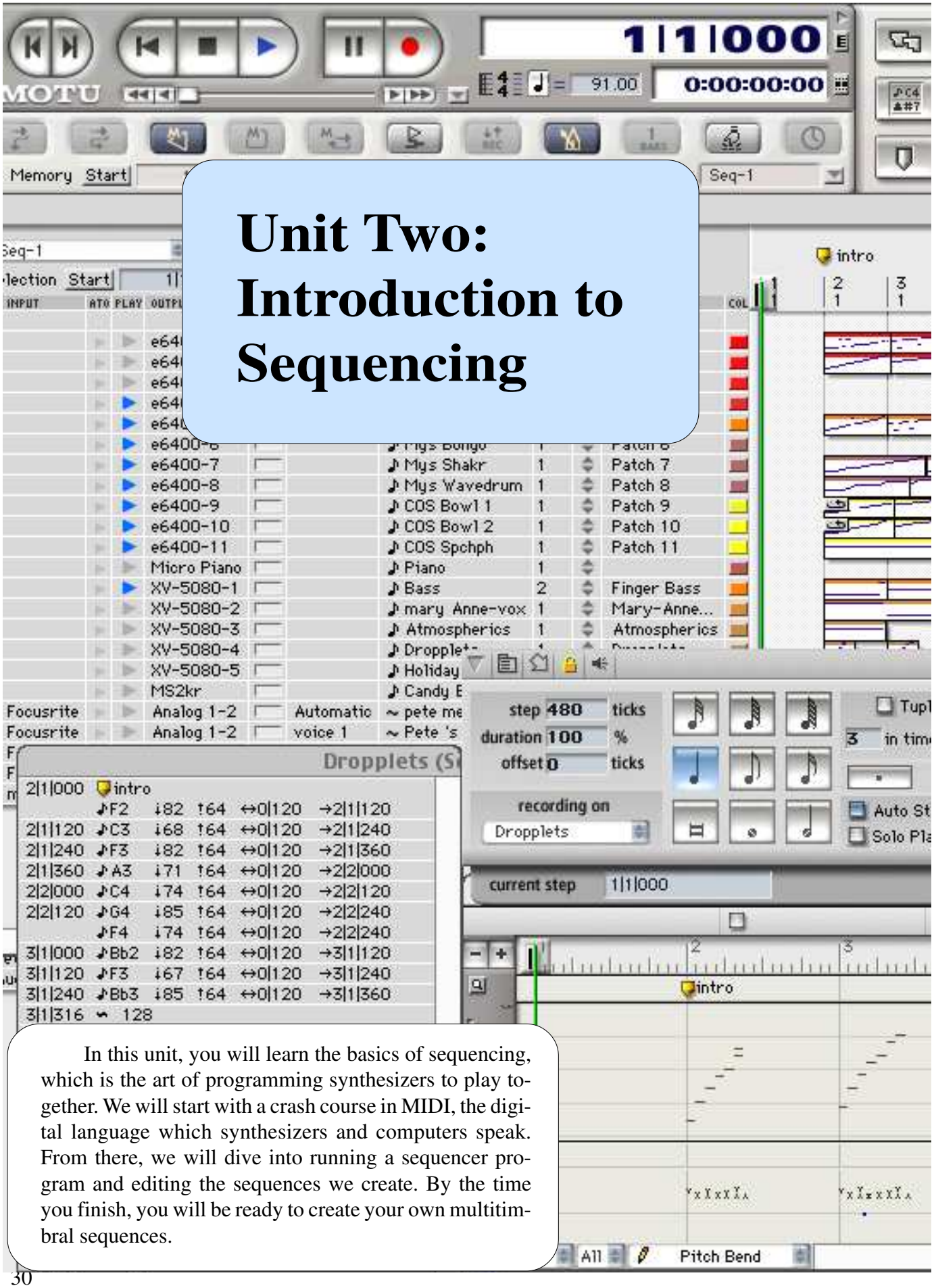
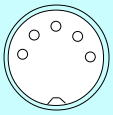


Unit Two: Introduction to Sequencing

In this unit, you will learn the basics of sequencing, which is the art of programming synthesizers to play together. We will start with a crash course in MIDI, the digital language which synthesizers and computers speak. From there, we will dive into running a sequencer program and editing the sequences we create. By the time you finish, you will be ready to create your own multitimbral sequences.

LESSON
5

Functional MIDI



After completing this lesson, you should be familiar with the following concepts:

- MIDI is a digital computer language used by synthesizers, tone modules, mother keyboards, keyboards and many other MIDI devices to communicate by sending MIDI messages.
- Common MIDI messages include note on and off messages, continuous controller messages, and pro gram change commands.
- There are three kinds of MIDI jacks which are found on MIDI equipment: IN, OUT, and THRU. Each jacks serves a specific function (accepting incoming data, putting out data, and relaying incoming data respectively) and cannot take the place of any other jack.
- These 5-pin MIDI jacks are connected with MIDI cables which also have 5 pins.
- MIDI messages are sent on one of 16 possible channels. MIDI devices transmit on a specific channel which the user can set, and the receiving device must be set to a matching receive channel if the two devices are going to communicate properly.
- Most keyboards offer a local setting. When switched off, the keyboard will still send MDII signals out keyboard's OUT jack, but the keyboard will no longer play the sounds in that keyboard. The inter nal sounds will still respond to incoming MIDI messages from other devices, however.
- Many keyboards and tone modules offer a setting called Omni. When switched on, the keyboard or tone module will respond to any incoming MIDI messages, even if the incoming message's channel doesn't match the receive channel on the keyboard or tone module, it will still respond to the messages.
- MIDI merger boxes allow you to merge the outputs of two MIDI devices to one stream of data.
- Many modern synthesizers are multitimbral, which means that a single keyboard or tone module can take the place of 8-16 monotimbral keyboards or tone modules.

Glossary for this Lesson:

Continuous Controller- A MIDI message which carries the information of which controller was used (1-128) and where it was set (1-128). Common continuous controllers include volume, modulation wheel, and panning messages.

Channel- A setting which allows you to keep MIDI messages separated. Only devices set to receive a particular channel will respond to messages transmitted on that same channel.

IN- A MIDI jack which accepts incoming MIDI messages and data from another MIDI device.

Local- A setting which determines if a keyboard is connected to its internal sound source directly.

MIDI- Musical Instrument Digital Interface. A communications protocol used by all modern electronic musical instruments to share performance data.

MIDI Cable- A cable which has circular connectors with five pins. Used to connect MIDI jacks on different devices together.

MIDI Jack- A five-pin jack.

MMA- The MIDI Manufacturer's Association. The MMA must approve all changes made to the MIDI Specification, which determines how all MIDI devices must operate.

Mother Keyboard- A keyboard which has no sound source of its own, but just creates MIDI messages which can be used with other MIDI devices.

Multitimbral- The ability of some modern keyboards and tone modules to act like several different, independent synthesizers. They can produce several different sounds at once.

Note Off- A MIDI message which tells a keyboard or tone module to stop sounding a particular note.

Note On- A MIDI message which tells a keyboard or tone module to start sounding a particular note with whatever sound is currently selected.

Omni- A setting on many keyboards and tone modules which allows them to disregard channel information and respond to all incoming MIDI messages regardless of what their receive channel is set to.

OUT- A MIDI jack which sends out data created in that particular device.

Part- A section of a multitimbral synthesizer. Each part has its own receive channel and can have its own unique sound.

Program Change- A message which causes a tone module, keyboard, or other MIDI device to switch to a different program, sound, or setting.

Release Velocity- A measure of the speed at which a key on a keyboard was released. Few modern keyboards can create these messages.

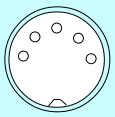
THRU- A MIDI jack which relays information which came in the IN jack to another MIDI device. The THRU jack passes a copy of whatever came in the IN jack back out so that it can be routed to another device.

Tone Module- A synthesizer with no keyboard of its own. It must be played via MIDI note messages created by other MIDI devices.

Velocity - A MIDI message which describes how hard a key was played.

LESSON 5

Functional MIDI



MIDI is a very complex and expansive subject. Today's lesson is intended only as an introduction to working with MIDI. This lesson will give you only the bare essentials of working with MIDI. MIDI will be the subject of an entire course later on.

Of Acronyms And Agreements

MIDI is an acronym. Its letters stand for Musical Instrument Digital Interface. The term MIDI is used to talk about both the hardware parts of these communications and the data which flows from device to device. What's really wonderful about MIDI is that all devices which use MIDI jacks work together even if they weren't made by the same company. In the world of music technology, there are often many different standards and connections for different devices. For personal computers, there are currently more than ten different connectors and formats for connecting something as simple as a printer. In the world of music technology, it is a different story. All devices which need to share performance data share it using MIDI. All MIDI devices have to conform to the data format laid out by the MIDI Specification. This document was created by and is still overseen by the MMA (MIDI Manufacturer's Association). Manufacturers must adhere to the MIDI specification when designing equipment. This sometimes adds frustration for manufacturers, but the end result is that all MIDI devices work together.

MIDI (pronounced mid-ee) allows synthesizers, tone modules and other devices to communicate different information. One of the most important things to understand about MIDI is what is not transmitted from one device to the next. MIDI cables do not carry sounds like instrument or microphone cables do. Instead MIDI transmits performance information from one device to the next.

There are literally hundreds of different kinds of messages which go down a MIDI cable, but for the purposes of this lesson, we will just learn about a few of the most important MIDI messages.

Note On, Note Off

The most common MIDI messages are those turning a particular note on or off. A note on message tells a synthesizer or tone module that it should start playing a particular note. A note off message tells the synthesizer or tone module to stop playing a particular note. Once in a while, a synth or keyboard will receive a note on message without a corresponding note off message later on. In this case, the note will keep sounding on and on, never stopping. For this reason, most sequencers and MIDI interfaces provide you with panic buttons. A panic button sends out a message called all notes off, which turns off any notes which might be sounding. Most sequencers (and DAW programs) send out an all notes off message when you push stop to ensure that no notes continue to sound.

Velocity messages describe how hard a note was played. Velocity messages are measured in 128 steps from 0 to 127. A very few keyboards can generate release velocity messages, which can sense how quickly keys are released.

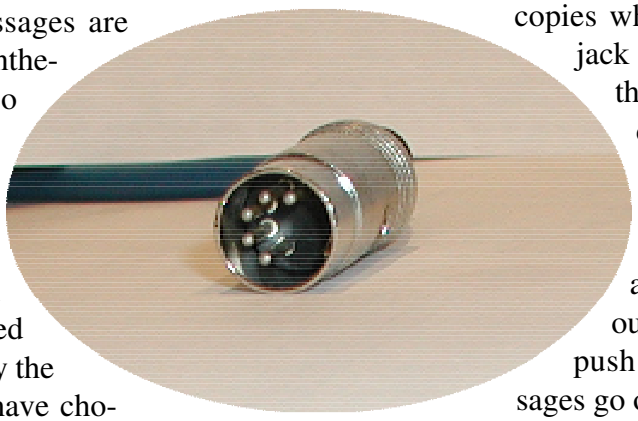
Are You In Control?

Another common MIDI message is a continuous controller message. Examples of continuous controllers include hold pedals, the modulation wheel, volume controls, and even breath controllers. Each controller is assigned a specific number. There are 128 possible controllers, but not all of them have been assigned by the MMA. Some of the most important controller numbers which you should memorize include the modulation wheel (#1), volume (#7), pan (#10), and the sustain pedal (#64). The controller's number is followed with a specific value. For instance, when you push the damper pedal down, you create a controller message #64 followed by a value of 127. When you let the pedal up, you send controller 64 with a value of 0. Although these controllers, and many others are designated to do a specific thing, most modern synthesizers can use incom-

ing controller data to change anything about the sound you wish, from adjusting the cutoff frequency to changing attack time.

Get With The Program

Program change messages are messages which cause a synthesizer or tone module to change the sound they are set to. DAWs can send out program changes so that you can pick the sounds you want to use right from your DAW. More advanced DAWs will actually display the name of each sound you have chosen. There are only 128 possible program change commands which can be sent, numbered from 0 to 127. However, you can also send bank change commands, which allow you to select a different bank of 128 sounds on synthesizers and tone modules which have more than 128 patches.



The Hardware

MIDI jacks have 5 small holes in a semicircle which accept the 5 pins of a MIDI cable. You can see pictures of both a MIDI jack and a MIDI plug on this page. When plugging into a MIDI jack, it is important to align the pins before plugging the cable in, because the jack can become scratched, and worse yet, the pins of the cable can become bent or broken. At $1-3 per foot, MIDI cables aren't cheap to replace. With a little care, they last a long time.

There are three kinds of MIDI jacks: IN, OUT, and THRU. Each jack serves a very specific function. The IN jack only allows MIDI signals to come in. The OUT jack only puts out data which was created *in that device*. OUT jacks can only be connected to IN jacks. They can't be connected to another OUT or THRU jack. The THRU jack copies whatever has come in the IN jack and puts an exact copy out the THRU jack. THRU jacks can only be connected to IN jacks.

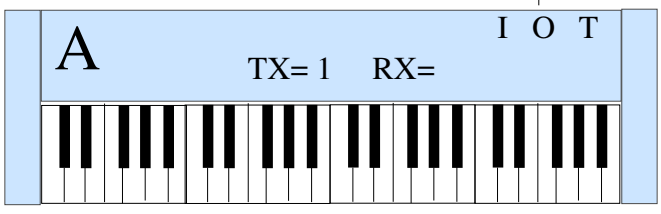
When you play the keys of a keyboard, note on and note off messages come out the OUT jack. When you push the pedal, controller messages go out the OUT jack.

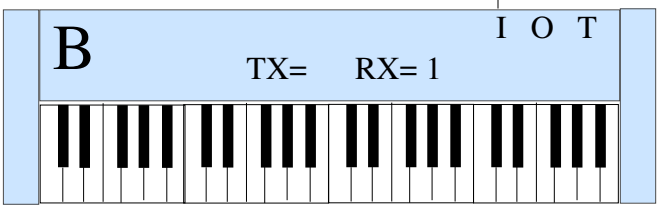
Tune In To This Channel

When a TV station broadcasts your favorite show, they send out waves at a specific frequency. We call this specific frequency a channel. If the TV station is broadcasting on channel 2, you have to tune your TV to channel 2, or you will not see what the TV station is broadcasting. MIDI uses channels as well. In the world of MIDI, devices like keyboards can transmit *and* receive. This means that each MIDI device in the studio has a setting for the transmit channel, and a separate setting for the receive channel. MIDI allows 16 different channels per MIDI cable.

Using MIDI

MIDI allows us to do many wonderful things. One of the most basic things MIDI allows us to do is to play one keyboard from another keyboard. To make one keyboard play another, we first have to make the correct connections between the keyboards. The OUT jack on keyboard A must be connected to





the IN jack on Keyboard B. Once the hardware connections have been made, we have to set the receive and transmit channels. Keyboard A must be set to transmit on the same channel that Keyboard B is receiving on. You can see this configuration at the bottom of the last page. Keyboard A's receive channel and Keyboard B's transmit channel don't matter at all in this example, because Keyboard A is not receiving anything, and Keyboard B is not transmitting anything.

Once you have created this configuration, when you play Keyboard A, it will play as usual, but Keyboard B will also play. The keys on Keyboard B won't actually move, but sounds will come out its output jacks. This technique is really cool to build thick keyboard sounds, because this way, you can have a piano sound on keyboard A, and a string or choir sound on Keyboard B, and both sounds will play at once.

Using The THRU

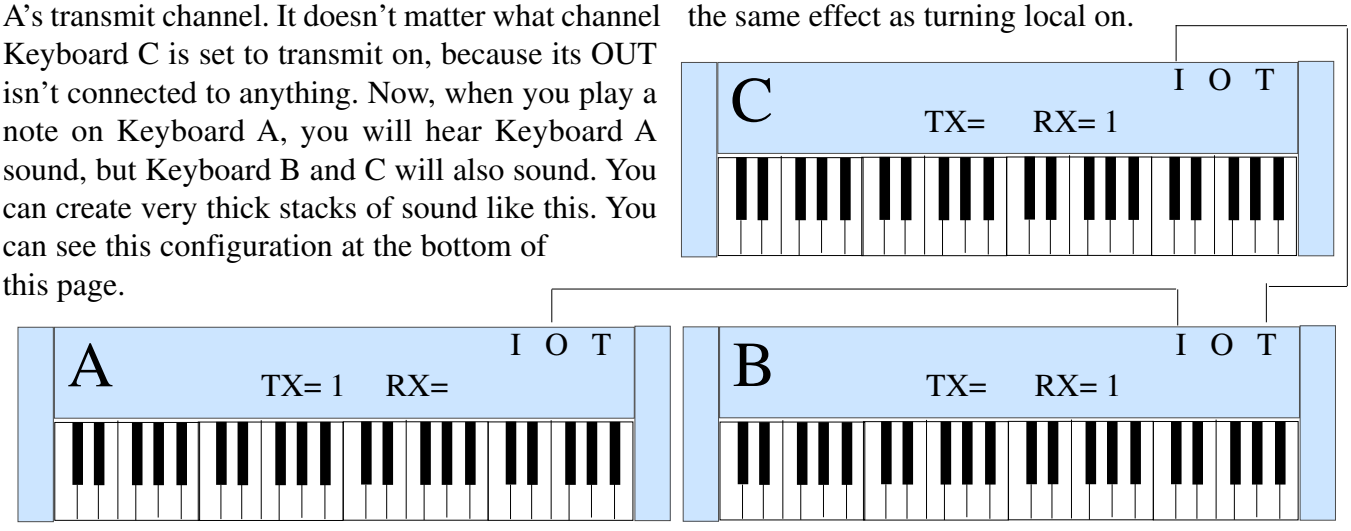
This example was easy enough, but what if you want to play Keyboards A, B, and C by playing Keyboard A's keys? To accomplish this task, we must make additional hardware connections. You could use the setup we already created but connect Keyboard B's THRU jack to Keyboard C's IN. This way, anything that comes into Keyboard B's IN will be passed along to Keyboard C. As before, Keyboard C's receive channel must be set to match Keyboard A's transmit channel. It doesn't matter what channel Keyboard C is set to transmit on, because its OUT isn't connected to anything. Now, when you play a note on Keyboard A, you will hear Keyboard A sound, but Keyboard B and C will also sound. You can create very thick stacks of sound like this. You can see this configuration at the bottom of this page.

Using this configuration, what happens if Keyboard B's receive channel is changed to 2? Keyboard B will no longer respond to notes played on Keyboard A, but Keyboard C will continue to respond to Keyboard A.

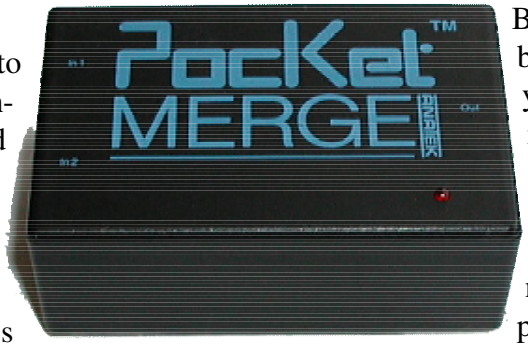
Getting With Local Talent

Most keyboards with MIDI also have a setting called local. Local can either be set to on or off. To understand what local does, you have to think of a keyboard as two parts. A keyboard is really just a mother keyboard and a tone module. You will recall from *Basic Music Technology I* that a mother keyboard is a keyboard which can generate MIDI messages, but has no sounds of its own, and as such, can't make any sound. A tone module is a synthesizer without a keyboard of its own. As a tone module responds to incoming MIDI note messages from other keyboards, it will play.

As long as local is set to on, the keyboard makes a MIDI connection to the tone module it houses. When you play the keys, the tone module produces sound. When local is set to off, then this internal connection is broken. When you play on the keys, you will hear nothing, but note messages will continue to come out the MIDI OUT jack. You can use a MIDI cable to simulate the effects of the local setting. If you set local to off, the keyboard will no longer make sound when you play its keys. If you connect a MIDI cable from the keyboard's OUT to its IN, it will have the same effect as turning local on.



Merging Into Traffic



So now you know how to make one keyboard play two others, but what if you and a friend both want to play one tone module at the same time? No configuration of INs, OUTs, and THRUs will solve this problem for you. The problem comes down to the fact that you need to connect the OUTs of two different keyboards to the single IN on the tone module. The solution to this is a MIDI merger. A MIDI merger can have many different shapes and sizes, but the simplest ones are just small black boxes which have two MIDI INs and one OUT. You can see a picture of a simple MIDI merger to the right. Although MIDI mergers are very simple devices, they tend to be fairly expensive because the MIDI signals have to be rearranged and kept in order when they are combined. A small MIDI merger like the one picture here typically costs $70-$100, while mergers with more inputs cost between $300-$1000.

To use a merger to allow two keyboards to play one tone module, the OUTs on the two keyboards are connected to the two INs on the merge box, and the merge box's output is connected to the IN on the tone module, as shown in the picture at the bottom of this page. Of course, the tone module's receive channel must also be set to match the transmit channel on the two keyboards.

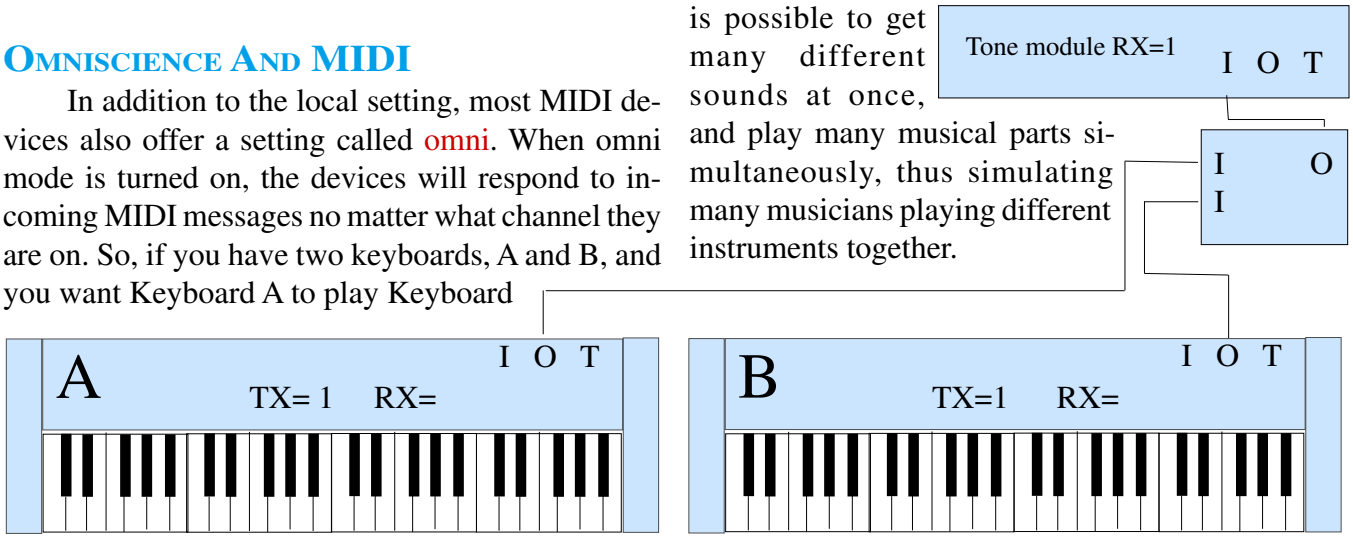
Omniscience And MIDI

In addition to the local setting, most MIDI devices also offer a setting called omni. When omni mode is turned on, the devices will respond to incoming MIDI messages no matter what channel they are on. So, if you have two keyboards, A and B, and you want Keyboard A to play Keyboard B, you merely have to turn on Keyboard B's omni setting, and then you don't have to worry about matching MIDI channels. Keyboard B could be set to receive on channel 16 and Keyboard A could be set to transmit on channel 3 and Keyboard A will still play Keyboard B as long as Keyboard B's omni setting is on.

Getting Multitimbral

Many modern synths and tone modules are multitimbral. This means that they are capable of producing more than one sound at once. Multitimbral synths have different sections called parts. Each part has its own receive channel, and can produce its own sound. This is a very important feature, because it means that a single tone module can act like several synthesizers at once. A multitimbral tone module can receive different note messages on different channels, and each channel can be set to a different sound. Using two different keyboards merged into a single tone module, if each keyboard is set to transmit on a different MIDI channel, each keyboard could play a different sound in the tone module.

Using synthesizers as multitimbral sound sources will become even more important in the next lesson when we learn how to record MIDI messages and play them back. Using only one tone module, it is possible to get many different sounds at once, and play many musical parts simultaneously, thus simulating many musicians playing different instruments together.



Let's Review

1. What does MIDI stand for, and what organization oversees its development?

2. What kind of information travels down a MIDI cable, and what information doesn't MIDI carry?

3. Name the five MIDI messages discussed in this lesson, and tell what each one represents.

4. What are the three kinds of MIDI jacks, and what does each one do?

5. What are MIDI channels, how many are there, and how do we interact with them?

6. What do the local and omni controls do? When would we want to use each of them?

7. How do MIDI channels relate to multitimbral synthesizers?

8. What is needed to combine the signals from two MIDI out jacks?

Words To know:

Continuous Controller
Channel
IN
Local
MIDI
MIDI Cable
MIDI Jack
MMA
Mother Keyboard
Multitimbral
Note Off
Note On
Omni
OUT
Part
Program Change
Release Velocity
THRU
Tone Module
Velocity

DID YOU KNOW?

Shortly before MIDI arrived on the scene in early 1983, there were several different systems which were trying to gain dominance in the marketplace, but every manufacturer had their own system, which was completely incompatible with everyone else's system. However, before that time, there was another standard for transmitting performance information. The system was called Voltage Control, and raw voltages were sent from one instrument to another (and often from one part of an instrument to another part of that same instrument) to communicate performance information. This system was very limited in that only one note could be sent at a time, and machines that could actually record these voltages were few and far between.

Experiments:

1. Locate the MIDI jacks on several different devices. Are these jacks found solely on synthesizers? Do all devices have all three MIDI jacks? Do some devices have more than three jacks? Why is this?

2. Connect two keyboards together using two MIDI cables. What settings do you have to check? Can you make one keyboard play the other?

3. Can you find the local setting on the keyboard you are playing? How can you confirm that you have turned it off or on?

4. Can you find the omni setting on the keyboard you are playing? How can you confirm that you have turned it off or on?

5. Play a multitimbral keyboard or tone module from the keyboard you are playing. Set its different parts to different channels, and to different patches. How can you access the different parts from your keyboard?

6. Create a setup so that two different keyboards can play the same tone module simultaneously.

7. Try sending program change commands to a tone module or to another keyboard. What happens?

Control Freak

Hundreds of different software programs have been created that can do just about anything you can imagine with MIDI signals. One of the most common features of DAW software packages is their ability to route MIDI signals to different devices in a studio. Most studios are configured so that the DAW (and any keyboard connected to the DAW) can instantly play any (or all) MIDI devices in the whole studio. This means that from one mother keyboard, you could play a hundred synthesizers at once! This is a very powerful feature, since it means you never have to leave your chair to play a different instrument.

On the Web:

There are many great sites on the internet that will teach you even more about MIDI. Check out as many as you can!

http://www.midifarm.com/
http://www.midi.com/
http://www.harmony-central.com/MIDI/
http://www.midiworld.com/index.htm
http://www.northwestern.edu/musicschool/links/projects/midi/expmidiindex.html

The MMA's official website:
http://www.midi.org/