Roland

What's MIDI

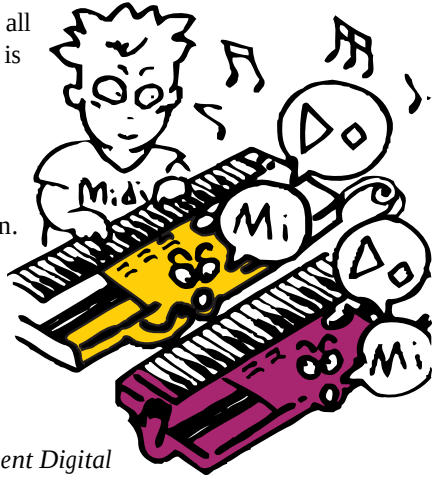
MIDI GUIDEBOOK

A System Which Greatly Expands the Potential of Musical Composition

1. What Is MIDI?

Conversation Between Musical Instruments

Musical instruments finally started "conversing" in the 80's a revolution in the world of electronic music. Now, one instrument can say to another: "play middle C, with about 60% strength, then play E4, a little louder". The second instrument "listens" to this dialogue, and plays the notes in turn, as long as it can understand the language used. The "language" that is now shared by all electronic musical instruments is called "MIDI" and is featured in this document.



Even though the instruments can now "converse", they still have no will of their own. In fact, the only link in the chain with its own will is you, the musician. Then the instruments interpret your performance by "translating" it into MIDI. "MIDI" is considered to be "the language which is used to send 'performance information' from one instrument to another".

For your reference: *MIDI* stands for *Musical Instrument Digital Interface*.

Applications of MIDI

MIDI is widely known as a language between musical instruments.

MIDI has expanded in such a short period of time, because it can be applied to many different situations. Today it is also very common to use MIDI to connect a MIDI device such as a Keyboard Controller to the Computer.

MIDI is a universal term. Although there are many different languages in the world, such as English, French, and so on, MIDI has only one language, and can therefore provide communication between Japanese and American instruments, or German and Italian instruments.

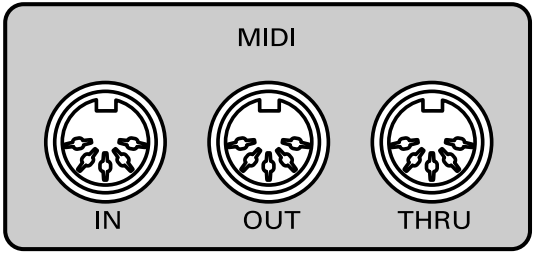
MIDI can even be used to "talk" to instruments from different manufacturers.

Furthermore, MIDI can be applied to many kinds of instruments. In this way it is possible for an electronic piano to "talk" to a synthesizer, a drum machine or a Computer. Consequently, MIDI has an enormous potential for growth and development, as you shall see in this document.

What Makes A MIDI Conversation Possible

Just how is MIDI used to provide this "conversation"? To explain this, look at the back of your instrument. There are several connectors, including a MIDI IN, a MIDI OUT and often a MIDI THRU. These are the ears and mouth of the MIDI conversation, and are essential for any MIDI communication. The actual "MIDI conversation" travels from the MIDI OUT of one instrument, to the MIDI IN of another, by connecting only one MIDI cable between these two sockets. The cable used has five pins, matching perfectly the five holes in each of the sockets.

It is also frequently used the MIDI/Joystick Cable that allows to connect a MIDI device to the PC.

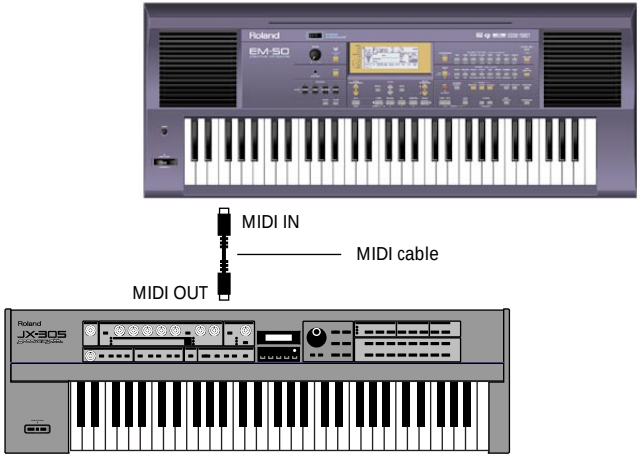


So why do we need three sockets? (MIDI IN, MIDI OUT and MIDI THRU)

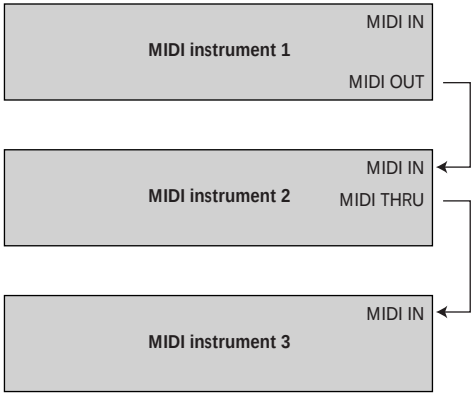
MIDI IN —is for "listening" to a MIDI conversation, that is, it is the entrance for MIDI information.

MIDI OUT —is for "speaking", to send the conversation from an instrument, and is therefore the exit point for MIDI information. See the illustration for a simple MIDI OUT to MIDI IN connection.

MIDI THRU —Retransmits a copy of the MIDI messages received via MIDI IN.



As you can see, MIDI IN and MIDI OUT are relatively easy to understand, and use, however, MIDI THRU is comparatively more difficult. MIDI THRU is similar to MIDI OUT: it does “speak” to the next instrument, however, it doesn't provide its own conversation, it merely “repeats” anything that is heard by the MIDI IN. For example, the picture shows the flow of information through a possible set-up of three keyboards.



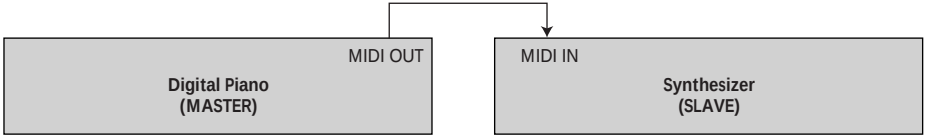
In this set-up, MIDI information is fed out of the MIDI OUT of keyboard number 1, and into the IN of keyboard number 2. It is also passed on to the IN of keyboard number 3, via the THRU of keyboard number 2. This way, the information from keyboard number 1 can control both keyboards 2 and 3.

Such a combination of more than 2 instruments is referred to as a “MIDI system”. Of course, without the MIDI THRU, this set-up could not become a MIDI system, and therefore it is easy to understand the use of MIDI THRU in expanding a *MIDI system*.

One-Way MIDI Conversation

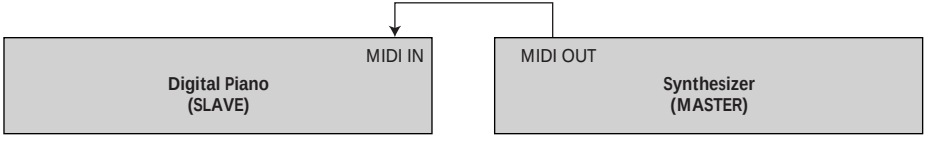
MIDI information is sent from MIDI OUT to MIDI IN, or from MIDI THRU to MIDI IN.

For example, a MIDI "keyboard controller" which has no sound of its own, or a MIDI "Sound module" which has no keyboard, and so on, have fixed roles. They are either the "master" or the "slave". In this situation, the "speaker" is called the master, and the "listener" is the slave. The picture below shows the information from an electronic piano being sent to a synthesizer. In this case the piano is the master, and the synth is the slave.



Play on this instrument.

However, in the following picture the information is being sent from the synth to the piano, and so their roles are reversed. The synth is now the master, while the piano is the slave. Consequently, the connections are very important.



Play on this instrument.

Furthermore, the master/slave relationship is not always one to one, as in the previous 2 diagrams. The number of slaves can be increased, 1:2 or 1:3, etc.

2. The Benefits Of MIDI

Two Instruments in Unison

This section will cover the uses of MIDI in performance, expression, and other facets of the music making process. Firstly, we shall discuss playing 2 instruments in unison.

Suppose you connected the MIDI OUT of a digital piano to the MIDI IN of a synth. If the synth is set to play a string sound, playing the piano will cause the synth to play exactly the same part on a string sound. The result is a very BIG sound, simulating the pianist being doubled by an orchestra.

Combinations of Different Kinds of Instruments

As was mentioned earlier, it is possible for different kinds of instruments to "converse" through MIDI. Naturally, we expect keyboards to "talk" to each other, but what would a piano say to a drum machine. It is hard to imagine just what the result might be.

Think at what could happen if you connect a piano to a drum machine: the piano as the master, the drum machine as the slave. You may choose the sounds (assign them) so that by playing the key "C" on the piano, a bass drum sound is heard. Then perhaps a "D" would play the snare, and a "G" a Tom, etc. Then, rather than using drum sticks, you can play the "drums" by pressing the appropriate keys on the keyboard.

What happens if the drum machine is the master and the piano is the slave. This time, if the drum machine is playing a pre-programmed rhythm, the piano will play the same part. However, whenever the drum machine plays a bass drum, the piano will play a "C", etc. Just the reverse of the previous situation.

Note: *A drum machine has limited note capability, and can usually only remember a few songs, so it would be more practical to use a Sequencer, or a Computer, for this type of automatic performance (as we shall discuss later).*

This combination of a piano and a drum machine is only one example. Using MIDI performance information it is possible for many types of instruments to "converse" via MIDI.

A Sequencer System

One of the most powerful uses of the MIDI language is to include a sequencer. Performing in unison, as we mentioned earlier [↗], can provide some excellent sound possibilities, but still limits the performance to that of the master keyboard.

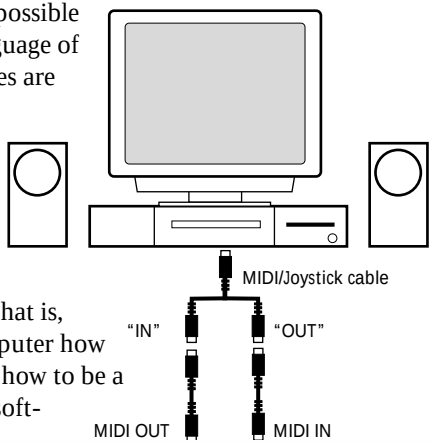
A sequencer on the other hand, is capable of remembering several instrumental parts, making it possible to use a variety of sound sources. It then becomes a simple matter to construct an entire arrangement, or orchestration.

For example, a sequencer as the master can control a synth for melody, a piano for chords, another synth for bass, and a drum machine, all at the same time. The data for this 4 part performance being stored in the sequencer. Then, as the data is sent to all 4 instruments, the automatic performance begins.

As you can understand, a sequencer can be seen as multi-track recorder that enables you to record your own songs.

Computer Music System

Computers did not seem to relate to music very well before MIDI. Certainly it was hard to understand any "conversation" between computers and musical instruments. Thanks to MIDI it is now possible to build small boxes that can "interpret" the language of computers into MIDI, and vice versa. These boxes are called "interfaces".



Today most of the sound cards have a built-in MIDI interface. So sometimes you won't see any Box: in fact, in this case, the connection can be made directly between the MIDI device and the PC through a *MIDI/Joystick* cable.

All computers use what we now call "software", that is, information on computer disk that tells the computer how to behave. A "software" may teach the computer how to be a sequencer, while another could be "sound edit" software, teaching the computer how to modify the sounds of a synthesizer.

The picture shows an example of a basic computer music system with sequencer software. The advantages of such system are the huge computer memory available, in order to store many songs, or long songs, and the size of the computer display, making it easy to visually check all the data for a performance.

MIDI Systems Expanding Into Various Fields

We have already discussed a variety of MIDI systems, but there are still many more possibilities. For example, when changing a sound in a MIDI system it is also possible to switch a variety of effects on or off, delay or Reverb for example. A MIDI sequencer can also be synchronized with a multi track tape recorder (MTR), expanding the number of available tracks or with a Light System. The concept is extremely versatile, and the future is still wide open.

For further information about MIDI future, you can visit the Internet site of the *MIDI Manufacturers Association*: For that purpose, if you have Internet connection, click on the picture below.

3. Information Transmitted Via MIDI

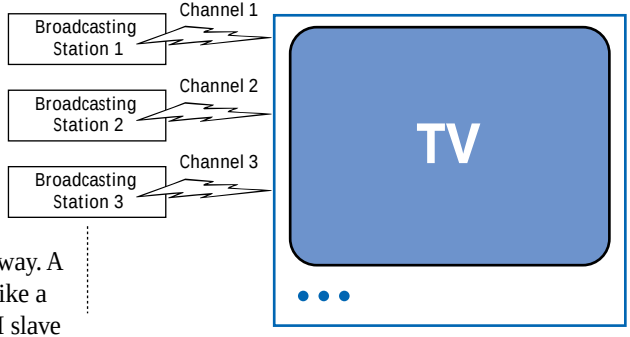
3.1 MIDI Channels and Modes

Channels

MIDI is able to transmit a variety of information from a master instrument to a slave instrument. Either one part, or multiple parts can be transmitted via a sequencer system.

For this purpose, MIDI includes 16 "channels". All the different types of MIDI conversation can be sent via these 16 channels separately. So, with one MIDI cable, 16 parts can be played at the same time.

The concept of MIDI channels is similar to that of TV broadcasting in a sense. Many TV stations are transmitting their respective programs at the same time. This means that your TV antenna must be receiving all these channels at once. However, since all the stations are transmitting on different channels, you simply need to select the channel you wish to watch.

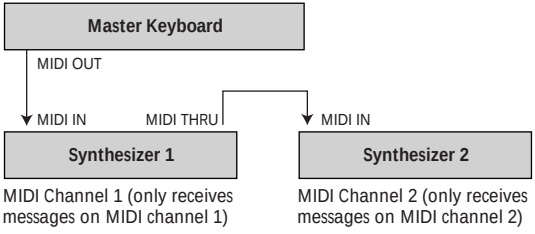


Select the channel you want to watch.

MIDI channels behave the same way. A MIDI master instrument is just like a TV broadcasting station. A MIDI slave instrument is like your TV receiver, although rather than transmitting through the air, MIDI is transmitted via cables.

The MIDI master instrument can then decide which channel to transmit, rather like choosing which TV station it wants to be. On the other hand, the slave instrument can behave like the TV set and choose which MIDI channel it wants to "hear". Even if information is received on all 16 MIDI channels, the slave will only "hear" the chosen channel, just like the TV set.

For example, if the instruments are set up as in the picture below:

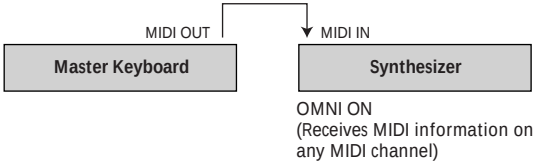


Synth 1 only receives the information from channel 1 through the MIDI cable. Synth 2 only receives channel 2. So that if the master keyboard is sending on MIDI channel 1, only synth 1 will respond. Similarly, if the master transmits on MIDI channel 2, only synth 2 will respond. Even though synth 1 is obviously receiving the information, it cannot respond because it is switched to a different channel. In other words, for this system to work correctly, YOU MUST CAREFULLY SELECT MIDI CHANNELS.

With this concept of MIDI channels, it is possible to play 16 different parts on 16 instruments, via a sequencer, and all originally down one MIDI cable.

Modes

As we mentioned before, it is necessary to use the same MIDI channel if a master and a slave are to play in unison, but there is another way to make the slave receive the information. This mode is called “OMNI ON”, and it forces the slave to respond to all 16 MIDI channels at once. In a simple set-up of one master and one slave this becomes very useful, because the slave doesn't care which channel the master is using, it will “hear” them all. However, when using a sequencer playing on a variety of different channels, it is impossible to separate the information. So then the slave must be in the OMNI OFF mode, so that it will only receive information on the chosen channel.



There are also *POLY* and *MONO* modes in MIDI, which decide whether the information is to be sent as “monophonic” information (one note), or “polyphonic” information (more than one note). Keyboard instruments such as pianos and synths usually use the POLY mode.

MONO mode is mainly used for MIDI guitar controllers. In this mode, the information belonging to each guitar string is sent on separate channels, dividing the sound source into 6 monophonic synths. Then a BASIC MIDI channel is set, and the next 5 channels are used for the other strings.

To choose whether to use POLY, MONO, OMNI, etc., MIDI includes a *MODE selector*, with 4 possibilities: MODES 1~4.

MODE 1: OMNI ON, POLY —Receives information on all channels and plays all notes.

MODE 2: OMNI ON, MONO —Receives information on all channels but will only play one note at a time.

MODE 3: OMNI OFF, POLY —Receives only on the chosen MIDI channel and plays all notes. Useful with sequencers.

MODE 4: OMNI OFF, MONO —Receives on specific MIDI channel(s), will only play one note per channel. Useful with guitar controllers.

3.2 Main Contents of MIDI Information

MIDI contains many kinds of information, that it can use to transmit performance details from the master to the slave. Such things as when each note was played or released (note on and note off), damper pedal on and off, etc. Also, the information is divided into "channel messages" and "system messages".

Channel Messages

These messages are sent via the individual MIDI channels to specific instruments in the system, and therefore only affect those instruments receiving that channel. Channel Messages include note on/off, damper pedal on/off, pitch bend, etc. These Channel Messages are further divided into 2 categories, "Voice Messages" and "Mode Messages".

a) Voice Messages

Note Information —Note Information is the most basic, it simply says which key was pressed, when it was pressed, and when it was released.

Program Changes —A Program Change is used to cause the slave to change its sound. Synths, electronic pianos, and sampling machines have memories full of many sounds. With these Program Changes the musician can choose which sound to use. It is also possible to switch the memories of MIDI effects devices.

Control Changes —A Control Change can add subtle nuance to a performance, things like modulation (i.e., vibrato and tremolo), hold (damper) pedal, soft pedal, and Portamento. These messages are not used in all MIDI instruments. An electronic piano will send and receive damper pedal information, but it certainly does not require Portamento. So even if the master instrument has Portamento, the piano will not respond to that information. To find out which controllers a specific instrument will respond to, refer to the "MIDI Implementation Chart" usually found on the last page of the Owner's Manual.

After Touch —Synths and samplers can control vibrato, brilliance, volume, etc., by simply pressing the key harder after initially playing it. This effect is called "After-Touch", and can be used to transmit the same effects via MIDI. Of course, the slave instrument must then decide how to use the Aftertouch information it receives.

Pitch Bend —If the master has a Pitch Bender, the information may be sent via MIDI. Once again, the slave will decide if it wants Pitch Bend, and even how much it wants.

b) Mode Messages

As mentioned earlier, MIDI has 4 MODES, and MODE messages are used to switch the MODE of a slave. Some synths or electronic pianos are in MODE 1 (OMNI ON, POLY) whenever they are switched on, so you will need to switch them to MODE 3 (OMNI OFF, POLY) for use with a sequencer. Because of this, some sequencers will automatically send a mode message to turn the slaves to MODE 3. Turning the slaves power on first, followed by the power of the master, will often achieve the same result.

System Messages

System Messages can be sent no matter how the MIDI channels of the slave or the master are set, since they are used to control the whole MIDI system, that is, every instrument connected by the MIDI cables.

For example, they can be used to synchronize a sequencer and a drum machine, so that they play exactly in time together, or to start and stop the performance, or simply to avoid MIDI system problems.

Also, there are messages called "System Exclusive Messages", which are messages that are exclusive to a particular manufacturer. So that each manufacturer has an *ID Number* which their instruments will recognize. Any system exclusive data received with the wrong ID Number will be ignored by that manufacturers machines.

With Exclusive Messages it is possible to exchange an incredible number of different kind of information between musical MIDI devices.

Generally you can find information about SysEx messages of your instrument in the MIDI Implementation Manual.

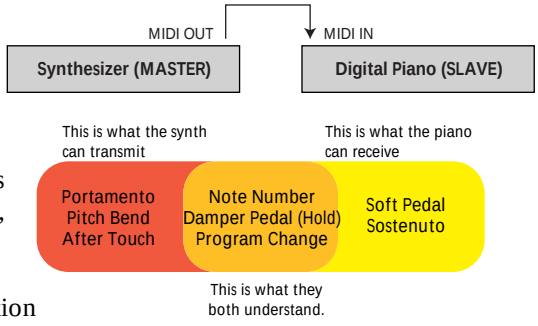
4. Actual MIDI Implementation And Notes

MIDI Implementation Chart

The Effective Way to Check MIDI "Communication"

Although MIDI made it possible for a wide variety of instruments to communicate, this does not mean that all instruments will understand the entire MIDI language.

For example, connecting a MIDI cable from a synth to a MIDI effect may not make the effect behave the way you want. Or using a synth with pitch bend connected to an electronic piano will not force the piano to change its pitch. Simply connecting a MIDI cable is not enough. For the information to be successfully communicated between the two instruments they must both understand that particular piece of information.



As the picture shows, when using a synth as a master to play an electronic piano sound, they can only communicate in area A. The synth has Portamento and pitch bend functions, and sends the relevant information via MIDI OUT, but the piano does not have these functions. Consequently, the piano can neither receive nor perform either of these effects. On the other hand, the piano has a soft pedal, a control the synth cannot understand.

So, in a multi-instrument MIDI system, you will need to check what information each instrument can send and receive. To quickly check this information, each instrument has a User's Manual that will include a "MIDI Implementation Chart", showing the kinds of information that can be sent and received.

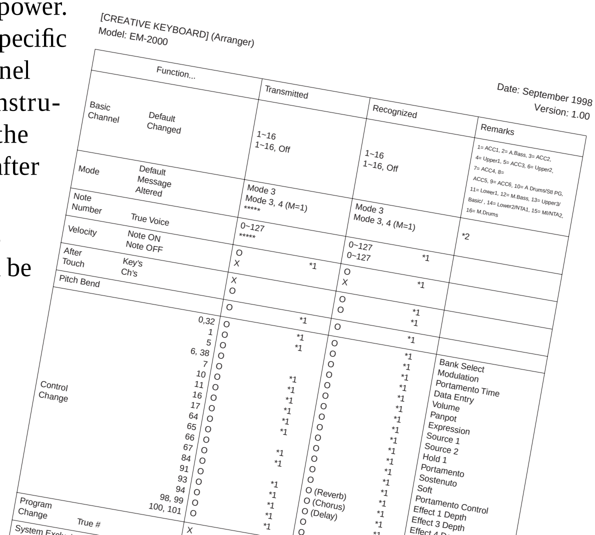
How To Read MIDI Implementation Charts

On the left hand side of the chart, various types of MIDI information names are listed. The transmit/receive column shows the capability of the instrument to transmit or receive that data, using a 'O' for *yes* and a '✕' for *no*. If its capability depends on other factors, the additional information will also be shown.

Basic Channel

There are two columns for Basic Channel: *Default* and *Changed*. *Default* shows which channel is set when you first turn on the power. Most instruments can remember a specific MIDI channel. In this case, the channel you selected last time you used the instrument will be recalled, in which case the chart will say "memory works even after the power is turned off".

The next column says *Changed*. This shows how many MIDI channels can be set, usually 1–16.



[CREATIVE KEYBOARD] (Arranger)
Model: EM-2000

Date: September 1998
Version: 1.00

Function...		Transmitted	Recognized	Remarks
Basic Channel	Default Changed	1~16 1~16, Off	1~16 1~16, Off	1= ACC1, 2= A.Bass, 3= ACC2, 4= Upper1, 5= ACC3, 6= Upper2, 7= ACC4, 8= ACC5, 9= ACC6, 10= A Drums/Stl PG, 11= Lower1, 12= M.Bass, 13= Upper3/Basic/ , 14= Lower2/NTA1, 15= MI/NTA2, 16= M.Drums
Mode	Default Message Altered	Mode 3 Mode 3, 4 (M=1) *****	Mode 3 Mode 3, 4 (M=1)	*2
Note Number	True Voice	0~127 *****	0~127 *1 0~127	
Velocity	Note ON Note OFF	O *1 X	O *1 X	
After Touch	Key's Ch's	X O	O *1 O *1	
Pitch Bend		O *1	O *1	
Control Change	0,32	O *1	O *1	Bank Select
	1	O *1	O *1	Modulation
	5	O	O *1	Portamento Time
	6, 38	O	O *1	Data Entry
	7	O *1	O *1	Volume
	10	O *1	O *1	Panpot
	11	O *1	O *1	Expression
	16	O *1	O *1	Source 1
	17	O *1	O *1	Source 2
	64	O *1	O *1	Hold 1
	65	O	O *1	Portamento
	66	O *1	O *1	Sostenuto
	67	O *1	O *1	Soft
	84	O	O *1	Portamento Control
	91	O *1	O (Reverb) *1	Effect 1 Depth
	93	O *1	O (Chorus) *1	Effect 3 Depth
	94	O *1	O (Delay) *1	Effect 4 D...
	98, 99	O *1	O *1	
	100, 101	O *1	O *1	
Program Change	True #	X		
System Excl...				

Mode
There are 3 columns for Mode: *Default*, *Message*, and *Altered*. *Default* shows the mode the instrument is in when the power is turned on. The *Message* column shows if it is possible for the instrument to receive Mode messages, or after having received a mode message if it's possible to still change modes. The *Altered* column is only used is only used for instruments that can receive messages which will switch the instrument to a special mode, the special mode being explained in this column.

Note Number
This column shows the note range over which the instrument can receive or transmit. It can usually only transmit as many note numbers as it has keys, while it may receive over a much wider range. Some instruments may receive very high or low notes, but play a note in a different octave as a result. This too is mentioned in this column.

Velocity
There are *Note On* and *Note Off* entries for Velocity. This row shows whether the instrument can transmit or receive these two velocity types. It does not represent the actual note on/off capability, just the speed of the respective note on or off. If there is an "X" in either column, it does not mean that the instrument cannot recognize a note on or off, merely that it does not recognize how fast it was pressed or released.

After Touch
This shows if the instrument can receive/ transmit Aftertouch information. The columns refer to channel Aftertouch (*Ch's*= one value per MIDI channel) and polyphonic Aftertouch (*Key's*= a separate Aftertouch value for each note). On most instruments "Aftertouch" refers to channel Aftertouch.

Pitch Bend
Whether or not the instrument can receive/transmit Pitch Bend information.

Control Change
This column shows whether or not the instrument can transmit/receive control change messages, like modulation, damper pedal (Hold), Portamento, etc. These controls are particularly important when connecting two different synths, or a synth and a piano.

Program Change
Whether or not the instrument can transmit/ receive program change information, and what numbers it uses.

System Exclusive
This column indicates what kind of data can be transmitted/received via System Exclusive messages.

System Common
This section is for sequencer-based MIDI systems. It indicates if the instrument will understand MIDI Song Position Pointer, where the instrument can decide from which bar to start playing, and MIDI Song Selection, to decide which song to play, etc.

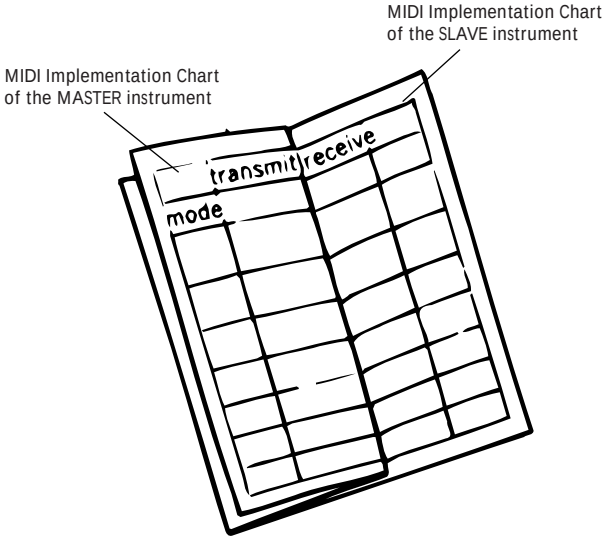
System Real Time
Real Time information refers to the ability of the machine to synchronize via MIDI. If it understands "Clock" information, then it can play in time with other instruments. If it understands "Command" information, it will also know when to start and stop.

Aux Messages
This column is used to display whether the instrument is capable of receiving information that will help it to avoid any MIDI problems.

This area is used for exceptions to the rules. For example, if the functions of an instrument change according to specific parameter settings, they will be indicated here.

As a result, when you connect two or more instruments via MIDI cables, you merely need to check the relevant columns of their respective Implementation Charts to see if the communication you require is possible.

See the picture for an easy way to compare charts. All MIDI Implementation Charts have a standard format, making it easy to arrange the charts as shown, and compare the transmit column of the master to the receive column of the slave. For instance, if either column has a "X" for *Pitch Bend*, then it will not be communicated.



5. Avoiding MIDI Problems

MIDI enables you to add expression to your music in many different ways. Things like after touch, synchronization, and Pitch Bend can substantially enhance the performance. However, this requires an exchange of large amounts of data, which also increases the possibility of error. Unexpected problems can be caused by faulty connections, lack of knowledge, and many other factors. In this section some of the common problems are discussed, and their solutions explained.

MIDI Cable Connections

Obviously, if the cables are connected incorrectly the performance information cannot be successfully communicated. Make sure they are connected from the MIDI OUT of the master, to the MIDI IN of the slaves (or MIDI THRU to MIDI IN).

MIDI and Drum Machines

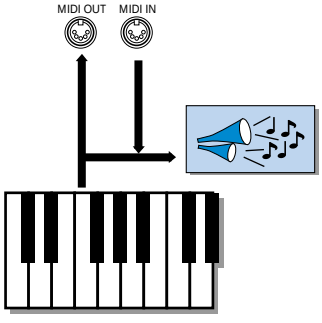
Drum machines can be used in synchronization with sequencers, or as a sound source for a MIDI keyboard, etc. When a synth is the slave, to change the sound you need to send a Program Change message. However, to change the sound of a drum machine you need only to play a different note. "C" for bass drum, "D" for snare, etc.

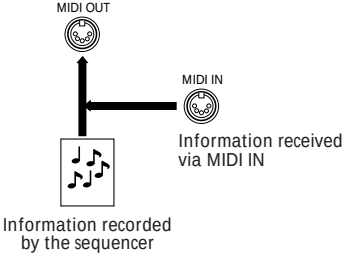
Consequently, in a MIDI system using a sequencer, it would be better to set the MIDI channel of the drum machine to a channel that is not being used, so that it does not receive unexpected note on data. Please note that the standard MIDI channel for drums is 10.

Local Control

Some synths and equipment have a parameter related to MIDI called “Local Control”. When this control is “on” (LOCAL ON) the keyboard of the instrument can play its own sounds, if LOCAL is OFF, the instrument can only send MIDI information out, without playing its own sounds.

The basic structure of synths and electronic pianos is shown in the picture. The information generated by the keyboard usually goes to the sound generating part to play a sound. This connection is broken if the LOCAL switch is OFF. However, even if LOCAL is OFF, the MIDI OUT connection will still operate.





LOCAL OFF is convenient when using sequencers which are capable of sending the MIDI IN information to the MIDI OUT information. This function is called SOFT THRU (see the picture below). In this case, when you record using LOCAL OFF, and SOFT THRU is on, you can listen to the performance of the sequencer as well as your performance on the keyboard.