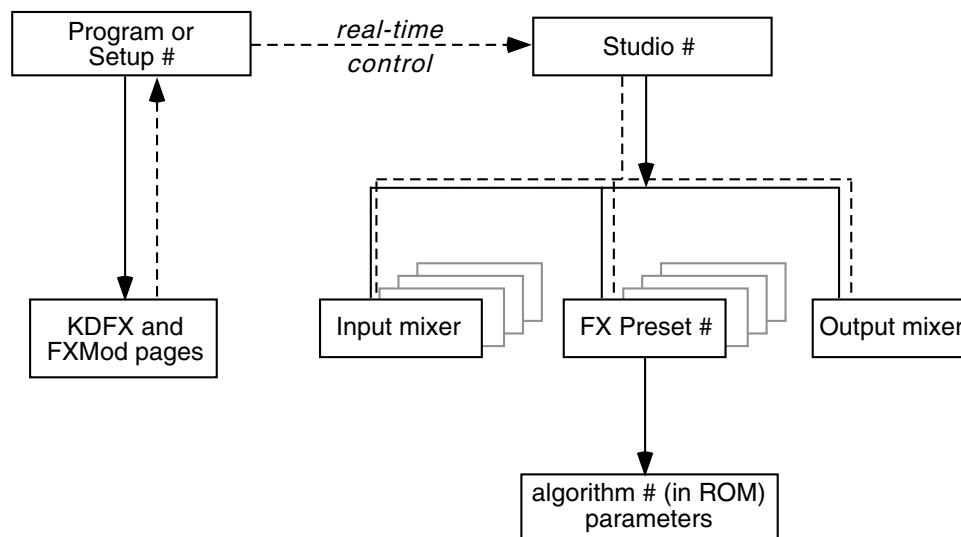
Like K2661 programs, the software user interface of a studio is organized in a "top-down" way: A studio is an "object" in K2661 terms, and the FX presets within a studio are also objects, "dependent" to the studio, the way a keymap is to a program. Studios and FX presets that you modify or create are stored in program RAM. A studio contains up to five FX presets. The same way that one keymap can be used in multiple programs, a particular FX preset can be used in multiple studios.

Like ROM samples in the K2661, effects algorithms are stored in ROM and you cannot modify them. Each FX preset contains one algorithm. Just as you can use the same sample in more than one keymap, you can use the same algorithm in as many FX presets as you like.

## Controlling Effects

Studios can be called up from the front panel, or they can be assigned to specific K2661 programs or setups. If you set the FXCtrl parameter on the Effects-mode page to **Program**, **Setup**, or **Auto**, then changing a program (if the current channel agrees with the FX Channel) or setup will select the studio associated with the new program or setup. Of course, just as you can use the same keymap in multiple programs, you can use the same studio in multiple programs or setups. Setting FXCtrl to **Master** means that the studio does not change with the program or setup.

Studio parameters are set in the Studio Editor and are initially set up with fixed values, but they can be also be controlled in real time by a wide range of control of sources, including K2661 onboard knobs, sliders, and ribbon; various internal software functions; and external MIDI commands like those coming from a sequencer or a dedicated controller.



**Figure 9-3** KDFX Software organization

## Controlling Effects with a Sequencer

A useful technique for sequencer users is to set up the program controlling the studio parameters to be a dummy program with no keymaps but with all of the FX Mods you need in place—this program produces no sound by itself, and exists only to control the studio. That way you can use a dedicated MIDI channel for studio control. See page 15-29 and page 12-21 for more information.

## Controlling Effects with SysEx messages

You can also control KDFX by sending MIDI system-exclusive (SysEx) messages to the K2661. See Appendix B in the *Musician's Reference* for specific information.

## A Tour of Effects Mode

Let's take a tour through several studios. After you do this, you should have a pretty good idea of what it's like to work with Effects mode.

## Load the Tutorial files

From the CD-ROM or SmartMedia card that came with your K2661, load the file **KDFXTUTR.K26** into bank 200...299. You can use a different bank if you like, but then your numbers and the ones in this chapter will be different.

1. Insert the disk into the drive
2. Press **Disk** to go into Disk mode
3. Set Current Disk to **SMedia**
4. Press **Load**
5. Cursor down to **KDFXTUTR.K26** (or **TUTOR1B.K25**)
6. Press **OK**
7. In the "Load this file as:" dialog, select **200...299**
8. Press **OK**
9. Press **OverWrt** (this will erase all objects in the 200s bank, so you should save anything you want to keep before doing this; press **Cancel** if you want to save objects before proceeding). If the 200s bank is empty, then **OverWrt** will not be displayed as an option; in this case, you should select **Append**.
10. Press **Exit** to get back to Program mode.

## A Simple Studio

Call up Program #199 on your K2661. In Program mode, either scroll the Alpha wheel to Program **199 Default Program**, or press **1-9-9-Enter** on the Alphanumeric pad.

Now go into Effects mode by pressing the **Effects** button, and you'll see this page:

```

KDFXMode:MAIN  FXCtrl:Auto  <>Enable
Studio:199 Default Studio  Free:4
FX1  199 No Effect  -  Size:0
FX2  199 No Effect  -  Size:0
FX3  199 No Effect  -  Size:0

Aux  199 No Effect  -  Size:0
MAIN CTRL EQBYP FXBYP BUSMUT BYPAL1

```

Now press the **CTRL** soft button. A screen such as this one will appear:

```
KDFXMode:CTRL    FXCtrl:Auto    <>Enable
Studio:199 Default Studio
FXCtrl:Auto
FXChan:Current
```

```
MAIN  CTRL  EQBYP  FXBYP  BUSMUT  Bypass
```

Scroll the Alpha wheel until the Studio parameter's value is **200 Simple**.

### The FXBus Page

Now press **Edit**, and this page appears:

```
EditStudio:FXBus Size:1 Free:3 <>FXBus:1
FX1 → D19 →
FX: 200*4 Tap BPM
Wet/Dry : 35%wet
Tempo   : 120BPM
Alloc:Auto
<more> INPUT FXBUS Add Remove <more>
```

This is the FXBus page for FXBus1. It is where an FX preset is assigned to the FXBus. Put the cursor on the box containing **200\*4 Tap BPM**—this is the current FX preset, a four-tap delay whose speed is expressed in terms of tempo (**Beats Per Minute**). Use the Alpha wheel to scroll through the many other FX presets that come with KDFX. Like all K2661 objects, those that are in RAM (like this one) will have an asterisk in their name, and those that are in ROM will not. There are three more FXBuses, which you can view by pressing the **Chan/Bank** buttons. In this studio they are all empty (No Effect).

Go back to **4 Tap BPM** on FXBus1, and play the piano sound from your keyboard. The arrow next to FX1 flashes, showing that there is audio passing through this FXBus. The arrow keeps flashing as long as the FXBus is processing audio.

Below the FX preset selector is a Wet/Dry control, which determines how much of the signal will pass through the FX preset. Below that is a Tempo control, which sets the timing of the delays. These parameters are called “bus overrides,” because they override parameters which are actually inside the FX preset itself—these parameters can be adjusted from inside the FX preset, or they can be set from out here, where they are much more convenient. If you change FX presets, these values change, because their values inside the various FX presets are all different. There will be more about bus overrides a little later in this chapter.

### Allocation

The Allocation parameter determines how many processor allocation units (PAUs) are reserved for this FXBus. The number of PAUs an FX preset uses is dependent on the algorithm at the core of the FX preset. Algorithms can use anywhere from 1 to 4 PAUs, depending on their complexity. As you scroll through the FX presets, the number of PAUs required by each one is shown on the top line of the display (Size:), along with the number of PAUs that are available (Free:) for other algorithms.

The 4 insert FXBuses have 4 PAUs to share among them, so if any bus uses more than one PAU, it means that some buses cannot be assigned an FX preset. This is a very common situation, as

you shall see. The Aux bus has its own set of 3 PAUs, which are completely independent and are not shared with the insert FXBus PAUs.

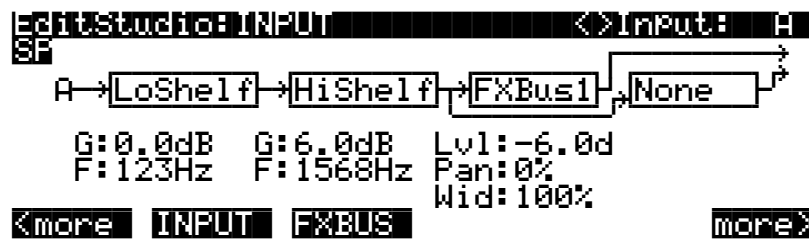
You can preassign a PAU value to an FXBus, in which case any FX presets that require more PAUs than you have given the bus cannot be loaded into the bus. If you try to put an FX preset into a bus that requires more PAUs than are currently available on that bus, the preset's name appears in parentheses—exactly the way a KB3 program appears if you try to select it on a channel that's not the KB3 channel.

In most factory studios, the Allocation parameters are set to Auto, in which case PAUs are assigned dynamically as you assign FX presets to the various buses. As you unassign FX presets from buses, or assign FX presets with smaller PAU requirements, the PAUs freed up are automatically reassigned to other buses where they are needed.

A value of Auto for this parameter allocates PAUs to lower-numbered FXBuses first, as needed.

### The INPUT Page

Press the soft button labeled **INPUT**. This page appears:



This is the INPUT page. It is showing Input A, as indicated both on the left side and in the upper right corner—use the **Chan/Bank** buttons to view the other three inputs. Input A is the first stereo signal pair coming from the K2661 Program Editor's Output section. Depending on how the K2661 is set up, this could be a single layer of a single program, or multiple layers, or multiple programs, or one or more zones from a setup, or the output from one or more MIDI channels.

The **S** at the upper left says that the Input A is being handled as a stereo feed; this can be changed to two mono feeds. The **P** means that the stereo feed has a Pan control; you can choose to make this a Balance control instead, by setting this parameter to **SB**.

If you play on the keyboard, you can see the arrow next to the letter **A** flashing, as audio is being passed through this part of the studio. The arrow on this page flashes only as long as there is an input signal present.

### Equalization

The first two blocks are the low and high EQs on the input. Put the cursor on either box and turn the Alpha wheel, and you will see the options you have available for types of EQ—these include **None**, which bypasses that EQ. The first block has more choices than the second.

The **G** underneath each block is its Gain; **0.0dB** is unity gain; the signal passes through without change. (There is no Gain parameter when a block is set to LoPass or HiPass.) **F** is the equalizer's

***Sends***

The Lvl control is the FXBus send; it sets the level of the signal to the FXBus above it. In this example, the level is backed off 6.0 dB, to compensate for the treble boost in the equalizer, so that the signal doesn't overload the FXBus.

Pan determines the position of the signal relative to the left and right sides. Width (which is not shown when you are using mono inputs) determines how much the left and right sides' signals will be separated or blended.

Press the **more>** soft button to display additional soft buttons for this page. Then press **FXSEND** to see the FXSEND page:



There are two sets of level and pan controls from the FX preset to the output mixer. Use the first set, Aux Lvl and Aux Pan, to specify how much of the sound will go to the global or Auxiliary effects bus, and how its two channels are panned. On this FXBus, the signal to the Aux bus is boosted 3.5 dB. The second set determines how much of the sound goes to the Mix bus. On this FXBus, the Mix signal is attenuated -4.5 dB.



### The AUXFX Page

Press the **AUXFX** soft button. This page appears:

```

EditStudio:AUXFX Size:3 Free:0
Aux→204*Big Chamber →Mix
Wet/Dry :58%wet          Lvl:0.0dB
Out Gain :1.5dB          Bal:0%
<more  FXSEND AUXFX OUTPUT  more>

```

This is the Global AUXFX page, and shows us what is happening on the Auxiliary effects bus. The Aux bus is a second processor, which follows the four insert FXBuses. It has its own FX preset, with bus overrides, and level and balance controls to feed it into the Mix bus. It doesn't share PAUs with the FXBuses; it has three PAUs of its own, and consequently doesn't have an Allocation parameter. The Aux bus can be routed all by itself to an output, as we'll see. In this studio, the Aux bus contains a chamber reverb.

### The OUTPUT Page

Press the **OUTPUT** soft button.

```

EditStudio:OUTPUT
Mix Lvl:0.0dB          Output A:Mix
Mix Bal:0%             Output B:FXBus1
                       Output C:Off
                       Output D:Off
<more  FXSEND AUXFX OUTPUT  more>

```

The OUTPUT page is the interface to the real world. It determines which of the signals going through the various effects buses show up at the K2661's physical outputs. Outputs A and B are brought out in analog format to two pairs of stereo jacks on the rear panel. Outputs A through D can be found in ADAT digital format at the ADAT/AES Out optical jack on the rear panel. Output A alone can be found in AES/EBU or S/PDIF digital format at the ADAT/AES Out optical jack. See *Digital Output Format* on page 11-11 for more information.

In this studio, Output A is carrying the Mix, that is, the combination of the outputs of the four FXBuses (only one of which is in use) and the Aux bus. Output B is carrying FXBus1, which is the signal after it passes through the delay on FXBus1, but before it gets to the reverb on the Aux bus. The other outputs are carrying no signal.



**Note:** If you're using the Mix audio outputs, keep in mind they carry the summed signals of audio outputs A and B. Normally you would assign each of the audio outputs differently. In the example above, you might set Output B to **Off**, or Output A to **AuxFx**, to avoid applying the FXBus1 effect to the entire program.

Here's a diagram of what this studio looks like:

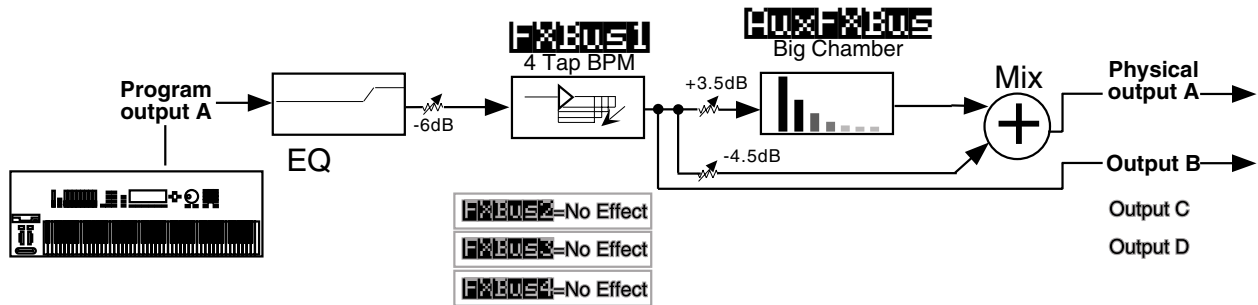


Figure 9-4 Structure of Studio 200\*

## A More Complex Studio

Press **Exit** as many times as necessary to get to Program mode. Call up Program **200 ElecPno/Flute**. This is a split keyboard program: On Layer 1, which has been assigned to the KDFX-A outputs, is an electric piano, whose key range goes up to B4. On Layer 2, which goes to the KDFX-B outputs, is a flute, whose key range starts at C5.

Press the **Effects** mode button, then the **CTRL** soft button and set FXCtrl to **Master**. Then call up Studio **201 RngMd/PFD/Plt**. Press **Edit** to look inside of this studio.

### FXBus1

On the first FXBus is an FX preset called **201\*Tut Ring Mod**. The algorithm this uses is a ring modulator, which is a processor that takes the sounds coming into it and combines them with static waveforms by adding and subtracting their frequencies, thereby creating interesting nonharmonic effects. Notice that this FX preset uses 1 PAU.

```

Edit:StudioH:FXBUS Size:1 Free:1 <>FXBUS:1
FX1  →RMod→
FX1: 201*Tut Ring Mod
Wet/Dry :90%wet
Out Gain :0.0dB
Alloc:Auto
<more INPUT FXBUS Add Remove more>
  
```

As you play on the lower part of the keyboard, the arrow next to FX1 flashes, but as you play on the upper part it doesn't. That's because the upper part of the keyboard (the flute sound) is routed to a different FXBus.

### Bus Overrides

There are two FX preset parameters on this page: Wet/Dry mix and Output Gain. These parameters actually exist inside the FX preset, and are placed on this page so you can control them without editing the FX preset itself. These are bus overrides, as described earlier. You can change both the value of the override parameter *and* the name of the parameter that shows up in the bus override: to select a different parameter, simply highlight its name and scroll the Alpha wheel. As you do so, you will see the other parameters inside the FX preset that can be brought out to this page.

These overrides (that is, which parameters are available, and their values) are stored as part of the studio, not as part of the FX preset, and therefore you don't have to create new FX presets just because you want to change a couple of parameters. There are two bus overrides available for each of the four insert FXBuses and the Aux bus.

If you don't want any parameter control on this page, select the parameter's names, then scroll the Alpha wheel until you see **None**.

## FXBus2

Press the **Chan/Bank Up** button to get to FXBus2:

```
EditStudio:FXBus Size:2 Free:1 <>FXBus:1
FX2 → FDRV →
FX: 202*Flg+Dly145BPM
Wet/Dry :90%wet
Out Gain :-2.5dB
Alloc:Auto
<more> INPUT FXBUS Add Remove <more>
```

On this bus is an FX preset called **Flg+Dly145BPM**, which uses a combination algorithm that has flanging, delay, and a reverb all rolled into one. The **145BPM** part refers to the fact that the delay times are based on a tempo of 145 BPM,

As you play the flute sound, the arrow next to FX2 flashes, and it keeps on flashing as long as the various feedback delays are sounding. It doesn't flash when you play on the lower part of the keyboard.

This FX preset uses 2 PAUs. Along with the 1 PAU in use on FXBus 1, this makes 3 of the 4 available PAUs accounted for, so the Free parameter has a value of 1.

Press **more>**, then **FXSEND** to see that the output configuration of this FXBus has the signal going to the Aux bus attenuated by -9.5 dB, and going to the main Mix bus at unity gain.

The other two FXBuses are empty, which you can confirm by pressing the **Chan/Bank** buttons a few times.

## Inputs

Now let's look at the inputs to the FXBuses. Press **<more>**, followed by the **INPUT** soft button, and see this page:

```
EditStudio:INPUT <>Input: A
SP
A → LoShelf → HiShelf → FXBus1 → None
G:12.0dB G:0.0dB Lvl:0.0d
F:370Hz F:1047Hz Pan:0%
Wid:100%
<more> INPUT FXBUS <more>
```

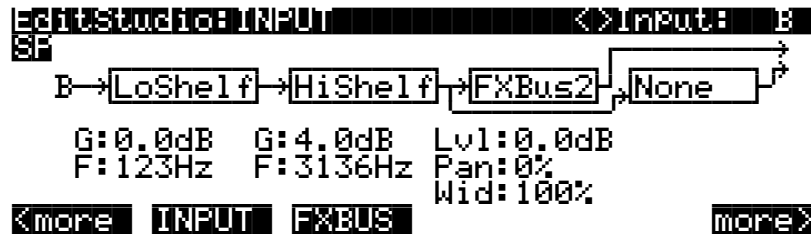
Input A carries the electric piano, coming from the program's KDFX-A outputs. Play on the piano part of the keyboard, and the arrow next to A flashes.

## Basic Effects Mode

### A More Complex Studio

This input is configured to be stereo. It has a large bass boost: 12.0 dB of everything at 370 Hz and below, which adds a strong low-frequency emphasis to the signal being ring-modulated. Its signal is being sent only to the first FXBus. The stereo separation (Width) of the signal is at maximum.

Use the **Chan/Bank Up** button to go to Input B.



This is the flute, coming from the program's KDFX-B outputs. It is also stereo. The incoming signal has a strong treble boost on it. It is sent directly to FXBus2 at unity gain and full width. Play on the flute part of the keyboard, and the arrow next to B flashes.

The other two inputs, C and D, are not assigned to any FXBus.

### AuxFX Bus

Now let's look at the Auxiliary FX Bus (Aux bus). Press **more>**, then the **AUXFX** soft button to look at its page.



Here is an FX preset called **MedWarmPlate**, which is just what it sounds like: a medium-sized, warm-sounding plate reverb. It has two bus overrides, Wet/Dry mix, and Output Gain. The Wet/Dry level is set to 100%, because the reverb can be applied to any of this studio's FXBuses. You set the actual wet/dry mix for these with the Aux and Mix levels on the FXSEND pages. By keeping the reverb level on the Aux bus set to 100%, you avoid mixing in a non-reverbed signal twice.

Since both Insert FXBuses have signal going to the Aux Bus, the arrow next to Aux will flash as long as any signal processing is going on in either of the insert FXBuses.

## Outputs

Finally, press the **OUTPUT** soft button to get to the OUTPUT page. Here we see that the four physical output pairs are each passing different parts of the studio. If the outputs are connected to an external mixer, you can treat each of them separately: recording them on different tracks of a tape deck, sending them to different outboard processors, or mixing them differently in a monitor mix.

```

EditStudio>OUTPUT

Mix Lvl:0.00dB   Output A:Mix
Mix Bal:0%      Output B:FXBus1
                Output C:FXBus2
                Output D:AuxFX

<more> FXSEND AUXFX OUTPUT <more>

```

Output A has the Mix bus. This is the combined output of the two FXBuses, plus the reverb on the Aux bus. Its gain and balance are at unity.

Output B has the output of FXBus1, that is, the ring-modulated piano, without any reverb.

Output C has the output of FXBus2, the delayed / flanged / flute, without any reverb.

Output D has the output of the Aux bus, which is *just* the reverb signal, with no dry component, since the value of Wet/Dry on the AUXFX page is 100%.



*Note: If you're using the Mix audio outputs, keep in mind they carry the summed signals of audio outputs A through D. Normally you would assign each of the audio outputs differently. In the example above, you might set Outputs B through D to **Off**, or Output A to **Off**, to avoid overlapping assignments.*

Here is the overall structure for this studio:

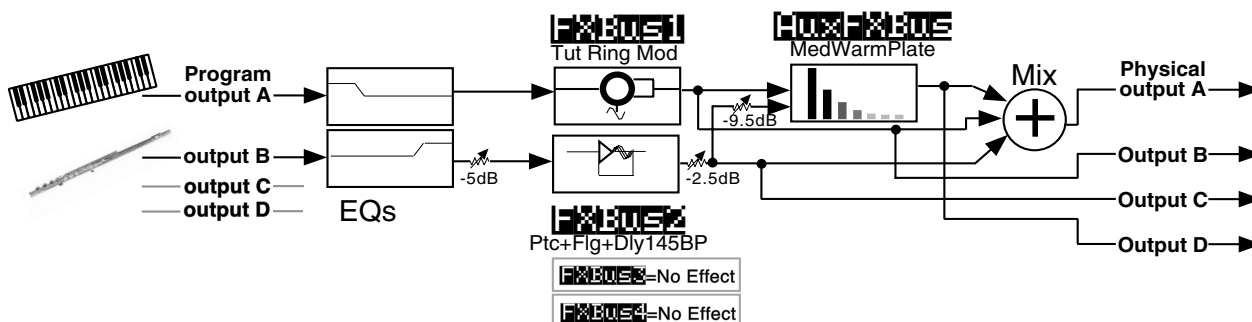


Figure 9-5 Structure of Studio 201

## Chaining Effects

Now we'll see how you can employ multiple effects on a single bus using the chaining feature.

Start in Program Mode by highlighting Program **199 Default Program**, then pressing the **Effects** button. From the KDFXMode:MAIN page, select Studio **199 Default Studio** (highlight the Studio name, then scroll through the list or press **1-9-9-Enter** on the Alphanumeric pad).

With the Studio name still highlighted, press the **Edit** button to enter the Studio Editor. As shown below, you can see that there is an empty effects block on this bus and all 4 PAUs are still available. If you'd like, you can press the **Chan/Bank** buttons to see that while there are other empty blocks (although not on FXBus 4, since it is generally used as a dry bus or a pass-through to the AUX FX), there are no effects on any of the buses in this studio.

```

EditStudio:FXBUS Size:0 Free:4 <>FXBus:1
FX1→[ ]→[ ]→[ ]→[ ]→
FX: 199 No Effect
None      :
None      :
None      :
<more> INPUT FXBUS Add Remove more>
  
```

Now press the **Add** soft button to add another empty effects block to this bus:

```

EditStudio:FXBUS Size:0 Free:4 <>FXBus:1
FX1→[ ]→[ ]→[ ]→[ ]→
FX: 0 None
None      :
None      :
None      :
<more> INPUT FXBUS Add Remove more>
  
```

We'll use a distortion preset for the first effect in this block. Use the cursor keys to highlight the first effects block, then select Preset **763 SubtleDistortion**:

```

EditStudio:FXBUS Size:1 Free:3 <>FXBus:1
FX1→Dist→[ ]→[ ]→[ ]→
FX: 763 SubtleDistortion
Wet/Dry   :100%wet
Out Gain  :-24.0dB
None      :
None      :
None      :
<more> INPUT FXBUS Add Remove more>
  
```

You can see that the effects block now displays an abbreviation that indicates the type of effect, while the lines below show the name of the specific effect preset, as well as the bus overrides defined for this preset. As always, you can select one of the bus overrides and change it to be any parameter of the effect's underlying algorithm. You'll probably want to keep the two overrides here, however, since they offer useful control over the preset: Wet/Dry lets you quickly adjust how much of the signal is effected, while Out Gain lets you boost or cut the audio level of this effect as it enters the next effect in the chain.

We'll now add phaser and reverb effects to the chain:

Use the cursor keys to highlight the second effects block, then select Preset **191 Slow Deep Phaser**:

```

EditStudio:FXBUS Size:1 Free:2 <>FXBus:1
FX1→Dist→Phsr→
FX: 191 Slow Deep Phaser
Notch/Dry :50%wet
None      :
Alloc:Auto
<more> INPUT FXBUS Add Remove <more>

```

Press **Add** again to add another effects block. If the “Out of effect blocks” message displays (and it should), then you'll have to remove an effects block from one of the other buses in this studio. Press the **Chan/Bank** up button to display the screen for FXBus 2; you will see an empty effects block, highlighted for your convenience; press the **Remove** soft button and that block (that you weren't using anyway) is now available for you on FXBus 1.

Press the **Chan/Bank** down button to return to FXBus 1. Use the cursor keys to highlight the **Phsr** block, then press the **Add** soft button. This time an empty effects block will be added to the chain for you. Note that the new block is added to the right of the current block, which is why we highlighted the **Phsr** block before pressing **Add**.

While the third block in the chain (the one you just added) is highlighted, select Preset **104 Cool Dark Place**. Your display will look something like this:

```

EditStudio:FXBUS Size:3 Free:2 <>FXBus:1
FX1→Dist→Phsr→RvrB→
FX: 104 (Cool Dark Place)
In/Out      :In
Out Gain    :-0.5dB
Alloc:Auto
<more> INPUT FXBUS Add Remove <more>

```

The outlined box around the effect block and the parentheses around the preset name indicate that we've chosen an effect that requires more processing power than we currently have available in this studio. Take a quick glance at the top line of the screen and the problem becomes obvious: Cool Dark Place wants 3 PAUs (as shown by the **Size** parameter) but only 2 are available (as shown by the **Free** parameter). KDFX will still let you place the preset on the chain, but it won't be active, so you won't hear the effect on your signal.

Let's see if we can find a reverb that fits in this chain. Press the + cursor button once to have a look at the next listed preset, **105 Gunshot Verb**:

```

EditStudio:FXBUS Size:2 Free:0 <>FXBus:1
FX1→Dist→Phsr→Dlvr→
FX: 105 Gunshot Verb
In/Out      :In
Out Gain    :-0.5dB
Alloc:Auto
<more> INPUT FXBUS Add Remove <more>

```

As you can see, this effect requires 2 PAUs, which is what happens to be available in the Studio. If you decide to use this effect, however, you will not be able to add any more presets to this

chain, since you have now maximized your PAU usage. If you'd like to have four effects on the chain, you'll have to select only effects that use a single PAU. (You'll also have to remove the empty effects block from FXBus 3 so that it will be available for FXBus 1.) Keep in mind, however, that you can still send the output of this chain to an effect on the Aux bus, which allows effects using up to 3 PAUs.

Now let's audition this effects chain: play your keyboard to hear the sound of the **199 Default Program** running through the 3 effects you've chosen in series.

To use this effects chain with any other program, you'll have to save the studio. Press **Exit**, then **Rename**. Enter a name for the studio, then press **OK**. The save dialog now offers you the next available studio number; you can save the studio at this location or choose a different one. Press **Save** and you are returned to the KDFXMode:MAIN page. Press **Exit** to return to ProgramMode. You can now select a program, press **Edit**, followed by the **KDFX** soft button (you'll have to press one of the **more** soft buttons a few times to bring the KDFX soft button into view), and select your new studio for the program's effects. Don't forget to check the program's OUTPUT page to be sure that its output pair is routed to KDFX-A.



## A Complex Studio with Real-Time Control

The final studio we'll look at is a bit more complex, not least because it's under real-time control. Getting real-time control of a studio requires doing some advance work. Most noticeably, you have to create a program or setup that uses a particular studio, then define a set of FXMods in that program or setup. The FXMods provide the settings that link physical controllers such as sliders to studio effects like wet/dry mix.

For all of this to work, the FXCtrl parameter on the KDFXMode:CTRL page must be set to a value of **Program** (if you want only programs to be in charge of studio selection and controls) or **Setup** (if you want only setups to have this ability)—or **Auto** if you want to control over studios whenever you select a program *or* a setup.

In this example, we'll work with *setup* control over a studio—the procedure for working with program control is almost exactly the same.

### Setting the FXCTRL

To enable real-time control of the studio from the setup, we have to put the K2661 in the correct Effects mode. Go to the KDFXMode:CTRL page (press **Effects** followed by the **CTRL** soft button), and set FXCtrl to **Setup**. FX Channel automatically goes to **None**.

```
KDFXMode:CTRL FXCtrl:Auto <>Enable
Studio:199 Default Studio
FXCtrl:Setup
FXChan:None
```

```
MAIN CTRL EQBYP FXBYP BUSMUT Bypass
```

### The Setup

Now let's look at the setup that we're going to use to control our studio. Press **Setup** to go into Setup mode, and select **200 KDFXCombo**.

This is a four-zone setup, with bass and drums at the bottom, electric piano in the middle, and a breathy flute-like sound on the top. Each layer goes to a separate KDFX output pair, so they can all get different processing.

If you edit the setup and look at the various zones, you'll see this:

- Zone 1: **Gtr Jazz Band**, a layered bass and drum program, going to KDFX-A.
- Zone 2: **Dual Slap Bass**, also going to KDFX-A. This and the previous zone are active from the bottom of the keyboard up to A3.
- Zone 3: **Pno & Epno & Pad**, an electric piano and pad program, going to KDFX-B, which is active from A<sup>#</sup>3 to F<sup>#</sup>5.
- Zone 4: **Hybrid Vox**, going to KDFX-C, and active from G5 and up.

### Looking at the Studio

Starting from SetupMode, press **Edit**, then **more>** three times. Now press the **KDFX** soft button to view the studio. The name of the studio associated with this setup, **202\*Complex**, appears.

```

EditSetup:KDFX All Zones
Studio:202*Complex

Bus: Param: Adjust: Source: Depth:
FX1 In/Out Out SoftPd 1
FX1 Aux Lvl -55.0dB MIDI27 52dB
FX2 L Fdbk Lvl 0% MIDI26 100%
<more> KDFX FX100% FX100% FX100% more>

```

Highlight the studio's name and press **Edit**, and let's dig into this studio.

Press **INPUT** if you'd like to look at the EditStudio:INPUT pages. These are all set up straightforwardly, with Input A going to FXBus1, Input B going to FXBus2, etc. Press the **Chan/Bank** buttons to move between the input groups.

### On the FXBus Pages

Press **FXBUS** to look at the FXBuses. On FXBus1 is **205\*CompresHK**, a hard-knee compressor.

```

EditStudio:FXBUS Size:1 Free:1 <>FXBUS:1

FX1 → Cmpr →

FX: 205*CompresHK
In/Out :FXMod
None : Alloc:Auto
<more> INPUT FXBUS Add Remove more>

```

The In/Out parameter shown on this page is a bus override, similar to the ones we've seen in earlier tutorials. However, instead of saying **In** or **Out** it says **FXMod**. This means that this parameter isn't controlled from inside the studio at all—it's controlled by something outside the studio.

As it happens, it's controlled by the soft pedal, MIDI Controller 67—Switch Pedal 3 for K2661 keyboard users. We'll see how this is done in a moment. Pressing this pedal causes the compressor to kick in, squashing the dynamic range of the sound. The MakeUpGain inside the compressor is set to 6.0 dB, however, so the level doesn't change much when the compressor is engaged.

Such control is useful on this program, which uses velocity-switching to change drum sounds. As you play harder, the drum sounds change, but they also get louder, which you may not want. With the compressor engaged, the drum sounds can change *without* getting louder.

There's another FXMod, which is assigned to the Aux bus send level; this is indicated on the EditStudio:FXSEND page (press **more>** followed by the **FXSEND** soft button to have a look). On the Aux bus is a reverb, so this FXMod controls how much of the signal coming through here will go to the reverb. It's under the control of Slider G, MIDI Controller 27. Play the bass and drums and move that slider, and you'll hear the reverb go in and out.

## Setting Up FXMods

How do you set up FXMods? That's back on the KDFX page in the Setup Editor. Exit the Studio Editor (if you have made any changes, don't save them—that will only confuse things!), then press the **KDFX** soft button. Listed here are the FXMods for this setup:

```

EditSetup:KDFX All Zones
Studio:202*Complex

Bus: Param: Adjust: Source: Depth:
FX1 In/Out Out SoftPd 1
FX1 Aux Lvl -55.0dB MIDI27 52dB
FX2 L Fdbk Lvl 0% MIDI26 100%
<more> KDFX FX100% FX100% FX100% FX100% more>

```

The first FXMod affects the signal on the FX1 bus, as shown by the Bus parameter. Param indicates which parameter is affected on the FX1 bus—in this case, the In/Out parameter. The Adjust parameter defines the initial condition of the parameter controlled on this bus—in this case, the In/Out parameter has an initial value of **Out** (disengaged).

The Source parameter determines what MIDI Controller affects the In/Out parameter. For the first FXMod it's **SoftPd** (that's MIDI 67, which is Soft Pedal according to the MIDI specification, and which the K2661 uses as the default destination for Switch Pedal 3). While this setup is current, *any* K2661 controller—be it a wheel, slider, ribbon, or button—that's programmed to send MIDI 67 will affect the In/Out parameter on the FX1 bus of this studio.

The Depth parameter defines how much the soft pedal affects the In/Out value, and here it is 1.

For parameters with binary value, like **In/Out** or **On/Off**, the Depth parameter can have only three values: 1, 0, or -1. If it is set to 1 or -1, then changing the state of the Source (in this case, depressing the soft pedal), changes the parameter's state—in this case, going from Out to In. (If it's set to 0, then the Source has no effect on the value of the parameter.)

Now look at the second FXMod. It sets the Aux Lvl on the FX1 bus to an initial value -55 dB. The Source, **MIDI 27**, can raise that level by as much as 52 dB, to put it at -3 dB. You can hear this in action when you move Slider G (which defaults to MIDI 27) while you play this patch.

## FXBus 2

Let's go back into the Studio Editor and look at FXBus 2. Highlight the studio name on the EditSetup:KDFX page, then press **Edit**; then press the upper **Chan/Bank** button to change buses. Here's our piano and string pad, going through an FX preset called **206\*Fast&RichChorus**.

```

EditStudio:FXBus Size:1 Free:1 <>FXBus:2
FX2 →Chor————→
FX: 206*Fast&RichChorus
None :
None : Alloc:Auto
<more> INPUT FXBUS Add Remove more>

```

## Basic Effects Mode

A Complex Studio with Real-Time Control

If you now go inside the FX Preset (by pressing **Edit**), you'll see that the left and right feedback levels ("Fdbk Lvl") are under FXMod control:

```
editFXPreset:PARAM1 EffectSize:1/1
FXAlgorithm:152 Dual Chorus 1
In Gain :0.0dB
L Wet/Dry :50%wet R Wet/Dry :50%wet
L Out Gain:0.0dB R Out Gain:0.0dB
L Fdbk Lvl:FXMod R Fdbk Lvl:FXMod
Xcouple :0%
<more> PARAM1 PARAM2 PARAM3 more>
```

Press **PARAM2**, you'll see that the Left and Right LFO Rates are *also* under FXMod control.

```
editFXPreset:PARAM2 Dual Chorus 1
L Tap Lvl :75% R Tap Lvl :75%
L Tap Pan :-100% R Tap Pan :100%
L LFO Rate:FXMod R LFO Rate:FXMod
L LFO Depth:5.0ct R LFO Depth:5.0ct
L Tap Dly :4.0ms R Tap Dly :4.0ms
L HF Damp :25088Hz R HF Damp :25088Hz
<more> PARAM1 PARAM3 PARAM4 more>
```

Go back out of the FXPreset and the Studio, to the Setup's **KDFX** page, by pressing **Exit** twice. Here we see that on FX Bus 2, L Fdbk Lvl has a starting value of 0%, and can be changed, using MIDI controller 26 (Slider F), up to 100%.

Press **FXMOD2** to go to the next page of FXMods, and you'll see the same slider changing the R Fdbk Lvl, only in this case the Depth is -100%, meaning the feedback on this channel will be out of phase with the main signal.

```
editSetup:FXMOD2 All Zones
Bus: Param: Adjust: Source: Depth:
FX2 R Fdbk Lvl 0% MIDI26 -100%
FX2 L LFO Rate 0.50Hz MIDI25 8.25H
FX2 R LFO Rate 0.52Hz MIDI25 8.25H
FX2 Aux Lvl -15.5dB MIDI27 15dB
FX3 Fdbk Level 10% MIDI24 89%
<more> KDFX FXMOD1 FXMOD3 FXMOD4 more>
```

Further down the FXMOD2 page are the assignments to the LFO rates: MIDI controller 25 (Slider E) is assigned to the left and right LFO1 Rates, with minimum values of 0.50 and 0.52 Hz, respectively, and maximum change of 8.25 Hz. Set the feedback level high and you can really clearly hear the LFO rate changing (but watch out that the effect doesn't go into oscillation at the highest feedback level).

Finally, MIDI Controller 27, our old friend Slider G, controls the Aux Level send on this bus as well, controlling the amount of this signal that will be sent to the Aux reverb, and thus how much reverb will be applied to the piano/pad sounds. As you can see, you can assign a single Source to multiple FXMods.

### FXBus 3

Press **Edit**, then use the **Chan/Bank** buttons to select FXBus 3. Here's where our breathy flute sound is, and it's going through a delay FXPreset called 207\*Adj Delay.

```

EditStudio:FXBUS Size:1 Free:1 <>FXBUS:3
FX3 → D19
FX: 207*Adj Delay
Wet/Dry :50%wet
Out Gain :0.0dB
Alloc:Auto
<more INPUT FXBUS Add Remove more>

```

Press **Edit** to inside the FXPreset, and you'll see that the Feedback Level is under FXMod control.

```

EditFXPreset:PARAM1 EffectSize:1/1
FXAlgorithm:131 4-Tap Delay
In Gain :0.0dB
Wet/Dry :50%wet Out Gain :0.0dB
Fdbk Level:FXMod
Dry Bal :0%
HF Damping:25088Hz Hold :Off
<more PARAM1 PARAM2 PARAM3 more>

```

Press **PARAM2**, and on that page, you'll see that Delay Scale is also under FXMod control:

```

EditFXPreset:PARAM2 4-Tap Delay
Loop Crs :480ms DelayScale:FXMod
Loop Fine :0.0ms
Tap1 Crs :120ms Tap3 Crs :360ms
Tap1 Fine :0.0ms Tap3 Fine :10.0ms
Tap2 Crs :240ms Tap4 Crs :480ms
Tap2 Fine :5.0ms Tap4 Fine :-5.0ms
<more PARAM1 PARAM3 PARAM4 more>

```

Go back out to the Setup (press **Exit** twice), press the **FXMOD3** soft button, and look at the **FXMOD3** page.

```

EditSetup:FXMOD3 HUI Zones
Bus: Param: Adjust: Source: Depth:
FX3 DelayScale 0.54x Foot 6.00x
FX3 Aux Lvl -50.0dB MIDI27 52dB
None None OFF
None None OFF
None None OFF
<more K0-FX FXMOD2 FXMOD3 FXMOD4 more>

```

On the **FXMOD3** page, we see that the Delay Scale, which scales all of the various delays in the Algorithm, is controlled by the Foot Pedal, MIDI controller 4—Control Pedal 1 for K2661 keyboard users. The delay is nominally a 480 ms loop, with four equal-spaced taps inside it, 120 ms apart. With the Foot Pedal, we can scale all those times by a factor of between 0.54 and 6.00.

Finally, the Aux Level from this bus, controlling another reverb send, is once again assigned to MIDI controller 27, Slider G.

### And in Conclusion...

Here's a summary of the FXMods in our complex studio with real-time control:

- Soft pedal puts compression on the bass and drums.
- Slider F controls the feedback on the chorus on the piano/string pad. Slider E controls the LFO speed on that chorus.
- Slider D controls the feedback on the delay on the breathy flute.
- Foot Pedal controls the delay time on the breathy flute.
- Slider G controls the reverb send for all three zones.

## Building Your Own

Now that we've walked you through a complex studio, here are some hints on how to go about building your own.

First, you want to decide whether you want the studio to be static, or to be dynamic under FXMod control from the K2661 keyboard and/or an external MIDI source. If it's to be static, the value of FXCtrl must be **Master**, while if it's to be dynamic, the value of FXCtrl must be **Program**, **Setup**, or **Auto**, depending on how you want to approach it.

Then you need to look at how your program outputs are arranged, so you can design the studio intelligently. If you're using internal setups, look at the output assignments on the various zones, and change them so sounds that need different effects are separated, and those that can use the same effects are grouped. If you're using a MIDI sequencer, you might want to use the Channel output overrides in MIDI Mode, and arrange your sequence so that the assignment of tracks to FXBuses is determined by the tracks' MIDI channel assignments.

Now set up your studio, assigning FX presets to the program outputs/KDFX inputs. Many of the ROM studios follow a common organizational plan, which might be a good starting point for your studios:

- FXBus 1 contains a relatively simple reverb with a low Size requirement.
- FXBus 2 contains an effect that does not increase the "length" of the sound (that is, no reverb or delay), something like chorus, flange, phaser, distortion, shaper, pitcher, enhancer, EQ, or EQ morpher.
- FXBus 3 contains effects that take up lots of time, such as delays, delays with reverb, or other "Lead" sounds.
- FXBus 4 is dry, since the first three FXBuses have probably used up all the PAUs.
- The AUXFX Bus contains a larger reverb (Size: 2 or 3), a compressor, or a graphic EQ. It can often be used instead of an FXBus reverb, such as the one on FXBus1. If you use it in this way (set the Aux Lvl on FXBus 1 to 0dB or higher), it frees up FXBus1 for use as an Enhancer, Stereo Image, Flanger, etc.

Finally, set up your FXMods in the setup or program you plan to use. If you're using a sequencer, you might want to dedicate a program *just* to the studio and FXMods, with no sound coming from that program.

Keep in mind that FXMods don't always have to be dynamic—they can be used to assign static (Source: OFF) values to the parameters in a studio that are different from the studio's normal parameters. This lets you create and store multiple variations on a studio without making each one a separate studio—when you want to call up the variation, merely call up the program that contains the correct FXMods.

One last reminder: don't forget to save your studio!





# Chapter 10

## MIDI Mode

Press the MIDI-mode button to enter MIDI mode. There are three pages in MIDI mode:

- TRANSMIT (XMIT soft button)
- RECEIVE (RECV soft button)
- CHANNELS (CHANLS soft button)

You'll use these pages to determine what MIDI messages the K2661 transmits, and how it responds to the MIDI messages it receives—as well as how each MIDI channel behaves.

When you enter MIDI mode, you'll see one of the three available MIDI-mode pages. When you exit MIDI mode, the K2661 remembers which page you were on. The next time you select MIDI mode, that page appears.

### The TRANSMIT Page

Press the XMIT soft button, and the TRANSMIT page appears. Use these parameters to control how the K2661 sends MIDI information to its MIDI Out port. These settings to some extent affect the K2661's response to its own keyboard and controllers, but they primarily affect the responses of other MIDI devices that are receiving MIDI from the K2661 on the channel specified with the Channel parameter on this page.

It's important to remember that many of the settings of the TRANSMIT page are in effect only when a *program* is selected, either in Program mode or in Quick Access mode. If a *setup* is selected, in Setup mode or in Quick Access mode, the setup's MIDI settings override the corresponding settings on the TRANSMIT page. The TRANSMIT page looks like this:

```

MIDI Mode: TRANSMIT
CtlSetup: 97 Control Setup
Channel : 2                      ProgChng : On
TransPos: 0ST                    Buttons  : Off
Control : Both                   ChgSetups: KeyUp
VelocMap: 1 Linear
PressMap: 1 Linear
XMIT  RECV  CHANLS  ProgChg  RsetCh  Panic

```

Parameter	Range of Values	Default
Control Setup	Setup list	97 Control Setup
Channel	1 to 16	1
Transposition	±60 semitones	0
Control	Both, MIDI, Local	Both
Velocity Map	Velocity Map list	1 Linear
Pressure Map	Pressure Map list	1 Linear
Program Change	Off, On	On
Buttons	Off, On	Off
Change Setups	Immed, KeyUp	KeyUp

## Control Setup

This is where you select the current control setup, Zone 1 of which sets the physical controller assignments for all programs while you're in Program mode. Refer to Chapters 6 and 7 for more information on the control setup.



***Note:** The control setup is responsible for many of the settings that each had separate parameters on the K2000 and K2500. If you've worked with one of these instruments, you'll need to relearn how MIDI transmission controls are defined. Instead of setting a number of parameters on the MIDI TRANSMIT page, you'll simply select a control setup. Zone 1 of that setup then determines the controller assignments (as specified on the SLIDER, RIBBON, WHEEL, and other pages in the Setup Editor).*

## Channel

This defines which MIDI channel the K2661 uses to transmit MIDI messages. The value for this parameter matches the current MIDI channel displayed on the top line of the Program-mode page. If you change the current MIDI channel while in Program mode, the setting of this parameter changes accordingly, and vice versa.

## Transpose

This parameter affects the transposition that's applied to the MIDI data stream. Adjusting this parameter transposes the K2661's notes, as well as notes on slaves receiving from the K2661. This transposition setting is not overridden when you use Setup mode, but is *added* to the transposition settings for the currently selected setup.

## Control

Here you determine where the K2661 sends MIDI information. A value of **MIDI** sends the MIDI signal to the K2661's MIDI Out port, but not to the K2661 itself. This is also known as Local Control Off.

If you're using your K2661 with a MIDI sequencer and have a MIDI loop (K2661's Out to sequencer's In, and vice versa), you'll need to select a value of MIDI when your sequencer's Patch Thru feature—also known as Play Thru, and Soft Thru—is on. This will prevent the K2661's MIDI signal from looping back on itself, which can cause problems. If you deactivate your sequencer's Patch Thru feature, set the Control parameter's value to **Both**, and the K2661 will play normally. Also, you may want to set the value of the LocalKbdCh parameter to **None** when you have a MIDI loop, because you can have problems with doubled notes and MIDI

overload. You won't have problems, however, as long as the channels transmitted by the K2661 are all different from the incoming MIDI channel.

A value of **Local** disables the MIDI Out port. Use this setting when you want to play the K2661, but not to send any MIDI information to other MIDI instruments (local control only). A value of **Both** (the default) enables you to play the K2661 and send MIDI information from its MIDI Out port.

## Velocity Map—Transmit (VelocMap)

The transmit velocity map affects the way the K2661 sends velocity information to its MIDI Out port. Different maps generate different velocity values for the same attack velocity—that is, they apply different curves to the attack velocities the K2661 receives and remap them to new velocities before transmitting them to the MIDI Out port.

Important: The MIDI velocity maps affect only those MIDI velocity values transmitted via the K2661's MIDI Out port, and are used exclusively to adjust the response of MIDI devices connected to the Out port. If you have a DX7 connected to your K2661, for example, and the DX is distorting, selecting a transmit Velocity Map like **Hard2** should handle the problem. Changing the velocity map on this page does not affect the response of the K2661's sound engine to its own keyboard, or to an external MIDI controller. That's done on the RECEIVE page. See Chapter 18 if you're interested in editing velocity maps.

Also important: Both the transmit and receive velocity maps should be left at values of **Linear** unless you really need to change them. The linear maps give you the most consistent results.

Keep in mind that the setting of the Veltouch parameter in Master mode also has an effect on the transmit velocity map.

## Pressure Map—Transmit (PressMap)

This is like the VelocMap, but it controls the aftertouch values sent by the K2661 to its MIDI Out port. Use this exclusively to adjust the response of MIDI devices connected to the K2661's MIDI Out port. Changing the pressure map on this page does not affect the response of the K2661's sound engine to its own keyboard, or to an external MIDI controller. That's done on the RECEIVE page. See Chapter 18 for information about editing pressure maps.

## Program Change (PChng)

When On, the K2661 sends program change commands to its MIDI Out port when you select programs or setups from the front panel or from your MIDI controller. Select a value of **Off** when you want to change programs on the K2661 but don't want to send program change commands to the MIDI Out port. This parameter doesn't affect the *type* of program change command that's sent; it just determines whether any command is sent at all. (The type of program change command is determined by the settings for three parameters on the CH/PRG page in the Setup Editor.)

## Buttons (Bttns)

If you set the value of the Buttons parameter to **On**, the System Exclusive (SysEx) messages generated by your button presses are sent to the MIDI Out port. This enables you to do two things: control a remote K2661 (or earlier model), and record sequences of programming button presses to a sequencer or SysEx software package.

If you have the MIDI In port of another K2661 (or K2600, K2500, or K2000) connected to the first one's MIDI Out port, the second instrument will respond to every button press on the first instrument, just as if you were pressing the buttons of the second one. Keep in mind that both devices must be in exactly the same state (the same page in the same mode, with identical lists of RAM objects) when you start. Otherwise the button presses you make on the first instrument may execute other functions on the second instrument.

Much more useful is to send streams of button presses to your sequencer. When you dump them from your sequencer back to the K2661, the K2661 responds as if the buttons were actually pressed. This enables you to set up a variety of "macros," which are strings of commands that can be executed all at once by a single initial command. For example, you can record a sequence of button presses that enters Disk mode, selects a specific SCSI device, and loads one or more banks of samples while you do something more entertaining. Again, it's important to keep in mind that the state of your K2661 must be identical to its state when you recorded the sequence of button presses. If you've added or deleted any objects stored in RAM, for example, the sequence of button presses will select different objects when you play back the button press sequence.



***Note:** Make sure this parameter is set to **Off** before you initiate a SysEx dump of any kind. If this parameter is **On** when you start a dump, the buttons you press to begin the dump will also generate SysEx messages.*

## Change Setups (ChgSetups)

This parameter determines the exact timing of setup changes when you select a different setup—either by a normal data entry method or via MIDI program change commands.

Choose **KeyUp** to indicate that you want setup changes to take place only when you've released all currently held notes. Choose **Immed** to indicate that you want such changes to happen immediately when you select the setup.

## The RECEIVE Page

Press **RECV** to select the RECEIVE page, where you define the K2661's response to incoming MIDI signals (with one exception pertaining to Quick Access mode, which we'll explain later).

```

MIDI Mode: RECEIVE
BasicChannel: 1 SysExID: 0
MIDI Mode : Multi SCSI ID: 6
AllNotesOff : Normal BendSmooth: On
ProgChgType : Extended LocalKbdCh: None
VelocityMap: 1 Linear BankSelect: Ctl 32
PressureMap: 1 Linear PowerMode : User
XMIT REC V CHANLS ProgChg RsetCh Panic

```

Parameter	Range of Values	Default
Basic Channel	1 to 16	1
MIDI Mode	Omni, Poly, Multi	Multi
All Notes Off	Normal, Ignore	Normal
Program Change Type	Program Change Type List	Extended
Velocity Map	Velocity Map list	1 Linear
Pressure Map	Pressure Map list	1 Linear
System Exclusive ID	0 to 127	0
SCSI ID	0 to 7	6
Bend Smooth	On, Off	On
Local Keyboard Channel	None, 1 to 16	None
Bank Select	0 only, 32 only, Ctl 0, Ctl 32	Ctl 32
Power Mode	User, Demo	User

### Basic Channel

The basic channel determines which channel will always be available to receive MIDI information. Depending on the MIDI receive mode (below), the Basic channel may be the only receiving channel, or one of several.

### MIDI Receive Mode (MIDI Mode)

The MIDI Mode parameter determines the MIDI receiving capabilities of the K2661. When set to **Omni**, the K2661 responds to incoming MIDI events on all MIDI channels, and plays them on the current channel. This is normally used for diagnostic purposes only.

At a setting of **Poly**, the K2661 responds only to events that are sent on the same channel as the K2661's current MIDI channel (the one displayed on the top line of the Program-mode page). In Poly mode, the currently selected channel is always the basic channel, so if you change channels, the basic channel changes accordingly.

With a value of **Multi** (the default), the K2661 responds to events on all active channels. This is the mode you'll use when you're driving the K2661 with a sequencer, since you can play a different program on each channel. At this setting, you can turn individual channels on and off (on the CHANNELS page, described later in this chapter).

## All Notes Off

If this parameter's value is set to **Normal**, the K2661 responds to All Notes Off messages received over MIDI. **Ignore** causes these messages to be ignored. If you're using a Roland product as a MIDI controller for your K2661, you'll want to set the value of this parameter to **Ignore**. This is because some older Roland products occasionally send an All Notes Off message when no keys are held down—even if you're sustaining notes with a pedal. You might find all your sustains missing from your sequence, for example, if you're driving your K2661 from one of Roland's hardware sequencers. Setting this parameter to **Ignore** takes care of this problem.

Regardless of the setting for this parameter, the K2661 always responds to its own **Panic** button by shutting off all active notes and controllers.

## Program Change Type (ProgChgType)

This determines how the K2661 responds to program change commands received via MIDI. See *Program Change Formats* on page 10-11 for an explanation of the various values available for this parameter.

## Velocity Map—Receive

The velocity map applies a preset curve to incoming velocity messages. It maps incoming velocity levels to new levels that correspond to the eight dynamic levels used by the VTRIGs and keymaps for velocity level selection. See Chapter 18. Normally you'll leave this set to **1 Linear**. Adjust this parameter's value only when you need to alter the K2661's response to the velocity messages from a MIDI controller, for example, if you're getting too much or too little volume when you play, or when you're using a sequencer to drive the K2661.

## Pressure Map—Receive

Like the velocity map, this determines how the K2661 responds to incoming pressure (aftertouch) messages.

## System Exclusive ID (SysExID)

The SysExID parameter differentiates between more than one MIDI device of the same model. You won't need to change the default setting of 0 unless you have multiple K2661s (or K2600s, K2500s, or K2000s) receiving SysEx messages from a single source. In that case, make sure each instrument has a different SysExID. Then you can direct SysEx messages to the appropriate K2661 with the SysExID byte that's included with every SysEx message. A value of **127** specifies "Omni Receive." That is, at this value, a K2661 responds to a SysEx message regardless of the SysEx ID of the message (as long as the manufacturer and device IDs match—see Chapter 7 of the *Musician's Reference* for more information about System Exclusive messages).

## SCSI ID

Use this parameter to change the SCSI ID of your K2661. You can ignore this parameter unless you've connected a SCSI device (external SCSI disk or CD-ROM drive) to the K2661's SCSI port. You can use the SCSI port to chain up to seven SCSI devices to the K2661 (a total of eight devices can be chained together); just be sure to set each one to a different SCSI ID. Most SCSI devices available today make it easy to change their SCSI IDs, so you may not have to adjust this parameter even if you have several SCSI devices connected. See Chapter 13 for more information about using SCSI devices.

## Bend Smooth

This parameter can improve your K2661's performance when you're driving it from a MIDI guitar controller. Its default value is **On**.

You may find that pitch bending seems to carry over from the previous note to the next note, causing it to start on the wrong pitch. This is probably due to the automatic pitch smoothing provided by the K2661. If this is happening, try setting the BendSmooth parameter to a value of **Off**.

## Local Keyboard Channel (LocalKbdCh)



***Note:** Changing the setting of the Local Keyboard Channel parameter is useful only when your K2661 is receiving MIDI information from an external source—maybe you have a favorite MIDI keyboard that you use to control all the gear in your studio, or you use a lot of outboard sequencing. If you're using the K2661 as a standalone music workstation or performance keyboard, you can ignore this parameter.*

The available values for this parameter are **None**, and **1–16**. The default is **None**, which disables the local keyboard feature, since you may not want to send your MIDI controller's MIDI information to devices connected to the MIDI Out port.

It's important to understand that a setup is a control-oriented object, designed to transmit note and control information. The K2661 itself transmits on up to eight channels when in Setup mode. If you are using a keyboard controller that sends information on only one MIDI channel, you need a way to distribute that information from one channel to eight channels. This is what the LocalKbdCh parameter does. It takes the signal coming in via one channel and turns it into different information, depending on where you are in the K2661.

The local keyboard channel changes the way the K2661 performs in other modes as well. It changes the incoming information depending on what you have displayed in the K2661. For example, if LocalKbdCh is **1**, and you're in Program mode with Channel 5 in the display, then the information coming in on Channel 1 gets remapped to Channel 5, and you hear the program assigned to Channel 5. But if you set LocalKbdCh to **None**, then if you send on Channel 1, you hear the program that is assigned to Channel 1, even if you're looking at Channel 5.

Local Keyboard does more than just change the MIDI channel. When Local Keyboard Channel is set to **None**, you will notice that the **Octav–** and **Octav+** soft buttons found in Program, Setup, Quick Access, and Effects modes do not function. If you use the Local Keyboard Channel parameter, however, they function correctly. And you can even use it to change one type of MIDI Controller number to another.

Here's how it works. The K2661 receives MIDI information on the channel that corresponds to the value you set for this parameter, and relays it to its MIDI Out port, using the MIDI channels currently shown in the display. If you're in Program mode (or in Quick Access mode with a program selected), the K2661 relays the LocalKbdCh MIDI information to the channel to which the program is assigned. If you're in Setup mode (or in Quick Access mode with a setup selected), the K2661 relays the LocalKbdCh MIDI information to all the channels currently used by the setup.

The K2661 also remaps certain MIDI Controller messages so that they correspond (in most cases) to the default assignments for the K2661's physical controllers (as listed in Table 10-1). This enables you to receive the messages listed in the first column of Table 10-1 from a MIDI source, and remap them to whatever values are available for the parameters listed in the second column. The third column lists the Studio-editor page where each parameter appears. Note that

when the Large Ribbon is configured in one section, you cannot remap incoming MIDI Controller messages through it.

The MIDI Controller number received from your MIDI source...	...gets sent to the MIDI Controller number assigned as the value for this parameter...	...which is on this page in the Studio Editor
Mono Pressure	Press	PRESS
Modulation (01)	ModWhl	WHEEL
Breath (02)	CPedal2	CPEDAL
Foot (04)	CPedal1	CPEDAL
Data (06)	Slider A	SLIDER
Ctl A (16)	Small Ribbon Position	RIBBON
Ctl B (17)	Small Ribbon Pressure	RIBBON
Ctl C (18)	Large Ribbon Section 1	RIBBON
Ctl D (19)	Large Ribbon Section 2	RIBBON
MIDI 20	Large Ribbon Section 3	RIBBON
MIDI 22	Slider B	SLIDER
MIDI 23	Slider C	SLIDER
MIDI 24	Slider D	SLIDER
MIDI 25	Slider E	SLID/2
MIDI 26	Slider F	SLID/2
MIDI 27	Slider G	SLID/2
MIDI 28	Slider H	SLID/2
Sustain (64)	FtSw1	FOOTSW
Sostenuto (66)	FtSw2	FOOTSW
Soft Pedal (67)	FtSw3	FOOTSW
Legato Switch (68)	FtSw4	FOOTSW
Freeze Pedal (69)	Switch 1	SWITCH
MIDI 70 (70)	Switch 2	SWITCH

**Table 10-1 Controller Remapping Via Local Keyboard Channel**

Here's a more specific example. Suppose your MIDI controller transmits on MIDI Channel 1, and you've set the LocalKbdCh parameter to a value of **1**. You've also set the MWhl parameter on the WHEEL page in the Setup Editor to a value of **Volume** (MIDI 07) for each setup zone. Then you've selected a setup that uses MIDI Channels 1, 2, and 3. When you send a Modulation message (MIDI 01) from the MIDI controller, you'll affect the K2661's volume (unless the VolLock parameter for Channel 1 is on), and the K2661 will send a Volume message to its MIDI Out port, on Channels 1, 2, and 3.

## Bank Select

BankSelect allows you to choose between having the K2661 respond to Controller 0 or Controller 32 or both. The reason for this is that various manufacturers have chosen one method or the other. The four possible values for this parameter are:

**0 only** Responds to controller 0 only.

**32 only** Responds to controller 32 only.



**Ctl 0**      Responds to 0 or 32.

**Ctl 32**     Responds to 0 or 32.

## Power Mode

Power Mode has two possible values: **User** and **Demo**. When set to **User**, the user's parameter settings are retained. When set to **Demo** several parameters are returned to default values when the unit is powered up. The default value for the Power Mode parameter is **User**.

The following parameters are reset when Power Mode is set to Demo.

Master mode:            KB3 Channel resets to **1**

MIDI Transmit:         Control resets to **Both**; Channel resets to **1**; Transpose resets to **0 ST**

Effects mode:          FX Mode resets to **Auto**, and FX Chan resets to **Current**

Disk mode:             Current Disk resets to **SMedia**

## The Channels Page

Press the **CHANLS** soft button to select the CHANNELS page, where you can define numerous parameters for each MIDI channel independently. Use the **Chan/Bank** buttons to select the MIDI channel you wish to work on.

The CHANNELS page is very useful when you're doing multi-timbral sequencing, with programs assigned to numerous MIDI channels. The CHANNELS page lets you set several control characteristics for each MIDI channel. This makes it easy to adjust the playback of the sequence without editing the sequence itself. For example, you might turn off the Enable parameter for one or more channels to mute the tracks on those channels. You could also set the VolLock parameter to **On**, to ignore any MIDI volume messages the K2661 receives on a given MIDI channel.

```

MIDI Mode: CHANNELS <> Channel: 2
Enable : On
Program: 1 Concert Piano 1   ProgLock: Off
Pan    : 64                 PanLock: Off
Volume : 127                VolLock: Off
OutPair: Prog
OutGain: Prog
XMIT   REC   CHANLS ProgChg RsetCh Panic

```

Parameter	Range of Values	Default
Enable	Off, On	On
Program	Program list	Program ID 1
Pan	0 to 127	64 (centered)
Volume	0 to 127	127 (maximum)
Output Pair	Prog, KDFX-A to KDFX-D	Prog
Output Gain	Prog, -12 to 30 dB in 6 dB increments	Prog
Program Lock	Off, On	Off

Parameter	Range of Values	Default
Pan Lock	Off, On	Off
Volume Lock	Off, On	Off

## Enable

Use this parameter to turn the currently selected channel on or off. When on, the channel will receive MIDI information, and the settings of the parameters on the MIDI CHANNELS page will be in effect. When off, the channel will ignore all MIDI information.

## Program

Use this parameter to assign a program to the currently selected channel. The channel will still respond to program change commands received via MIDI, unless the PrgLock parameter (described below) is set to **On**.

## Pan

This offsets the pan position of the current program as set on the OUTPUT page in the Program Editor. A value of **0** is maximum offset to the left, **64** is no offset, and **127** is maximum offset to the right. Changing the value of this parameter is like inserting a MIDI pan message. MIDI Pan (MIDI 10) messages will change the value of this parameter, unless the PanLock parameter (described below) is set to **On**.

If the Mode parameter on the OUTPUT page in the Program Editor is set to Fixed, changing the value of Pan on the CHANNELS page in MIDI mode has no effect.

## Volume

This sets the volume for any program assigned to the currently selected channel. A value of **0** is silence, and a value of **127** is full volume. The value of this parameter will change in response to MIDI Volume (MIDI 07) messages, unless the VolLock parameter (described below) is set to **On**.

## Output Pair (OutPair)

This parameter sets the audio output group for the program assigned to the currently selected channel. The default value of **Prog** means that the output group is determined by the program's value for the Pair parameter on the OUTPUT page in the Program Editor. In this case, the channel's output group changes depending on the program assigned to it, with the output group being routed on a per layer basis within the program. Values of **KDFX-A**, **KDFX-B**, **KDFX-C**, or **KDFX-D** fix the output group regardless of the program that's assigned to the channel.

## Output Gain (OutGain)

OutGain boosts or cuts the level at the audio outputs for any program assigned to the currently selected channel. This allows you to make a program louder or softer without having to edit the program.

## Program Lock, Pan Lock, Volume Lock

When the parameter locks are set to **On**, the three parameters they control do not respond to their respective MIDI controller messages. In that case, you could change the Program, Pan, and Volume settings from the front panel, but not via MIDI.

## Program Change Formats

The K2661 can store more programs than the MIDI program change specification can handle (MIDI lets you send program change numbers from 0 to 127 or 1 to 128 only). So we've designed a system that makes program selection more flexible. This is true whether you're selecting programs from the K2661's front panel, or via MIDI.

Program Change Type	For Use With:
Extended	Other K2661s (or K2600s, K2500s or K2000s) similarly set, plus all other instruments that use the Bank Change controller
Kurzweil	K1200s, and 1000s with version 5 software
0–127	Older MIDI devices that transmit program change commands in the range from 0–127 only.
QA Bank E	Other K2661s (or K2600s, K2500s or K2000s) similarly set, when in Quick Access mode
QA Bank K	K1200s and v5 1000s, when in Quick Access mode
QA 0–127	With the K2661 in Quick Access mode, when using it with older MIDI devices

First of all, the K2661's programs (and all of its objects) are numbered and grouped according to a decimal system, that is, in multiples of ten. This is much easier to keep track of than the binary-oriented groupings of many synths, which feature banks of 8, 16, or 64 programs.

Next, the K2661 gives you 999 program change numbers to work with. These are organized into ten banks of 100 each (the memory banks). A program's object ID is its program change number, as discussed on page 5-2. This makes it easy to keep track of your programs. The K2661 can use several different formats for interpreting program change commands. The value for the ProgChgType parameter on the RECEIVE page determines which format is used, and the one you should select depends on your MIDI system.

If you expect you'll always change programs from your K2661's front panel, you can finish this paragraph and skip the next few sections. In this case, selecting programs is as simple as entering the program change number (the program's object ID) on the alphanumeric pad, and pressing **Enter**. Even program numbers above the usual MIDI limit of 127 can be selected this way.

## Extended and Kurzweil Program Change Types

In the early days of MIDI, most instruments had small numbers of memory locations, usually 32, 64, or 128. As instruments began to have more memory locations, however, users ran against the limitation of only 128 values for program changes in the MIDI spec. Because of this, Bank Change Controller was added, allowing users to switch between banks of up to 128 programs per bank.

Previous to the addition of the Bank Change Controller, Kurzweil had developed their own method of switching banks by using two program changes, one to switch the bank, the second

to call up the program within the bank (as described below). The K2661 can respond to either the Bank Change controller or the double-program-change method. In a nutshell, the difference between the **Extended** setting and the **Kurzweil** setting is this: In **Extended**, the K2661 will receive and respond to the Bank Change controller. When set to **Kurzweil**, the K2661 will receive only the double-program-change method.

### Extended Program Changes

If you're controlling your K2661 from a MIDI device that can handle the MIDI Controller 0 or 32 program-change format, you'll have the greatest flexibility if you set the ProgChgType parameter to a value of **Extended** (or **QA Bank E**, but that explanation comes later).

When you're using the extended program change format, then depending on the value of the BankSelect parameter on the RECEIVE page in MIDI mode, the K2661 will respond to either MIDI Controller 0 or 32 program change commands for bank selection (Zeros through 900s), and standard program change commands for program changes within the current bank. Different values have different results, as shown in the following table:

Program Change Command Type	Value of Message	Result
MIDI controller 0 or 32 (MC 0 or MC 32)	0 to 9	Selects memory bank zeros–900s
	10 to 127	Ignored
Standard (PCH)	0 to 99	Selects correspondingly numbered program in current memory bank
	100 to 127	Selects correspondingly numbered program in next-highest bank

If your K2661 is already in the memory bank you want to use, you can send it single PCHs from 0 to 99, to select programs within that memory bank. The K2661's response depends on the setting for the MIDIBankMode parameter on the CH/PRG page in the Setup Editor. If you want to change the memory bank, the K2661 must receive either an MC 0 or 32 message with value 0–9. The next PCH in the range 0–99 will select the correspondingly numbered program in the newly selected bank. The following table of examples should help make it clear.

Bank Change Command Received	Program Change Command Received	Result
MC 0 or 32: value 0	PCH: value 99	Program 99 (0s bank, 99th program)
MC 0 or 32: value 1	PCH: value 42	Program 142 (100s bank, 42nd program)
MC 0 or 32: value 1	PCH: value 120	Program 220 (200s bank, 20th program)
MC 0 or 32: value 9	PCH: value 0	Program 900 (900s bank, 0th program)
MC 0 or 32: value 9	None	900s bank selected, no change in current program (bank selection is pending for next PCH)
MC 0 or 32: value 10	PCH: value 99	MC 32 message ignored; 99th program in current bank selected (for example program 199 if in 100s bank)

### Kurzweil Program Changes

When you use the Kurzweil program change format, you'll use PCH messages to select different memory banks, followed by a second PCH command to change the program within the current

bank, as the following examples demonstrate. You'll want to use this format if you're controlling your K2661 from a Kurzweil 1000- or 1200-series instrument.

1st Program Change Command Received	2nd Program Change Command Received	Result
PCH: value 39	None	39th program in current bank selected
PCH: value 99	PCH: value 27	27th program in current bank selected (99 is selected, then overridden by 27)
PCH: value 102	PCH: value 16	Program 216 (200s bank, 16th program)
PCH: value 105	PCH: value 44	Program 544
PCH: value 109	PCH: value 0	Program 900
PCH: value 127	PCH: value 99	99th program in current bank (1st PCH is ignored, since it's above 109)
PCH: value 127	PCH: value 104	No change in current program; 400s bank is selected pending next PCH

## 0-127 Program Change Type

You may be controlling your K2661 from an “old” MIDI device—one that was built before the MIDI Controller 0 program change format was developed. If your MIDI controller is one of these (if its manual doesn't mention MIDI Controller 0 program changes, it's an “old style” machine), you might want to set the ProgChgType parameter to a value of **0–127**. This will enable you to select programs 0–127 from the controller. This limits your range of program selection, but it configures the K2661 to respond predictably to the controller. (You'll have to select higher-numbered programs from the K2661's front panel) Of course, you could use the Kurzweil format, but in many cases you'll have to send two program change commands to get the program you want.

## Quick Access Banks—Extended (QA Ext)

Using this setting is similar to using the Extended program change format, but it goes one step further. Incoming program change commands are interpreted just as they are in the normal Extended format. But the resulting program change number, instead of selecting a program, selects a Quick Access bank entry (you must be in Quick Access mode for this to work). There are two advantages to using this format. First, it allows you to select both programs and setups using program change commands, without having to switch between Program and Setup modes. Second, you can remap incoming program change commands to select programs or setups with different IDs. This is handy if the sending unit can't send program change commands higher than 127.

First, a brief review of Quick Access bank structure. Each Quick Access bank can store ten entries, each of which can be a program or a setup. Each of the K2661's 10 memory banks can store 20 Quick Access banks (except the Zeros bank, which can store 75). Therefore when you're in Quick Access mode, you have access to 200 (or 750 in the Zeros bank) programs or setups without leaving the currently selected memory bank. The QA Ext program change format lets you select any one of those programs or setups via MIDI. If you select another memory bank, you have a different set of 200 programs and setups at your disposal.

## MIDI Mode

### Program Change Formats

---

When you're using this format, the K2661 will respond to MC 0 or 32 messages for selecting QA banks, and to PCHs for selecting entries within the current bank. PCHs select entries according to their "chronological" listing within the QA bank (not according to their IDs).

Command Type	Range of Values	Result
MIDI controller 0 or 32 (MC 0 or MC 32)	0 to 7	Selects QA bank 0n, 1n, 2n, 3n, 4n, 5n, 6n, 7n in current memory bank
	8–127	Ignored
Standard (PCH)	0–99	Selects last digit (n above) of QA bank, and entry within that bank
	100–127	Ignored

Depending on the QA bank entry you want to select, you'll send the K2661 either a PCH (value 0 to 99), or a MIDI Controller 0 or 32 message (value 0 to 7) followed by a PCH. Sending a single command will let you select from a range of 10 QA banks and select an entry within that bank (see the table below). To select a different range of QA banks, send an MC 0 or 32 message followed by a PCH.

The MC 0 or 32 messages selects the range of QA banks (0s through 70s), while the PCH selects the bank within that range, as well as the entry within that bank. Neither the MC 0 or 32 nor the PCH selects a different *memory* bank (Zeros through 900s). In fact, you can't change the memory bank via MIDI when using this format. All program and setup selections are made within the currently selected memory bank. You'll know which memory bank is selected by looking at the ID of the currently selected Quick Access bank in the top line of the Quick Access-mode page. Several examples follow.

### If the Zeros Memory Bank is Currently Selected

Bank Range Command	Bank / Bank Entry Command	Resulting Selection
MC 0 or 32: value 0	PCH: value 6	No change (K2661 interprets this as QA bank 0, entry 6. There is no QA bank 0. Lowest valid PCH value in this case is 10, which would select QA bank 1, entry 0)
None	PCH: value 9	Entry 9 in current QA bank
MC 0 or 32: value 0	PCH: value 32	QA bank 3, entry 2
MC 0 or 32: value 1	PCH: value 4	QA bank 10, entry 4
MC 0 or 32: value 1	PCH: value 28	QA bank 12, entry 8
MC 0 or 32: value 2	PCH: value 44	QA bank 24, entry 4

Remember that in the Zeros memory bank, the Quick Access bank IDs go through 75. So if the Zeros memory bank is the current memory bank, you can send MC 0 or 32 values as high as 7 for the bank range command. And you can send PCH values as high as 99 for the bank/bank entry command. (When you're in the other memory banks, you can send MC 0 or 32 values of 0 or 1, and PCH values of 0 to 99.)

### If the 200s Memory Bank is Currently Selected

Bank Range Command	Bank / Bank Entry Command	Resulting Selection
None	PCH: value 44	QA bank 204 or 214; entry 4
MC 0 or 32: value 0	PCH: value 6	QA bank 200, entry 6
MC 0 or 32: value 0	PCH: value 32	QA bank 203, entry 2
MC 0 or 32: value 0	PCH: value 99	QA bank 209, entry 9
MC 0 or 32: value 1	PCH: value 4	QA bank 210, entry 4
MC 0 or 32: value 1	PCH: value 28	QA bank 212, entry 8
MC 0 or 32: value 2	PCH: value 44	No change; MC 0 or 32 value 2 is invalid in 200s bank.

## Quick Access Banks—Kurzweil (QA Kurz)

This works almost exactly like the QA Ext format. The only exception is that within the QA Kurz format, the K2661 expects the bank range command to be a PCH, and not MC 0 or 32. MIDI Controller 0 or 32 messages are not recognized. The K2661 expects to receive PCHs of value 0–99 to select a bank and entry, or a pair of PCHs, the first having a value of 100–107 to select a different 10-bank range.

### QA 0–127

Finally, there's the QA Bank format for use with older MIDI devices (program change commands 0–127 only). It works similarly to the other QA formats, but the allowable range of values is limited to 0–107.

## QA Formats and MIDI Transmission

If you're in Quick Access mode and you're using one of the QA formats for the program change type, selecting QA banks or bank entries from the K2661 (with the alphanumeric buttonpad, the cursor buttons, the Alpha Wheel, the **Plus/Minus** buttons, or the **Chan/Bank** buttons) also sends corresponding program change commands to the K2661's MIDI Out port.

The nature of these commands depends on the value of the ProgChgType parameter. The K2661 sends either an MC 0 or 32 message followed by a PCH (when ProgChgType is **QA Ext**), or a pair of PCHs (when ProgChgType is **QA Kurz**) or a single Program Change command (when ProgChgType is **QA 0–127**). The following tables give specific examples.

Current QA Bank	Entry From Alphanumeric Pad	Commands Sent	
		MC 0 or 32	PCH
1	0	0	10
1	9	0	19
2	0	0	20
2	9	0	29
9	9	0	99
10	0	1	0
19	9	1	99
20	0	2	0
29	9	2	99
75	9	7	59
100	0	0	0
105	9	0	59
110	9	1	99
117	7	1	77
119	9	1	99

**Table 10-2 QA Extended Program Change Examples**

Current QA Bank	Entry From Alphanumeric Pad	Commands Sent	
		MC 0 or 32	PCH
1	0	100	10
1	9	100	19
2	0	100	20
2	9	100	29
9	9	100	99
10	0	101	0
19	9	101	99
20	0	102	0
29	9	102	99
75	9	107	59

**Table 10-3 QA Kurz Program Change Examples**



Current QA Bank	Entry From Alphanumeric Pad	Commands Sent	
100	0	100	0
105	9	100	59
110	9	101	99
117	7	101	77
119	9	101	99

Table 10-3 QA Kurz Program Change Examples

## The Soft Buttons in MIDI Mode

The first three soft buttons select the three MIDI-mode pages. The **PrgChg** soft button lets you send a program change command on any MIDI channel. The **RsetCh** soft button lets you return all channel parameters to their default values. The **Panic** soft button sends an All Notes Off and an All Controllers Off messages to the K2661 and on all 16 MIDI channels.

### Program Change (PrgChg)

When you press this soft button, a dialog appears:

Send Program Change:

On Channel 2, Send Program 0

Chan- Chan+ Prog- Prog+ Send Cancel

This dialog lets you send program changes out the MIDI Out port, but does not change internal programs.

The **Chan/Bank** buttons, the **Up/Down** cursor buttons, and the **Chan-** and **Chan+** soft buttons can all be used to change the channel on which the program change command will be sent. The **Left/Right** cursor buttons, the **Plus/Minus** buttons, the Alpha Wheel and the **Prog-** and **Prog+** soft buttons can all be used to change the program change number that will be sent. When you've set the channel and the program change number, press the **Send** soft button to send the program change command. Or press the **Cancel** soft button if you don't want to send it.

You can change the channel and the program number as many times as you want before you press **Send**. You also can use the alphanumeric pad to select a program number directly.

## Reset Channels (RsetCh)

When you press this soft button, the K2661 will ask you if you want to reset all channels, and a pair of **Yes/No** soft buttons will appear. If you press the **Yes** soft button, all settings on the CHANNELS page will return to their default values. For example, you may have set several MIDI channels to route their audio to Output Group B for a special project. When the project's over, you can reset the Channels to restore the audio routing to each individual program (a value of **Prog**), rather than selecting each channel's page and setting the Pair parameter back to a value of **Prog**. Press the **No** soft button if you decide not to reset the channels.

## Panic

This soft button sends an All Notes Off and All Controllers Off message both to the K2661 and over all MIDI channels.

# Chapter 11

## Master Mode

Press the **Master** mode button to enter Master mode, which contains parameters affecting the K2661'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 K2661, 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.

```
Master      Samples:8192K  Memory:485K
Tune       :0ct
Transpose  :05T
KB3Chan    :1
VelTouch   :1 Linear
PressTouch:1 Linear      Confirm :On
Intonation:1 Equal      IntonaKey:C
Object Delete Util Sample GM 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 K2661 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 K2661's notes, and not notes sent via MIDI.

## Transpose

Like the Tune parameter above, Transpose affects every K2661 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)

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 K2661 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 K2661'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 K2661 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 Flat 7th	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.
18–24	Obj vn.n	Not an intonation table; indicates version number of K2661 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 K2661 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<sup>#</sup>** will be a quartertone flat relative to equal intonation. If you change the intonation key to **D**, then **D<sup>#</sup>** 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 K2661 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 K2661 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:

```
Master2  GM Samples:8192K  Memory:485K
Vocoder   :Off
View Mode :List
Digital Output Format: AES
Digital Output Length: 24 Bit
Reset   MAST1
```

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 K2661 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 K2661 to the right mono jack of the Insert cable.

#### ***Using the K2661 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 K2661 to your mixer or amp. Don't use the Mix outputs.

#### ***MIDI***

If your external slave is a rack (or it is a keyboard but you want to use the K2661's keyboard to control the slave), connect a MIDI cable from the MIDI Out port of the K2661 to the MIDI In port of the slave.

### Setting Up the K2661

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 K2661 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 provided on the CD-ROM and SmartMedia card. 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 K2661 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 K2661 as the slave source), pitched to the note or chord you are playing.

Try moving Sliders A, B, and C, and listen for changes in the sound. 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 K2661 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 K2661.

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 K2661 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, but won't play a K2661 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 K2661 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 K2661, 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 K2661. 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 K2661 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 K2661 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 K2661 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 K2661 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.

## **View Mode**

Change the View Mode parameter on the MAST2 page to **Large** to set the view mode to large format; change it to **List** to return to 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.

## Digital Output Format

Change this parameter to set the K2661's digital output stream to your preferred format. The default is ADAT 8-channel digital. Other values are AES Pro, also known as AES/EBU, and AES Cons, also known as S/PDIF. This global parameter is remembered across power cycles, and is not part of any setup or KDFX studio.

To use ADAT In, the K2661's ADAT Out cable must be connected to the sending device. K2661 must be the "master," and the other device(s) must "slave" to it. Output sample rate (and therefore input as well) is fixed at 48 kHz.

## Digital Output Length

Change this parameter to set the digital word length of the K2661's digital output stream. The default is 24 Bit, which is preferred for most applications. Reducing the bit length will reduce the dynamic range of your sound and may increase audible noise. However, some older digital equipment may not be able to work with 24 bit data, and you may have more satisfactory results by reducing the word length at the K2661 output stage. This global parameter is remembered across power cycles, and is not part of any setup or KDFX studio.

If you are resampling digitally, you should be sure to set Digital Output Length to 16 bits. This is because the sampler is only 16 bits, and you want to make sure that the signal is properly dithered before resampling.

When you change the digital word length with this parameter, the signal going to the analog outs is affected, too. For this reason, if you are only using the analog outputs you should keep Digital Output Length set to 24 Bit (the default).

# The Soft Buttons in Master Mode

## Object

This soft button brings up the object utilities. They're described beginning on page 11-15.

## 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 K2661'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 11-15).

When you press the **Voices** soft button, the display shows the K2661's active voice channels as you play. Blocks of capital Xs in six columns of eight represent the 48 notes that the K2661 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 K2661 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 K2661 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.

## Sample

Press the **Sample** soft button to enter the K2661's sampler. Refer to Chapter 14 for complete information on the sampler.

## GM

Press the **GM** soft button to call up the GM page:

```
Master:GM
GM      :On
GM Studio :500 GM1 Room
```

**MAST1**

Enable GM Mode by setting the GM parameter to On. When GM is on, **GM** is displayed on the top line of the screen. This happens on all of the K2661's main pages.

The GM Studio parameter sets the FX Studio that will be used by GM programs. Six GM Studios are included, providing different reverb and chorus-plus-reverb settings.

```
500  GM1 Room
501  GM2 Chamber
502  GM3 Hall
503  GM4 Chorus+Room
504  GM5 Chorus+Chmbr
505  GM6 Chorus+Hall
```

The default GM Studio (GM1 Room) is at ID 500. Use the GM Studio parameter to select a Studio which best complements your program material.

Here's what happens when you enable GM Mode:

- On all channels except channel 10 (which GM uses for drums), you will see only the 128 GM programs. On MIDI channel 10, you will see the eight drum kits.
- The K2661 will modify the following entries in the master table:
  - FX mode (GM uses Master mode)
  - FX channel (GM uses None)
  - FX studio (GM uses the studio selected in GM Studio set on the Master: GM page)
  - Receive velocity map (GM uses the GM Receive Velocity Map)
  - progChgType (GM uses 0-127 mode)

Old settings will be remembered, however, so that when you turn GM Mode off the K2661 will restore your previous settings.

- Volume and expression controllers are mapped to a special GM curve, as in "GS" synths. (GS is a superset of General MIDI that is used by the Roland Sound Canvas and other products.)
- GM drum kits are mapped across program number space as in the "GS" synths, and have exclusive zones included with them.

- Program changes sent to the K2661 when it is in GM Mode will only select programs from the GM program set.



*Setups, Songs, and QA Banks created outside of GM Mode will not point to the correct programs within GM Mode (although you may find the results “interesting”).*

*Similarly, Setups, Songs, and QA Banks created within GM Mode will not point to the correct programs when you leave GM Mode. For this reason, when you create Setups, Songs, or QA Banks within GM Mode you may want to append the letters “GM” to the object’s name and/or store the objects only in certain banks.*

## Reset

This button is on the MAST2 page. Press the **Reset** soft button if you want to return your K2661’s memory to the state it was in when you bought it.



**Caution!** The K2661 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 K2661 will return to the Program-mode page.

## Guitar/Wind Controller Mode

If you are using a wind controller or guitar controller with your K2661, 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 K2661 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 K2661 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 K2661 under some circumstances.

Restarting the K2661 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 13-25. 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 13-35) 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).

## 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:Off

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 K2661, 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 3-7 for details.

## Delete

The Delete Objects utility is very useful for reclaiming unused object and sample RAM in your K2661. 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 K2661. 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 K2661 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 K2661’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.

# Chapter 12

## Song Mode and the Song Editor

### Getting Started with the Sequencer

The K2661's sequencer is a powerful and versatile tool for songwriters, composers, and anyone else who wants 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 K2661. Read through this section, however, to learn about the features that make the K2661'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 (that is, music) on a single 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-mode RECEIVE page (described on page 10-7). Local Keyboard Channel is especially important whenever the K2661 is going to receive MIDI information from an external source, since this enables you to record on different tracks without constantly switching transmit channels on your controller.

- Set the Local Keyboard Channel on the MIDI-mode RECEIVE page to a specific channel (1–16).
- Set your MIDI controller (keyboard, percussion controller, etc.) 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 recording track) as you scroll through the different recording tracks in the K2661's Song mode. Local Keyboard Channel performs a *rechannelizing* function that makes this happen.

#### Patch Through

There's one more use for Local Keyboard Channel. With any model of the K2661, 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 K2661 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 the programs such as the following:

- **54 Jazz Kit II**
- **30 Warm Bass 1^2**
- **22 Gospel Organ**
- **79 Modulead**

Set up your K2661 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.

If you use a KB3 program in a song, make sure that the channel to which you assign it is the KB3 channel; otherwise it won't play. You could always change the KB3 channel to match the channel you want to use for recording the KB3 program, but we recommend deciding on a channel that will always be the KB3 channel, and keeping it that way. Also keep in mind that KB3 programs require one voice of polyphony for every two tone wheels in the program. Since most KB3 programs use at least 79 tone wheels, that leaves only eight voices free for other programs.

Aside from the KB3-channel issue, it doesn't really matter which channel you use for the programs you want to record. In this example, we're going to put the drums on Channel 1 and the bass, organ, and lead on 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 display will look something like this:

Recording track is set to Track 1

```

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

```

Notice that the sequencer is ready for you to record a new song, and the recording track (RecTrk) is set to track 1. If **1 NewSong** doesn't appear as the value for the CurSong parameter, press the **MISC** soft button, then press the **New** soft button on the MISC page. You'll return to the MAIN page, and CurSong will say **1 NewSong**.

2. Use the **Down** cursor button to move the cursor to the Program parameter.
3. When this parameter is highlighted, type **54** then press the **Enter** button. You've now assigned **Jazz Kit** to channel 1.

```

SongMode:MAIN  Events:186K  STOPPED
CurSong:1  NewSong          Tempo:100
RecTrk :1      Vol:127 Pan:64  Mode :Erase
Program:54 Jazz Kit II       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

```

4. Press the **Up Chan/Bank** button. This changes the RecTrk parameter to 2, but leaves the Program parameter highlighted. Notice how the R in the Track region moves from Track 1 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, and so on.

Incidentally, when you're assigning programs on this page, you could use the **Up/Down** cursor buttons to highlight RecTrk, then change the recording track and select the Program parameter again, but the **Chan/Bank** button method is more convenient.

5. On Recording Track 2, select Program **30 Warm Bass 1^2**.

```

SongMode:MAIN  Events:186K  STOPPED
CurSong:1 NewSong          Tempo:120
RecTrk :2      Vol:127 Pan:64  Mode :Erase
Program:30 Warm Bass 1^2    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

```

6. Repeat the above two steps to assign 22 **Gospel Organ** to Channel 3 and 79 **Modulead** to Channel 4.

You've now chosen the programs for your first sequence. It's important to realize, though, that you have not recorded anything yet. The programs 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. Later on, we'll "unloop" the drum track for additional recording.

The length of the drum loop is determined by the current *endpoint*, so we'll start by recording four 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:186K  REC READY
CurSong:1 NewSong          Tempo:120
RecTrk :1      Vol:127 Pan:64  Mode :Erase
Program:54 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 K2661's built-in metronome begins clicking, and the Song-mode LED blinks in time with the current tempo.

Notice the Locat parameter on the right side of the display, which shows the current Bar and Beat number. When you begin recording, the K2661 provides you with a four-beat countoff, during which time Locat's Bar and Beat number are preceded by a minus sign.

You can change the length of the countoff by changing the value of the Countoff parameter on the MISC page.

Since we are recording four measures of silence, press the **Stop** soft button as soon as Locat reads **5:1**.



***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 OK. 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 if you want to save this song. Even though you've just recorded four measures of silence, go ahead and save it; this makes it an official song object. By the way, it's pretty easy to rename the song at this stage. Just press the Rename soft button during the Save Song dialog

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 parameter other than the Program parameter, then press the **Edit** button. The COMMON page appears. (If the Program parameter had been highlighted when you pressed **Edit**, you would have entered the Program Editor, which is not what you want to do right now.)

```

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

```

4. Now press the **EVENT** soft button to bring up the Event Editor, which looks something like this:

```

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

1:1      1:1.000 CTRL BANK  0
1:1      1:1.000 PCHG   54
1:1      1:1.000 CTRL VOL  127
1:1      1:1.000 CTRL PAN  64
Cut      Copy  Paste  View  AllOn  Done

```

The Event Editor gives you access to an editable list of all note events, controller events, 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 records, even when no notes are played.

5. Use the Alpha Wheel to scroll to the bottom of the event display.

The last event listed is the endpoint, which should correspond to the first beat of the Bar following the last Bar in the song. 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.)

6. 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** cursor 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 the **Done** soft button to return to the COMMON page in the Song Editor. If you changed the endpoint while in the Event Editor, save the song now (you may have to press one of the **more** soft buttons to see the **Save** soft button).

7. Press **Exit** to return to the MAIN page in Song mode.
8. Make sure that the RecMode and PlayMode parameters are set to **Loop**.

From the MAIN page, press the **MISC** button. Set the RecMode parameter on the 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 & PlayMode parameters

Quantize parameters

```

SongMode:MISC Events:100% Stop:10
RecMode : Loop Quant:Off CountOff:1
PlayMode: Loop Grid :1/16 Click :Rec
KeyWait : Off Swing:0% ClickCh :16
Locate : 1:1 Sync :Off ClickPrg:198
AutoIn : 1:1 Clock:Int ClickKey:C 4
AutoOut : 1:1 Tempo:Auto ClickVel:100
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 if you're having a bad rhythm day. When you quantize a track, the sequencer moves the elements of that track closer to a grid based on the time signature of the song. You can use quantization to tighten up a rhythm track subtly, or to create a precise, unwavering mechanical rhythm.

For our drum loop, we'll try the total quantization experience, so position the cursor over the Quant parameter, and turn the Alpha Wheel until the value is set to **100%**. Move the cursor down to the Grid parameter. The default value of **1/16** indicates that quantization will move the notes you play to the closest 16th-note division in the Bar. Try double-pressing the **Plus/Minus** 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**, so return to this value if you've changed it, then press **Exit** or the **MAIN** soft button to return to the MAIN page in Song mode.



***NOTE:** The K2661'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 TRACK page in the Song Editor See page 12-48.*

10. Make sure you are in Merge mode (it's the default, so you'll be in Merge mode unless you've changed the value of the Mode parameter). This is important, because you want to be able to overdub on the track as it loops. (In Erase mode, you would erase all existing notes every time the loop came around.) To activate Merge mode, go to the MAIN page and set the Mode parameter to a value of **Merge**.
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 countoff 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 voice 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 MAIN page.

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

Save 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 RecMode to **Unloop** while leaving PlayMode set to **Loop**. This means that the drum loop will keep playing while we record new unlooped 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 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 :Off   ClickPrg:198
AutoIn  : 1:1   Clock:Int  ClickKey:C 4
AutoOut : 1:1   Tempo:Auto ClickVel:100
Record  Play  Stop  New  In/Out MAIN

```

3. Press the **MAIN** soft button to return to the MAIN page.
4. Set the recording track (RecTrk) to Track 2.

This track already has material recorded on it.

```

SongMode:MAIN  Events:186K  STOPPED
CurSong:200*NewSong  Tempo:100
RecTrk :2    Vol:127 Pan:64  Mode :Merge
Program:30 Warm Bass 1^2  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 **Warm Bass 1^2** to channel 2, it should appear in the Program parameter when you set Track 2 as the recording 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 is preceded by a minus sign (-). No material is recorded during the count off, though anything you play during the countoff gets quantized to the first Beat 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 lead (or whatever instruments you've chosen) into your song.

1. Set the recording track (RecTrk) on the 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: 22 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 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 play through only 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.

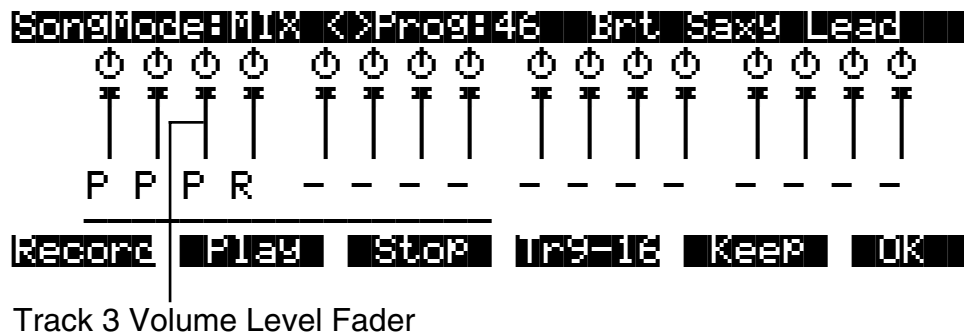
5. Continue recording instruments until you have played all the parts of your song.

### Mix Your Song

The 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 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. On keyboard models, the sliders control the volume of the current bank of channels (as indicated by the line near the bottom of the display). In the diagram below, the sliders would control the volume on Channels 1–8.



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

2. Use the **Right** or **Left** cursor button to position the cursor over Track 3's volume level fader on the MIX page.

Although the tracks aren't numbered on the MIX page, they're 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 save the change.

Now when you play back the song, Track 3's volume starts 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.

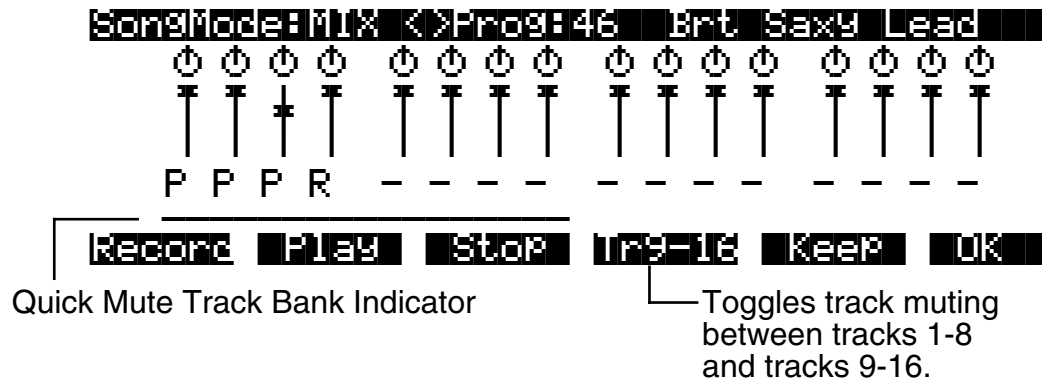
### Using the Mode Buttons to Mute a Track

Finally, we'll take a quick look at the track mute feature, which lets you use the K2661'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 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. The eight mode buttons



(Program, Setup, etc.) double as track-mute buttons while you're on the MIX page; press one or more of them and the corresponding tracks are muted. Press them again to bring the track back into the mix.



Use the **Tr 1-8 / Tr 9-16** soft button to toggle between banks of eight tracks, either 1–8 or 9–16. When you press this soft button, the horizontal bar repositions itself below the affected tracks. Table 12-1 shows the K2661's mode buttons, and which tracks they mute when you're on the MIX page:

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

**Table 12-1 Track Muting in Song Mode**

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

1. On the MIX page, press the **Play** soft button. The song, with all of its instruments, begins to play.
2. Press the **Program** mode button on the front panel. The Program-mode LED lights, and Track 1 (the drum track) is muted.
3. Press the **Program** mode button again to unmute the drum track.

Using your K2661's front panel buttons, you can mute one or more tracks at once, or even mute eight tracks at a time.

## Tutorial: Arrangements

The Arrangement Editor gives you a variety of ways to record and play songs. The following tutorial is designed to walk you through the steps of creating songs using the Arrangement Editor. Here are some typical tasks:

- Record two or more songs, then string them together in any order, as steps in an arrangement.
- Record additional tracks over the steps in an existing arrangement, saving the new tracks in the arrangement song.
- Use up to 32 tracks for recording and playback.
- Trigger songs or arrangements by striking keys (or triggering notes from any MIDI controller)—an excellent feature for live situations.

### Creating an Arrangement

This involves recording and saving each section of your final piece of music as a separate song, then using an arrangement to string together the sections in any order you choose. There are a couple reasons you might want to do this.

First of all, many pieces of music tend to be composed in sections. If your music tends to feature various sections repeated in varying order, using the Arrangement Editor can be easier than copying those sections and pasting them into a single song.

The second reason has to do with memory requirements. No objects (aside from samples) can be larger than 64k (10,000 to 16,000 notes, depending on the amount of controller information you record). You may need to use arrangements to accommodate the size of long musical pieces.

When you're recording songs that you intend to combine into arrangements, it can be helpful to name the songs Part 1, Part 2... or Chorus, Verse... That way, when you are putting together the arrangement, you can quickly identify each section. You might also want to organize the IDs of the songs. For example, you could start numbering the songs at the second ID in a memory bank (301, 302...), then save the arrangement song at the first ID in the bank (300, in this example). That way, you can create a file containing the arrangement song and its constituent songs, and when you load it into the K2661, the arrangement (the one you're likely to want to play) will be the first song in the bank.

Once you have your songs, you're ready to create another song to use as the arrangement.

1. In Song mode, select **1 NewSong** as the value for the CurSong parameter. This is important, because the arrangement song should not contain any note or controller information.
2. Press **Edit**, and set the Tempo parameter to match the tempos of the songs you'll be adding to the arrangement. (This assumes they are all the same tempo. We'll cover how to deal with different tempos later on.)
3. Press either **more** soft button, then press **Save**. The save dialog gives you the opportunity to rename the song and give it the ID of your choice. Press **Save** again when you've made the changes you want.
4. Press the **ARRANG** button (you're still in the Song Editor). The top line of the display tells you that you're looking at Step 1 of a song that contains one step. Cursor down to the Song parameter and select the song you want to use for the first step in the arrangement.

5. Press **Add** to add another step to the arrangement. Select the song you want for the second step. Continue adding steps and selecting songs as needed. If you need to get rid of a step, select the step and press the **Delete** soft button.

If you press the **Play** soft button while on this page, you'll hear the currently selected step. (If the step's Mode parameter is set to Next, the K2661 will play the next step when the current step finishes.) Once you have more than one step in an arrangement, you can use the **Chan/Bank** buttons to scroll through the steps.

6. Save, then press **Exit** to return to the MAIN page in Song mode. Notice that there are now three dashes below the Program parameter. Press **Play**, and the dashes change to indicate the current song, and the current step. The Locate parameter shows the bar number of the arrangement, not the bar number of the current step. In other words, if the first step in an arrangement has 16 bars, then when Bar 1 of Step 2 is playing, Locate will show **17:1**, not **1:1**.

## Arrangement Parameters: the ARRANGE page

### **Mutes**

The Mutes parameter has 16 values, in groups of 4. By default, tracks are unmuted (active), as represented by the dashes. To mute a track, move the cursor to highlight the dash corresponding to the track you want to mute, and press either **Plus/Minus** button to change the dash to **M**. This mutes the track.

Muting tracks is a good way to make several different-sounding steps out of the same song. If you create a multi-step arrangement in which each step uses the same song—with different tracks muted in each step—you can bring different parts in and out in each step.

### **Xpose**

You can transpose any step up or down. This allows you to transpose the song without having to edit the actual song data. Every track in the step gets transposed.

This is great for many sounds, but not so useful for steps containing programs that have different timbres assigned to different keys—like drum programs. Fortunately, you can designate any track as a drum track, which prevents it from getting transposed when you transpose the step. You must designate drum tracks in the individual songs that are used as steps in the arrangement, not in the arrangement song.

To designate a drum track, go to the COMMON page for the song containing the track in question. Find the DrumTrack parameter, and move the cursor to highlight the dash corresponding to the track in question. Press either **Plus/Minus** button to change the dash to a **D**. Don't forget to save.

### **Times**

You can set any step to play from 1 to 120 times before stopping or continuing to the next step in the arrangement.

### **Mode**

This is normally set to **Next**. In this case, the arrangement will play the next-highest-numbered step once the current step is finished. If the last step is set to **Next**, the arrangement will repeat Step 1. To make the song stop after the last step, set the last step's Mode parameter to **Stop**.

## Arrangement Parameters: the COMMON page

### **Start Step**

This determines the step number where the song will start. Normally this is 1, but it can be any step in the arrangement.

### **Tempo Control**

You can choose whether the tempo is controlled by the arrangement song itself or by each song in the arrangement. When set to **Song** it will use the tempo and time signature set in the arrangement song itself. When set to **Arrange**, it will use the various tempos and time signatures used in the arrangement's constituent songs.

### **Timing Issues**

You might encounter timing problems when using the Arrangement Editor; notes can be delayed when the arrangement switches from step to step. Here's how to avoid the problem.

Each time you record a track for the first time, the K2661 places four events right at the beginning of the track: Bank Change, Program Change, Volume, and Pan. If you go into the Event Editor, you'll see these four messages appearing at 1:1:000. Normally they'll be the first four events you will see.

By the time you have recorded several tracks, these events start to become a large number of events all occurring at the exact same point in time. The K2661 processes these events sequentially, and if enough events happen at the same time, some of them get delayed. For example, if you have 10 tracks, then you will have 40 of those initial events, in addition to any note and controller info that also exist at 1:1:000 in a track.

To prevent the delays caused by too many events, you can delete unneeded events. Typically, you are probably not changing the Program, Volume, and Pan settings for each track when you switch from step to step. For example, quite often you might have the same program on a MIDI channel for all the steps. In this case, all of the program and bank change messages after the first step are not needed, and can be deleted. You can edit each step in the arrangement this way (don't remove these events from the first step, however).

There are two different methods you can use to get rid of these messages in a track. The simplest way is to go into the Event Editor. Use the **Chan/Bank** buttons to select the track whose events you want to edit—you can select each track individually, or select All to view the events from every track in the step. To remove an event, highlight it and press **Cut**.

The second way is to use the Erase function in the Track Editor. Again, choose the track with the **Chan/Bank** buttons. Set the From parameter to **1:1** and the To parameter also to **1:1**. Of course, you won't want to have Events set to All, or you'll erase any note events that occur at 1:1, as well as the unwanted events. If you set Events to **Program Change** and press **Go**, you'll erase the Program and Bank change events. You can then set it to **Controllers**. At this point, if you leave Ctl set to **All**, you can delete both the Pan and Volume events with one operation. But if you have other controller info that occurs at 1:1 (such as Mod Wheel or sustain pedal) then you would also be erasing those events. So you can use the Ctl parameter to select just Volume and just Pan, pressing **Go** after each selection.

In general, the Event-editor method is quicker if you are working on one track at a time. But if you have many tracks and know that you want to erase these events from all of them, using the Track Editor with all tracks selected is faster.

Of course, if you are changing the program changes, pan, or volume in a track when the song changes from step to step, you need to leave those events in, but typically you might be doing that in only one track, while six or seven other tracks stay the same.

Timing problems can also occur between steps due to improperly-located endpoints. If you have a timing problem, you should check in the Event Editor to make sure the end point of each step falls on the first beat of the bar *after* the last bar of the step. For example, if your step is 8 bars long, the end point should be at 9:1:000.

### Removing Initial Events from Step One

Since you almost always want to have initial Program, Bank, Volume, and Pan events in an arrangement to make sure it plays properly, it makes sense to have those events in each track of the song used for the first step on the arrangement. But what if that same song is used in a later step, or Step 1 plays a number of times? In this case, as soon the step restarts, you have unneeded events that could contribute to delays.

In this case, the solution is to delete those initial events from the Step 1 song, and record them into the arrangement song. In the arrangement song, select a recording track, press **Record**, then highlight the Program parameter and select the program you want for the track. You can also select the Pan and Volume parameters if you want to set them to a specific volume. Once you have these parameters set, press **Stop**. You need to do this for each track that you are using in the song.

## Recording Additional Tracks

So far, all of the recording we've described has been done in the individual songs used as steps. The arrangement song we created has no data in it.

But you can also record tracks in the arrangement song. For example, you might want to record a series of rhythm section grooves: just bass, drums, and maybe some comping parts. Now you can use those grooves as step in an arrangement, then record lead lines through the entire arrangement.

1. Follow Steps 1 through 6 of *Creating an Arrangement* on page 12-12 to create an arrangement song, using some different grooves you have created.
2. Start recording new tracks in the arrangement song. Remember that each MIDI channel can have only one program assigned to it. Therefore you may want to select the tracks you record in the arrangement song so that their MIDI channel assignments are different from those of the tracks in the step songs.

## Maximizing Track Use

If you extend the previous example, you'll realize that you can actually make use of 32 MIDI channels—by creating an arrangement containing steps that use all 16 channels, then recording 16 channels of music in the arrangement song itself. Both the step songs and the arrangement song can play back through the K2661, through another instrument connected to the K2661's MIDI Out port, or through both.

It is important to remember that there are still only 16 MIDI channels, and any one MIDI channel can play only one program. Therefore there is no way to have more than 16 different programs playing at the same time on the K2661. But there are two reasons why you would want to use more than 16 tracks.

First, you can have two or more tracks assigned to the same MIDI channel. For instance, if you were recording drums, you might want to put different drums from the same program on different tracks, to make recording and editing easier. On the bottom line of the display on the MAIN page in Song mode, there are 16 channel parameters, one for each track. The numbers don't represent *tracks*; they represent the track's MIDI channel assignment. (The dashes—or

other characters—above the numbers represent the tracks. The dashes and characters are the values for the Track parameter.) You can assign any track to any MIDI channel.

The second way you can use more than 16 tracks is if you have an external sound module in addition to the K2661. Each track can be assigned to play only the K2661's internal sounds (local), or to be sent only to the MIDI Out, to play the external instrument. To assign the track for local or MIDI playback, go into the Song Editor and on the COMMON page you'll see 16 Track Destination parameters. If the value is a dash, the track is going to both the K2661 and its MIDI Out port. **L** means local, and plays the K2661 only. **M** means that the track goes only to the K2661's MIDI Out port. A value of **x** means that the track is muted.

In the following example, all the tracks in the step song play the K2661, while all the tracks in the arrangement song go to the MIDI Out port. You can also have the step songs and arrangement song set to the same track destinations, as long as it's OK for them to play the same sounds.

1. Create a song with multiple tracks and save it.
2. On the COMMON page in the Song Editor, set the TrackDest value for each track of this song to **L**. Save, then press **Exit** to return to the MAIN page in Song mode.
3. Using the Cursong parameter, call up **1 NewSong**, and press **Edit**, then **ARRANG**.
4. Using the Song parameter, call up the song you just recorded. This makes your song a step in the arrangement song, which you're currently editing. Press **Done** to return to the COMMON page.
5. Set the TrackDest parameter for each track to **M**, so that the arrangement song won't also play the K2661.
6. Save, then press **Exit** to return to the MAIN page in Song mode.
7. Record additional tracks as part of the arrangement song. You might have to record programs changes in each track of the arrangement song to set up the external instrument properly.

## Triggering Arrangement Steps From the Keyboard

Each step in an arrangement can be triggered by playing a key (or triggering a note from any MIDI controller). This can be great for live performance, because you can repeat each step as many times as you like.

1. Go to the TRANSMIT page in MIDI mode, and assign a control setup that has a value of **On** for the Sync parameter on the COMMON page of the Setup Editor. Create one if you need to, and name it **SongSetup**. We'll explain why shortly why you need to use this control setup.
2. Go to Song mode, and create an arrangement, following Steps 1 through 6 of *Creating an Arrangement* on page 12-12.
3. Go to the COMMON page in the Song Editor. There are two parameters on this page for use with key triggering of steps:

**TriggerChan:** Notes on this MIDI channel can trigger the current step. Notes on any other channels will not trigger the step. Set this to match the MIDI channel of the K2661 or whatever controller you're using to trigger the steps.

**TriggerCtl:** This determine if the keys will trigger the steps. Set this parameter to **ON**, so when you hit the appropriate note on the trigger channel, the step will start playing. If you set it to a MIDI Controller number that has a physical controller assigned to it, then the keys will trigger the step only when the controller is on (for switch controllers) or above its halfway point (for continuous controllers).

4. Press **ARRANG**, then press the **Chan/Bank Down** button until you see Step 1 in the top line of the display. Set the Mode parameter to a value of **Stop**.
5. Note the values for the LoKey and HiKey parameters, then set them to **E 1** and **E 2** by doing the following:
  - Press the **SetRng** soft button
  - Strike E 1 on the K2661 or your MIDI controller
  - Strike E 2

Note the new values for LoKey and HiKey. Now the current step starts playing when you strike E 1. Strike another note (say E 2) while the step is playing, and on the first beat of the next bar, the step repeats, and all the tracks that aren't drum tracks get transposed up a corresponding number of semitones (in this case, an octave). Generally, when you're triggering steps using keystrokes, you'll hit the triggering key somewhere in the last bar of the current step. This causes the new step to start right after the end of the current step.

You can also cause steps to start as soon as you hit the triggering note. To do this, go to the **TRANSMIT** page in MIDI mode, and assign a control setup that has a value of **Off** for the Sync parameter on the **COMMON** page of the Setup Editor.

6. Set the Latch parameter to a value of **On**. Now the steps will continue playing after you've released the triggering note.
7. Set the VelTrk parameter to a value of **Off** if you want the step to play back at the level at which you recorded it. Set it to **On** to vary the playback level according to the velocity of the keystrokes that trigger the step.

## RAM Tracks

If you have the K2661 sampling option, you can create RAM Tracks, which combines Song mode with the sampler. The RAM Tracks feature enables you to create a sample during song playback, then have the K2661 do the work of building a program out of the sample. The K2661 also inserts the sample into the song so that it plays back in sync with the song.

Possibly the best feature of RAM Tracks is how it affects polyphony. By sampling the K2661's audio output, you can condense an entire song into a single track that uses only two voices of polyphony.

## Two Important Concepts

### RAM Tracks and Song Playback

You don't have to be *recording* a song to create a RAM track. The idea behind RAM tracks is that you can make a quick sample during playback of a song, then integrate that sample into the song. Of course, you *can* create a RAM track while recording a song, but it's often best to take one step at a time.

#### A Matter of Timing

The song must *already be playing* when you start sampling. There are several ways you can do this:

- Start the song, then at the appropriate location in the song, press **Record** on the SampleMode page, and start the sample input.
- Set the Thresh parameter on the SampleMode page to a dB value—one you know you'll exceed with your sample input signal. Start the song, then start your sample input at the appropriate location in the song. This method makes it easy to sync your sample with the song.
- Set Thresh to **Key**. Start the song, then at the appropriate location, then trigger the sampler by striking a key either on the K2661 keyboard (or on a MIDI source that's controlling the K2661), and start the sample input. You must send the trigger on the K2661's current MIDI channel.

### Creating RAM Tracks

1. Start by configuring the sampler: go to Program mode, and press the **Sample** soft button to go to the SampleMode page. Set the Input parameter to **Analog**, and set the Source parameter: **Ext** for an external source, **Int** if you want to sample the K2661's output. Set the Time parameter to give you enough time to record the sample you want. Set the Mon parameter to **On** if you want the K2661 to play the sample input through its audio output. Set the Mode parameter to Mono (L), Mono (R), or Stereo, depending on your input signal. Adjust the Gain parameter to bring the signal level as close to 0 dB as possible.
2. Go to Song mode, and call up a song that has at least one empty track. If you plan to start sampling right at the top of the song, you might want to set the Click parameter (on the MISC page) to a value of **Cnt**, which gives you a countoff before the song starts playing (the value of the CountOff parameter—also on the MISC page—sets the number of bars of countoff).
3. Set the RecTrk parameter to an empty track.
4. Start the playback of the song. On keyboard models, press the **Play/Pause** button. On rack models, press the **Left** and **Right** cursor buttons at the same time.
5. Press **Record** to start sampling. When you have the sample, press **Stop** (if you run out of time or sample memory before you press **Stop**, the K2661 stops sampling automatically). Either way, the song stops its playback as well.
6. The K2661 prompts you to strike a key to set the sample root. Strike any key (or trigger any note) that you want, or press **Default** to set the root at C 4. The K2661 shows you the maximum signal level, or if the sample clipped, it shows you the number of clips. It also prompts you to save the sample.
7. Press **Yes** if you like the sample, and the save dialog appears. We recommend naming the sample now, to make it easy to keep track of it. The name you give the sample will also be the default name for the song, when the time comes to save the song.
8. Once you've saved the sample, the K2661 asks you if you want to place the sample into the current song. When you press **Yes**, the K2661 asks you to pick the track you want to use for the sample's playback. Pick any empty track: the track must be empty because the



K2661 will eventually create a new program for the sample, and that program must be on its own MIDI channel. Press **OK** when you've selected a track.

9. The K2661 creates a program, and shows you the new program's ID. It then prompts you to strike a key to specify a note for triggering the sample during the playback of the song. It can be any key; the K2661 automatically handles the transposition required to ensure that the sample plays back at the right pitch.
10. Go to the MAIN page in Song mode, and on the recording track you set in Step 3, you'll see the new program. The program assigned to this track has the same name you gave the sample when you saved it. If you look at the events on this track (on the EVENT page in the Song Editor), you'll see a PCHG event that selects your new program, and a note event corresponding to the key you struck to set the sample trigger.
11. Repeat Steps 5 through 9 to create additional samples and insert them into the song. You can use the same track, or a different, empty track. For this example, we'll use the same track. In this case, when you press **OK**, the K2661 asks you if you want to add the sample to the program that's already being used for the track (the program you created when you pressed **OK** in Step 8. Press **Yes**. The K2661 prompts you to strike a key, as in Step 9. If you strike a key that's already being used by one of the samples in the program, the K2661 alerts you, and lets you to assign a different key (press **No**) or the same key (press **Yes**—although if you do this, then the song will trigger all samples that share the same trigger key, every time the trigger key gets played).
12. If you use the same track, the K2661 creates another layer in the program on that track, and assigns the sample to it. You can record up to 32 samples on this track, since a VAST program can contain up to 32 layers. If you use a different track, the K2661 creates another new program when you place the sample into the song.

When you've recorded all the tracks you want, you can edit the programs assigned to those tracks, using all the features available for VAST synthesis.



***Note:** There's no way to synchronize sample playback exactly to MIDI. While the K2661 handles the synchronization extremely accurately, it's possible for the sample to drift away from the song playback, at a rate of up to .5 milliseconds per minute of playback. That's a minuscule amount, but it might matter to you. If you need to be more precise, we recommend that you keep your sample time short, and record a relatively large number of short samples, as opposed to a small number of long samples. This will virtually eliminate the risk of your samples drifting out of sync with the song playback.*

## Using Song Mode

### 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 K2661 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

K2661 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 TRANSMIT page is turned on.

If you want to use a different program for playback than the one originally recorded, you have two alternatives. First, you can edit the song, changing each individual Program Change event (PCHG) to reflect the desired program IDs. This is done on the EVENT page in the Song Editor. Or you can set the RecTrk parameter to the track on which you want the program change, highlight the program parameter, press **Record**, select the desired, program, and press **Stop**.

Another alternative is to defeat the song's recorded program changes, and manually set each MIDI channel used by the song to play the desired program. Start by setting the ProgLock parameter to a value of **On**. The ProgLock parameter is found on the MIDI-mode CHANNELS 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 K2661's sound engine will ignore all program changes it receives on that channel, whether it's via MIDI or from the K2661's front panel, or from within the song.

When you've set the ProgLock for each of the channels used in the song, 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. When you set ProgLock back to Off, the originally-recorded program changes take effect again.

## Effect Selection During Recording and Playback

When you're recording or playing back a song, the setting for the FX Chan parameter (on the Effects-mode page) determines which studio (plus FXMods) gets applied to the song. Only one studio 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 Chan 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 Chan parameter to a value of **Current**—which is the most generally useful combination of settings for these parameters. In this case, while you're in Song mode—even if you move to another mode during playback or MIDI recording—the value of the FX Chan parameter automatically changes to match the song's effect channel (the value of the EffectChan parameter on the COMMON page in the Song Editor). Therefore the effects applied to the song are determined by the program assigned to the song's effect channel, 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 K2661 has an internal MIDI clock, which is always running at a speed set by the Tempo parameter (on the MAIN page). When you're in Song mode and the Clock parameter (on the MISC page) is set to a value of Internal (**Int**)—and the Sync parameter (also on the MISC page) is set to **Xmit** or **Both**—songs will sync to the K2661's internal clock. At this setting, the clock signal is sent to the K2661's MIDI Out port. This is standard MIDI Sync, and any device that accepts MIDI Sync will play in sync with the K2661.

If you set the Clock parameter to external (**Ext**), the K2661 expects to receive MIDI clock at its MIDI In port. Make sure that the Sync parameter on the MISC page in Song mode is set to **Recv** or **Both**, to enable the K2661 to receive sync messages as well as MIDI clock.

To play back a song, press **Play**, and the K2661 starts as soon as it receives both MIDI clock and a Song Start message. Or if you want to record, press **Record**, and the K2661 starts recording as soon as it receives both MIDI clock and a Song Start message.

If Sync has a value of **Off** or **Xmit**, the K2661 still uses the external clock, but you can't trigger recording or playback remotely; you have to use the K2661's soft buttons.

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 **Ext**, the K2661 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 **Int**, 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 **Ext**, tempo changes will not carry over to the K2661, and will not be heard when you play back the song.

Finally, keep in mind that when the Clock parameter is set to **Ext**, 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 clock signal is received, the Clock control sources are disabled.

### Songs and Effects: A Brief Tutorial

On page 12-20 we discussed how to keep the effects constant during song playback and MIDI recording. For more complete control over the effects used in a song—including real-time control—we recommend dedicating one track of the song to effects control. Here's how to do it.

1. Go to Effects mode, and make sure that the value of FX Mode is **Auto**, and the value of FX Channel is **Current**.
2. Go to the MAIN page in Song mode, and call up a song.
3. Press **Edit**, which takes you to the COMMON page in the Song Editor.
4. Set the EffectChan parameter to the channel you want to use for effects control. Choose a channel that isn't being used for any of the existing tracks in the song. Channel 16 is the default click-track channel, so you probably don't want to use Channel 16. For this tutorial, we'll use Channel 15.
5. Save the song and exit the Song Editor. You're back on the MAIN page in Song mode.
6. Set the recording track (the RecTrk parameter) to match the effects channel that you set in Step 4—in this case, Channel 15. Note the **R** in the Track status indicator line.
7. Move the cursor to highlight the current program. This is where you select the program to be used for the current recording track. Since you aren't going to record any notes on this track, the program assignment doesn't matter. On the other hand, if you already have a program that uses a studio and FXMods that you like, use it here.
8. Press **Edit**, and since the cursor was highlighting an editable object (the program), you'll enter the Program Editor.
9. Go to the KDFX page, and choose a studio. Assign any FXMods that you want to use.

10. Save the program (we recommend renaming it as well), then press **Exit** to return to the MAIN page in Song mode.

So far, so good. Your song uses Channel 15 for effects control, and the effects for the song are determined by the studio used in the program on Channel 15. That is, every program in the song directs its output to this studio, according to the value of the Pair parameter on the OUTPUT page in the Program Editor.

You'll recall that a studio has four inputs, each of which can be responsible for a different effect (or no effect at all). You may want to edit each program in the song, to send its output to the desired studio input. There's a quicker way, however, that doesn't involve editing programs: you can determine program output (and consequently studio input) based on MIDI channels. For example, you can send Channel 1 to KDFX-A, in which case any program on Channel 1 will send its output to KDFX-A, regardless of the program's output settings.

1. Press the **MIDI** mode button to enter MIDI mode, and press the **CHANLS** soft button. This takes you to the CHANNELS page.
2. Press the **Chan/Bank Up** or **Down** button to display the settings for one of the channels you've used in your song. (the top line of the display shows you the current channel).
3. Change the value of the OutPair parameter. Any value other than **Prog** means that the output settings are determined by the *channel*, not by the program assigned to that channel.
4. Change the value of the OutPair for the other channels that you've used in the song.

Now you have all the outputs directed to the right studio inputs. Keep in mind, though, that the output settings are customized for the current song. You'll need to repeat this process for each song—unless you set OutPair back to **Prog** for every MIDI channel.

Next you'll need to make sure that the song always calls up the right studio.

1. Go to the MAIN page in Song mode, and make sure that the recording track is still the one that uses Channel 15. Also make sure that the program on that track is the one containing the studio you want to use for the song.
2. Record a couple of bars. This automatically puts a Program Change command at the beginning of the track. Each time you play the song, the K2661 selects that program—and consequently the studio and FXMods associated with that program.

That's it. If you want the song's effects to change during playback, use this track to record movements of the controllers that are designated as FXMods.

With one small difference, this approach to effects control also works if you're using an external sequencer. When you're recording in Song mode, the EffectChan parameter (on the COMMON page in the Song Editor) determines the channel used for effects control. When you're using an external sequencer, it works a bit differently.

1. Press the **Effects** mode button to enter Effects mode.
2. Set the value of the FX Mode parameter to **Program**.
3. Set the value of FX Channel to whatever channel you want to use for effects control.
4. Create a program that uses the studio and FXMods you want, and use that program on the channel you chose as the FX Channel.

## 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" field 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 K2661 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 K2661 always reserves a minimum of 4K of RAM.

If you want to delete a song and the K2661 won't let you enter the Song Editor, select the default song (**1 NewSong**). Since it's smaller than 4K (as long as you haven't saved any changes to it), 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. (You could also delete the song in Master mode—by pressing Object, then Delete, to get to the file deletion dialog.)

## Loading MIDI Files From Disk

If you have a Type 0 or Type 1 MIDI sequence file stored on a SmartMedia card or a SCSI device, you can load it into one of the RAM banks, and the K2661 will be able to play it from Song mode. If the MIDI file has the GM On Sysex message in it, the imported song will have the GM parameter set to On automatically.

## 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 K2661's MIDI In port. Select Song mode, and set the Clock parameter to External. This will sync the K2661 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 K2661 will record only information coming in on the channel that the RecTrk parameter is set to.

Press the **Record** button, and the K2661 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 K2661 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 K2661 will stop recording and ask if you want to save the song.

# Song Mode: The 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:375K  STOPPED
CurSong:1  NewSong  Tempo:120
RecTrk :1  Vol:127  Pan:64  Mode :Erase
Program:1  Concert Piano 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
```

Parameter	Range of Values	Default
Current Song (CurSong)	Song ID & Name	1 NewSong
Recording Track (RecTrk)	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 (Locat)	1:1 – 9999:9	1:1

The Events field on the top line displays the number of events that you can store in RAM. 375 K in the above example represents maximum available memory. Note that this figure shows the number of free *events*, each of which takes up about four bytes of RAM. That’s why the number you see here is typically about 25% of the free RAM (in *kilobytes*) you see in the Samples field in the top line of the Disk mode and Master mode pages.

When the Song Status is REC READY or RECORDING, the Events field changes to Used, and indicates the percentage of the recording buffer that you have filled, instead of the free event space.

```
SongMode:MAIN  Used: 0%  REC READY
```

**Song Status**, also on the top line of the display, is always one of the following:

<b>STOPPED</b>	The default sequencer status; also appears when you press the <b>Stop</b> or <b>Pause</b> button.
<b>PLAYING</b>	Appears when the <b>Play</b> button is pressed, but only if the following conditions are true: the <b>Record</b> was not pressed prior to pressing Play, the Key Wait parameter is set to <b>Off</b> , and the Clock parameter is set to <b>Int</b> (or Clock is <b>Ext</b> and MIDI clock is detected).
<b>REC READY</b>	Appears when the <b>Record</b> button is pressed while Song Status is STOPPED. REC READY flashes, indicating that the sequencer is waiting to start recording.
<b>RECORDING</b>	Appears when the <b>Play</b> button is pressed while REC READY is flashing (unless Key Wait is <b>On</b> or the Clock is set to <b>Ext</b> ). RECORDING also appears if the <b>Record</b> button is pressed while Song Status is PLAYING.
<b>KEY WAIT</b>	Appears when the <b>Play</b> button is pressed, while Song Status is STOPPED or REC READY, if the KeyWait parameter on the MISC page is set to <b>On</b> . KEY WAIT flashes, indicating that recording or playing will begin when you strike a key.
<b>EXT. CLOCK</b>	Appears when the <b>Play</b> button is pressed, while Song Status is STOPPED or REC READY, if the Clock parameter on the MISC page is set to <b>Ext</b> . EXT. CLOCK flashes to show that the K2661 is waiting for an external MIDI clock message to start recording or playing.

## Current Song (CurSong)

This shows the ID and 16-character name of the song currently 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 match the setting of the Tempo parameter.

## 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 on the COMMON page in the Song Editor. 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 12-45), 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.

## Recording 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 region (above the Track and Channels region). 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 change according to the value of RecTrk and in one case, the mode from which you enter Song mode. 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 **None**, the display changes to show the Channel parameter followed by the Program parameter (although the Program parameter's *name* doesn't appear, just its *value*). If you switch through the channels, the program also changes, showing the program currently assigned to that channel.

A setting of **Multi** makes the parameters below RecTrk dependent on the mode from which you entered Song mode. If you enter Song mode from Program mode, the Channel and Program parameters appear below RecTrk. If you enter Song mode from Setup mode, only one parameter, Setup, appears.

## Program

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

This parameter's name is not visible when RecTrk is set to **None** or **Multi** (to make room for the Chan parameter); just its value appears.

You'll see the Setup parameter instead of the Program parameter when you've entered Song mode from Setup mode. The Setup parameter functions similarly to Program.

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



To change a track's program quickly, press **Record**, select the program, then press **Stop**. Or you could press **MIXER** to go to the MIX page, change the program as desired, then press **Keep**. This preserves all changes you have made to any other tracks: volume, pan, tempo, etc.

```

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

Track   :R R R 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 available only when RecTrk is set to **None** or **Mult**. When RecTrk is Mult, this parameter appears only if you have entered Song mode from Program mode. In this case, the Channel parameter gets squeezed onto the same line as the Program parameter, which is why you don't see the Program parameter's name, just its value.

## Setup

Displays the ID and name of the setup to be recorded. This parameter is available when RecTrk is **Mult**, and you enter Song mode from Setup mode. The display diagram below shows an example of the Song-mode page with the Setup parameter replacing the Channel and Program parameters.

```

SongMode:MAIN  Events:188K  STOPPED
CurSong:1  NewSong          Tempo:120
RecTrk :Mult Vol:127 Pan:64  Mode :Merge
Setup   :B13 Jungle Jammer  Locat: 1:1

Track   :R R R 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

```

Using setups in Song mode takes a bit of extra planning. Since each zone in a setup uses a separate MIDI channel, you need to make sure that each setup zone has a corresponding track and channel allocation. It's important to know how many channels, and consequently how many tracks, are needed for recording a particular setup. Each setup can have up to eight 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.

To change a track's initial volume quickly, press **Record**, change the value of Vol, then press **Stop**.

## 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 initial value for the current track is modified in real time.

To change a track's initial pan position quickly, press **Record**, change the value of Pan, then press **Stop**.

## 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 appear only for tracks that already contain data.

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 the track contains data.

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 buttons to position the cursor onto a Track Status Indicator, you can toggle an empty track (-) into Record (R) with the Alpha Wheel or **Plus/Minus** 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 display an (R), designating it as the recording 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), 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 1–16 of a new song are assigned to Channels 1–16 respectively, although a track can play or record on any channel and the same channel can be used for more than one track. Keep in mind, however, 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'll play the same program—the one on the higher-numbered track, since that's the most recent Program Change command received on that channel.

## Soft Buttons on the MAIN Page

These buttons are similar to the transport controls on a tape deck. Some of those decks require you to press Play and Record simultaneously to begin recording. The K2661's transport buttons aren't like that, however. It's important that you press only one of these soft buttons at a time to insure proper recording 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 changes to this song?" dialog (described below), 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:

- There are 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 MIDI-mode 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 parameters) or from the record-enabled track if the cursor is positioned elsewhere. As described on page 12-32, 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—unless you've saved the earlier version to another ID. So before you start removing data that you may not want to lose permanently, you should also check out *Delete* on page 12-54, and *Song Editor: Track Functions* on page 12-52.

The Erase function doesn't show a dialog or erase any data if an empty track is selected.

**MISC** accesses more sequencer control parameters found on the Song-mode 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 12-32.

**MIXER** lets you view a graphic representation of knobs and level faders for each track on the MIX page. You can modify program changes, volume and pan position on this page. Record status can also be selected here. The section on the Mixer begins on page 12-39.

## Save this song? Dialog

The following dialog 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 changes to 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 (that is, 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 IDs available for song objects per bank in banks 100 - 900. In the zeros bank, there are 75, IDs 1-75. In the 200s bank there are IDs 200-219, in the 300s bank, IDs 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 11-15. 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 parameters, 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.

## Song Mode: The MISC Page

This page contains parameters that determine how the sequencer behaves during playback and recording, and when the K2661 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 :1/16   Click   :Rec
KeyWait  :Off    Swing:0%   ClickCh :16
Locate   : 1:1   Sync :Off   ClickPrg:198
AutoIn   : 1:1   Clock:Int  ClickKey:C 4
AutoOut  : 1:1   Tempo:Auto ClickVel:100
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	100

The top line displays the amount of free event 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 recording mode is set to.

When you record the first track, all recording 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 recording 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 on the EVENTS page.
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 Song-mode 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 RecMode, 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've 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 *Song Editor: The ARRANGE Page* on page 12-64.

**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, we recommend that you include a few measures of silence either at the end or the beginning of each song in the Chain.

When you want to string songs together more seamlessly, you can create an arrangement with the Arrangement Editor. See page 12-64 and page 12-12.

## 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**. When this parameter's value is highlighted, pressing the **In/Out** soft button updates only the AutoIn parameter; it doesn't update the AutoOut parameter.

## 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 12-55.

## 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, for example,  $1/12 = 1/8\text{tr}$ ,  $1/24 = 1/16\text{tr}$ , and so on) are available as the size of the Input Quantize grid. You can select commonly used Grid values by double-pressing the **Plus/Minus** 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.

### Song Select

Song Select is a command similar to a Program Change command; it enables you to select songs via MIDI. Like other MIDI messages, Song Select has 128 values. That's about half the number of songs you can store in the K2661, so if you have a lot of songs, you can't use Song Select to select all of them.

The K2661 maximizes the number of songs you *can* select by remapping Song Select values of 0–127 so they match the way songs are stored in the K2661's RAM banks (75 in the Zeros bank and 20 in each of the other nine banks). If you want to use Song Select extensively, you should number your songs according to the song IDs listed in the following table. Songs in the Zeros, 100s, and 900s banks are not available for selection with Song Select.

Song Select Value	Song ID
0–19	200–219
20–39	300–319
40–59	400–419
60–79	500–519
80–99	600–619
100–119	700–719
120–127	800–807

### Song Position Pointer

The K2661 sends a Song Position Pointer message via MIDI when you start a song. The message indicates the location (Bar and Beat) at which the song starts playing—often that's the start of the song (1:1), but you can set the Locate parameter anywhere you want, and start the song from

there when you press **Play**. The K2661 also responds to Song Position Pointer messages received from an external source.

When you have the K2661 synced to another sequencer, the Song Position Pointer message shifts the auxiliary sequencer's start point correspondingly. This is normally an extremely convenient feature, but there's one thing to avoid.

The K2661 lets you set a negative value for Locate, which gives you a countoff before the song starts (if you do this, set the Countoff parameter on the MISC page to **Off**, so you don't repeat the countoff). Keep in mind that the Song Position Pointer message doesn't support negative values, so your auxiliary sequencer might lose sync if you use a countoff. The safest approach is never to start a song with a Locate value less than **1:1**.

## Clock

Specifies the source clock as being internal or external. When the Clock is set to external (**Ext**), the K2661 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 parameter is set to **Off**, the CountOff setting will have no effect. If the click is set to record only, then the CountOff will happen only 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**. A value of **Cnt** generates a click only 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 gets played at exactly this velocity level while the other beats scaled to about 90% of this value as a way to provide an accent.

If there's not enough of a distinction between Beat 1 and the other beats, you can do the following to extend the dynamic range of the click program:

1. Go to Program mode, and select Program **198 Click**, then press **Edit**.
2. Press the **more>** soft button, then the **F4AMP** soft button.
3. Select the Velocity Tracking (VelTrk) parameter, and increase its value. The more you increase it, the more difference you hear between Beat 1 and the other beats. Don't go too high, though. This may cause Beats 2, 3, and 4 to drop too low. When you save, we recommend using a different program name and ID, or at least a different ID.
4. Go back to Song mode, select the new program's ID as the value for the ClickPrg parameter, and start recording.

## Soft Buttons on the MISC Page

**Record**, **Play(Pause)**, and **Stop** work the same as they do on the MAIN page, described on page 12-24.

**New** selects **1 NewSong** as the current song and jumps back to the 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, based on your current location in the song. When you press **In/Out**, the K2661 updates either the AutoIn or AutoOut parameter (or both), depending on their current values and the value of the Locate parameter.

Typically, you'll start song playback, press **In/Out** when you reach the desired punch-in point, then press it again when the song reaches the desired punch-out point.

More specifically, the first time you press **In/Out**, the K2661 sets AutoOut to match the current value of Locate. Press **In/Out** again, and the K2661 shifts the current value of Auto Out to the value of Auto In, and updates AutoOut to match the current value of Locate.

If the value of AutoIn or AutoOut is highlighted when you press **In/Out**, the K2661 updates only the selected value.

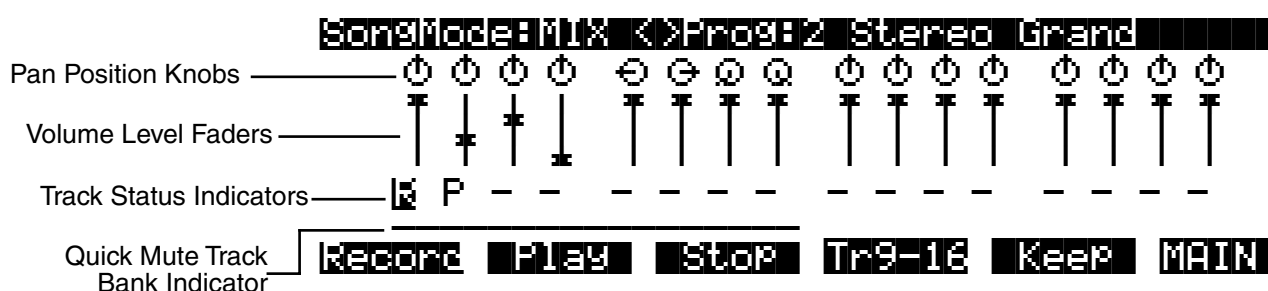
The AutoIn and AutoOut parameters display only Bars and Beats. However, the actual In and Out points will be precise to the Tick occurring at the time the **In/Out** button was pressed.

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

## Song Mode: The 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, 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 RecMode is set to **Merge**, or else the data previously recorded on the track will be erased.



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** cursor buttons. 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 alphanumeric buttonpad 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 alphanumeric buttonpad 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).

You can also use the sliders to adjust the volume of the currently selected bank of eight tracks.

## Track Status Indicators

Using the **Up**, **Down**, **Left**, and **Right** cursor buttons to position the cursor onto a Track Status Indicator, you can toggle an empty track (-) into Record (R) with the Alpha Wheel or **Plus** or **Minus** 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 display an (R), 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), 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), the RecTrk remains 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**/**Tr 9-16** soft buttons, below. It also indicates which tracks respond to the keyboard model's sliders for adjusting track volume.

## 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 display 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 an 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 Song Editor

In general, you'll get to the Song-editor pages by pressing the **Edit** button any time you're in Song mode. There's one exception: if the Program parameter is currently highlighted on the display, you'll enter the Program Editor when you press **Edit**.

There are a few conventions shared by all of the Song-editor pages (except the ARRANGE page). Displayed at the top of each Song-editor page is the name of the page and the currently selected track (1–16, or all tracks). All of the values for the parameters found in any of the Song-editor pages are saved in the song object.

### Soloing the Current Track

On all of the Song-editor pages, you can solo the current track by pressing either the **Setup** or **Quick Access** mode buttons (you can press the **Program** mode button to mute the current track). 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 solo mode.

## Song Editor: The COMMON Page

Press the **Edit** button on the front panel of the K2661 to display the 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-editor pages.

```

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

The top line of this page displays the selected track or tracks. Select the available current track 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**, this parameter specifies which channel will control effects (studio and FXMods) while in Song mode and during playback of a song. In this situation, you'll hear the studio assigned to the program that's on the channel designated as the EffectChan will be heard.

Every song can have its own setting for EffectChan. This makes it convenient to change studios every time you change a song, but you'll need to pay more attention to which programs are controlling the studios—especially if you're creating arrangements with lots of steps. If you want to limit changes in studios as you change songs, you can do it easily. See *Effect Selection During Recording and Playback* on page 12-20 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.



## GM

When you set GM On (if the K2661 is not already in GM Mode), the unit will put itself into GM Mode when the song starts, and get out of GM when the song stops. The GM Mode setting will be saved with the song file.

GM Mode can also be turned on from an external MIDI device that sends the “General MIDI On” message as defined in the MIDI Specification. When you insert a General MIDI On message at the beginning of a sequence you are telling the playback device (e.g., K2661) to immediately start GM when you play the song. On pause, stop, or end of song, GM Mode is turned off automatically.

In addition, when you import Standard MIDI Files (SMF) with the GM sysex embedded in them, the K2661 sets the GM parameter in the song to ON. And when the K2661 exports a song to SMF with the GM parameter set, it embeds the GM On sysex message.

Note that the K2661 will only respond to sysex broadcast messages when the K2661’s sysex id is set to 127. (Sysex broadcast messages have <device ID> set to 7F, or 127 decimal.)

The Sysex messages are:

### GM ON

F0 7E	UNIVERSAL NON-REAL TIME SYSEX HEADER
<device ID>	ID of target device
09	GM Message
01	GM ON
F7	EOX

### GM OFF

F0 7E <device ID> 09 02 F7	
02	GM OFF

## Parameters Used with the Arrangement Feature

The remaining parameters, as well as the Drum Track parameter, are used in conjunction with the Arrangement feature (via 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 12-64 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. For example, suppose you set this parameter to a value of **MIDI 29**, which by default is the destination of Panel Switch 2 (Button 10). When Panel Switch 2 is on, striking any key in the trigger range (specified by the LoKey and HiKey parameters on the ARRANGE page). If you set this parameter to **ON**, striking a key in the trigger range always triggers a step.

**TrackDest**

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

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

–: This is the default setting for all tracks in a new song. It means that MIDI data on the track are transmitted both locally to the K2661, and out the MIDI Out Port on the track's channel to any external devices.

**L**: The track's MIDI data will be transmitted locally only, to the K2661's internal sound generator. None of the track's MIDI data will be sent to the MIDI Out Port.

**M**: The track's MIDI data will be transmitted only to MIDI Out.

**x**: Disable the transmission of the track's MIDI data either locally or to MIDI. This is a quick way to mute a track more permanently without erasing it.

**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 nonpitched 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) designating 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 on the TRACK page in the Song Editor. Any tracks defined as Drum Tracks are transposed when a transposition is applied to these tracks from the TRACK page.

## Soft Buttons on the COMMON Page

**EVENT:** Accesses an Event-list style editor. On the EVENT page in the Song Editor, you can scroll through, modify, add, or delete any or all of the tracks' MIDI events. The EVENT page is described on page 12-45.

**TRACK:** Brings you to the 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 TRACK page is described on page 12-48.

**STEP:** Brings up the STEP page, from which nonreal-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 ARRANGE page. The STEP page is described on page 12-61.

**ARRANG:** Enters the ARRANGE page, gaining access to a group of arrangement specific parameters. The ARRANGE page is described on page 12-64.

## Song Editor: The 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 location (Bar and Beat, in the first column) is highlighted. Use the Alpha Wheel, the **Up** and **Down** cursor buttons, or the **Plus/Minus** 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 (MIDI 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 example, 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 (or press **Chan/Bank Up** or **Down** until you scroll past the highest- or lowest-numbered track in the song). As you scroll through each event, the track and channel for the selected event appear in the top line of the display.

## 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 alphanumeric buttonpad 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 alphanumeric buttonpad.

## 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 alphanumeric buttonpad, 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 and Value region displays the MIDI event type (and related information) at each Event-list location in the song. Different event types display different kinds of information, and have different editable values. You can't change the type of an event, but you can change the values associated with that event type.

For example, you can't change a CTRL message to a PCHG message, but for any control message, you can change both its destination and value. A Note On is always a Note On—but you can change the values for its note number, attack velocity, release velocity, and duration. Incidentally, you won't see the event type for Note Ons, just the editable values.

Table 12-2 lists the ranges of the editable event values.

MIDI Events	The PCHG, BEND, MPRS, SYSEX, TMPO, and END events will show their values in the display region directly to the right of the Event Type region. All of their values can be edited. Here are the ranges of values for each event type:	
	Program Change (PCHG)	0–127
	Pitch Bend (BEND)	-8192–8191
	Mono Pressure (MPRS)	0–127
	System Exclusive (SYSEX)	The message in hexadecimal form
	Tempo (TMPO)	1.00–255.00 bpm
	End Point (END)	Any Bar: Beat: Tick setting
MIDI Note Events	Note events have four editable values: Note Number, Attack Velocity (indicated by a “v”), Release Velocity (indicated by a “^”), and Note Duration.	
	Note Number	C-1 – G9
	Attack Velocity	v1 - v127
	Release Velocity	^1 - ^127
	Note Duration	Beats: Ticks
MIDI Controller Events	Controller events have two editable values: Controller Type and Controller Value. Defined controllers are referred to by their names.	
	Controller Type	0 - 127
	Controller Value	0 - 127

**Table 12-2 MIDI-event Value Ranges**

## Soft Buttons on the 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 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 View page determines which events you see when you're on the EVENT page. If you set a parameter on the View page to **Off**, you won't see any of the corresponding events on the EVENT page, and the events won't be executed as they normally are when you scroll through the Event 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 View page are set to **On**, the Event list on the EVENT page will display all of the recorded MIDI data on the selected track(s). If at least one of these view parameters is set to **Off**, the **View** soft button on the 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). If you press **AllOn**, the brackets disappear, and you'll see all event types.



Press the **Toggle** soft button to switch all View filter parameters to the opposite of each one's currently set value.

**AllOff**: Available only on the View page. Sets all View filter parameters to Off. The only item viewable on the EVENT page will be an END point.

**AllOn**: Available on both the EVENT page and the View 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. If you're on the View page, the **Done** soft button returns you to the EVENT page. The view filter settings will be applied to the Event list.

**Done**: On the View page, returns to the EVENT page. On the EVENT page, returns to the COMMON page.

## Song Editor: The TRACK Page

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

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

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 TRACK page for the Bounce function.

Region/Criteria Box

Edit Song: TRACK		<> Track: 1	
Function: Bounce		From : 1:1 To: 2:1	
Dst Track: 1		Events: Notes	
Mode : Erase		LoKey : C -1 Hi: G 9	
		LoVel : 1 Hi: 127	

Locate : 1:1

From To Play Stop Go Done

You will notice that the page is divided into two halves, with the right half being a separate box. This is called the Region/Criteria box. The parameters in this box are used to select the range of events (from a start Bar and Beat to and end Bar and Beat) for modification, as well as which types of events function will affect.

The parameters in this box will generally be the same for most functions. For some functions, however, some 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 box parameters, the Locate parameter is also found on each function.

Since these parameters are common to most Track 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 page. Two functions, (Quantize and Reference Quantize) have a unique parameters in their Region/Criteria boxes. 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 for every function on the TRACK page.

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 Box 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 adjusted 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.

### High Key (Hi)

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.

### High Velocity (Hi)

The Hi 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 value of Hi are not 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.

### High Value (Hi)

Hi defines 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 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 parameter in the Region/ Criteria box and then press the **Play** soft button. During playback, every time you press **FromTo**, the K2661 updates the value of From to match the current playback location. Position the cursor over the To parameter to change the value of To in a similar fashion.

If you haven't selected either the From or To parameter, pressing **FromTo** during playback updates From or To—or both—depending on the current playback location (the value of the Locate parameter) at the time you press **FromTo**. If you press it while the Locate value is earlier in the song than the current To value, the K2661 updates the From value. If you press FromTo again (without stopping playback) while the Locate value is later than the current From value, the K2661 updates the To value.

**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.

## Song Editor: 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
  
```

### 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 box 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 box 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.

## 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.

Edit Song: RHYTHM		<> track: 1	
Function: Bounce		From : 1:1	To: 2:1
DstTrack: 1		Events: Notes	
Mode : Erase		LoKey : C -1	Hi: G 9
		LoVel : 1	Hi: 127
Locate : 1:1			
From To	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 box 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.

## 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: RHOK (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.

## 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: RHOK (Effects all tracks)
Function: Delete From : 1:1 To: 2:1

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

## 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.



**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. Press the **Plus/Minus** buttons simultaneously for quick selection of grid values.

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

The Swing percentage is applied to the quantize grid. **0%** 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 Box

**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.

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

**RefSong:** 1 NewSong

Use this parameter to select a song from which you'll then select a reference track. Select the song containing the desired reference track as the RefSong value.

**RefTrack:** 1 – 16

Use RefTrack to select the track to be used as the reference track. The timing of this track's note events is used as the quantization grid that's applied to the current track when you use Reference Quantize.

**Timing:** Off / 1% – 100%

The Timing parameter determines how much the selected note events are moved toward grid locations. If set to **Off**, no aligning of previously recorded Notes to grid locations occurs. If set to **100%**, every recorded note event gets aligned to the closest grid location, defined by the timing of note events from the reference track. Notes are moved up to the limit set by the value of the Quant parameter—for example, a value of **50%** for Quant means that notes get moved half the distance from their current location to the nearest grid point.

**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 want to reference only the note event timing when using Reference Quantize. Set Velocity to **100%** to have the velocity values of the note events being quantized exactly match the velocity values of the note events on the reference track. A setting of **50%** changes the velocities to values halfway between the velocity values originally recorded and the values on the reference track.

### Region / Criteria Box

**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.

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.

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

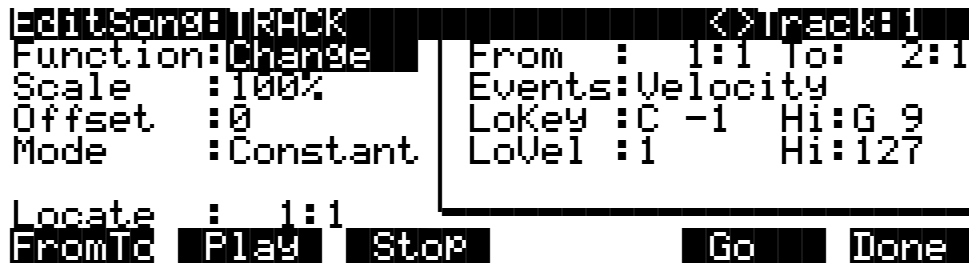
**Semitone:**  $-128\text{ST} - 127\text{ST}$

An increment of one semitone represents a change of one MIDI Note number. You can transpose Note events only within the range of MIDI Note numbers 0 to 127.

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 between each controller event.



**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.



## 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 excess of MIDI data.

```

EditSong: [RHCK] [Track:1]
Function:Thin
Percent :50%
From : 1:1 To: 2:1
Ctl :MPress
LoVal :0 Hi:127
Locate : 1:1
FromTo 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.

## 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: [RHCK] [Track:1]
Function:Remap
Old :MWheel
New :Volume
From : 1:1 To: 2:1
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.

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

```

**SrcSong:** 1 NewSong

The Source Song parameter is set to the ID and name of the song in RAM 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 box 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.

## Song Editor: The STEP Page

```

EditSong:STEP                                     <>Track:1
DubMode :Erase
Duration:1/16
Modifier:None
GateTime:75%
Velocity:Played
Locate   : 1:1000
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

The Step Editor allows for nonreal-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 box 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 box 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 example, if the Duration is set to 1/8, you can quickly enter a quarter note by holding the note and pressing **Step>** once. This adds an extra eighth-note duration to the eighth-note duration created by triggering the note. Enter a half note by pressing the **Step>** button three times (adding three eighth-noted durations). Note that if you are using this method and the Gate Time parameter is set to less than 100%, the truncated duration is applied to the step created by triggering the note. All other steps get the full

duration. 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 box on the right will change to show only the notes you are still holding with their longer durations.

If you make a mistake while entering notes, you have two ways to fix it. With Dub Mode set to **Merge**, if you're still holding the note, you can press the **<Back** soft button. If you set the Dub Mode to **Erase**, you can press the **<Back** button to erase the previous event, 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 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 Song-mode MAIN page, and if the DubMode in the Step Editor is changed, the Mode parameter on the MAIN page 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 parameters.

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 **Plus/Minus** buttons is one method of selecting a step's duration.

Use the alphanumeric buttonpad to quickly select a note duration and modifier whenever the cursor is positioned over the Duration parameter. These are the buttonpad 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 choose to use a modifier.

Use the triplet modifier for a resulting duration  $2/3$  of the value set in the Duration parameter. (For example, 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. (For example, 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 **Plus/Minus** buttons to increment this value in 20% intervals.

**Velocity**

Each note event entered in 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 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>** moves 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 COMMON page when you are finished step editing.

## Song Editor: The 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 normal song data, but could include only the arrangement data, by which the pieces of a composition are assembled. 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 ARRANGE 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 use them.

When a song containing an arrangement is selected as the CurSong on 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 Status 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 display 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** buttons to select the different steps in your arrangement. StartStep on the COMMON page (see page 12-41), 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. Tracks displaying an (M) for the Mutes parameter are muted during playback of the song in the arrangement.

### Xpose

Each step in an arrangement can impose a transposition on the song selected for 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 COMMON page of the step's song. See page 12-44 for a discussion of drum tracks.

### Times

Times sets the number of times that an arrangement step plays before the arrangement moves on to its next step. You can set this value to play the step 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.

## 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 (that is, return to step one after playing the song's final step) set the final step's mode to stop.

## 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. See page 12-64 for more information.

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 12-44), 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.

The control setup's Sync parameter (on the COMMON page in the Setup Editor) affects how steps begin—immediately, or at the beginning of the next measure—when you trigger the steps from a key. See page 7-34 for a description of the Sync parameter.

### LoKey

The first parameter, LoKey, 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 LoKey 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 each song's 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 LoKey. The HiKey must be a higher note value than the LoKey in order to trigger any step from the keyboard.

### Latch

If Latch is set to **Off**, any step you trigger with keystrokes continue to play only as long as you hold down the key that triggered it. Set Latch to **On** if you want to trigger the step with a keystroke, and have it continue playing after you release the key. In this case, the step plays for its entire length, or until 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 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.