Oberheim

Matrix-6/Matrix-6R

6-Voice Polyphonic MIDI Synthesizers

MIDI SPECIFICATION

Third Printing – May, 1987
For Matrix-6 and Matrix-6R units with Software Revision 2.13 or later.

Oberheim
A Division of ECC Development Corporation
11650 Olympic Boulevard
Los Angeles, California 90064
USA

Matrix-6/Matrix-6R MIDI Specification - Part No. 950079

Oberheim
MATRIX-6/MATRIX-6R
MIDI SPECIFICATION

Introduction

This document describes the MIDI implementation of the Matrix-6 6-Voice Polyphonic Synthesizer keyboard unit and Matrix-6R rack mount unit. This document assumes that the reader is familiar with both the Matrix-6 or Matrix-6R and the MIDI 1.0 Specification. Unless otherwise noted, this description applies equally to the Matrix-6 and Matrix-6R. The abbreviation "M-6" is used to refer to both. This specification applies to those units using software version 2.13 or later.

Some simple notation is used as shorthand in this document. Hexadecimal numbers are written as two digits followed by an "H". Binary numbers are written as eight digits followed by a "B". Decimal numbers have no suffix. Thus, 19H = 25 = 00011001B. In addition, in a MIDI status byte, the character "b" stands for any hexadecimal digit 0 thru F, corresponding to the MIDI channel 1 thru 16 on which the status byte is being transmitted.

M-6 parameter names are written just as in the Matrix-6 and 6R Owner's Manual, with the addition of an initial letter before the Patch Number to indicate on which page the parameter appears. "P" indicates a Patch parameter which is edited with a particular Patch, "M" indicates a Master parameter which is in the Master Edit section, and "S" indicates a Split parameter that is edited along with Splits. A feature often uses more than one parameter on more than one Page. The user should be careful when setting up the parameters, since the results can be unexpected when one parameter required is a Patch parameter (and thus changes each time a new Patch is selected), and the other is a Master parameter (and thus is constant across all Patches and Splits).

MIDI Modes

The M-6 receives MIDI mode messages as described below.

MIDI Messages Recognized

Channel Mode messages are received on the Basic Channel. An All Notes Off operation is done and the MIDI mode (parameter M01 OMNI MODE and M18 MIDI MONO) is changed to the proper mode.

The All Notes Off message causes all notes which were originally received on the same Channel as the All Notes Off message to have their gates turned off. In the case of the other Mode Change messages, the gates of all Voices will be removed prior to entering the new mode.

When a Mono On message is received, all six Voices will always be made available for Mono Mode, regardless of the value of the last byte of the Mono On message.

Although the Omni On message is recorded in Mono Mode, Omni Mode is always treated as being off by the M-6 when in Mono Mode.

Notes and Voice Assignment

This section describes how incoming MIDI Note On and Note Off messages are handled, how notes are assigned to Voices, and when and how MIDI Note On/Off messages are generated and sent out the MIDI Out port.

MIDI Poly Mode

This section describes the handling and generation of MIDI and keyboard Note On and Note Off messages when the synthesizer is in MIDI Poly Mode. The next section describes note handling when the unit is in MIDI Mono Mode.

If the parameter M01 OMNI MODE is ON, then the M-6 will respond to any MIDI Note messages on any MIDI Channel, 1-16 inclusive. If M01 OMNI MODE is OFF, then the M-6 responds to MIDI messages only on its Basic Channel.

Note On / Key Press Handling

Whenever a note comes in, either from MIDI or the keyboard, the M-6 checks to see if there are any ungated internal Voices available to play the note. If there are one or more voices available, then the unit picks one based on the current keyboard assignment mode (P48 KEYBOARD MODE), as follows:

- If parameter P48 KEYBOARD MODE is ROTATE, the next ungated Voice in sequence is picked.
- If P48 KEYBOARD MODE is REASSIGN, a check is made to see if any of the ungated Voices last played a note of the same pitch as one now being assigned. If so, that Voice receives the note. If no such Voices exist, it proceeds just as with ROTATE.

In either case, once we have a Voice to which to assign the note, the Voice's pitch is updated to the new note's value, the Voice is gated, and both a single and multi trigger are generated on the Voice. If the note came from the keyboard, a MIDI Out message for the same note (with appropriate Attack Velocity) is generated. If the note which has been assigned internally came from MIDI, no MIDI Out message is generated. Note that since the only source of notes on the MATRIX-6R is MIDI, MIDI Out messages are **never** generated for notes which are played by one of the MATRIX-6R's internal Voices.

If all of the available internal Voices of the M-6 are gated, then special handling is required. This handling involves two parameters: P48 KEYBOARD MODE and M13 SPILLOVER. There is a variation REASSIGN mode available on P48 KEYBOARD MODE: REAROB (Reassign-Rob). When this is selected, the M-6 is said to be in "rob" mode. This allows already gated internal Voices to be "robbed" of their current note in order to play new notes. M13 SPILLOVER, which is ON or OFF, indicates that any notes which cannot be played by internal Voices should be "spilled-over" out MIDI: that is, MIDI Out messages should be generated, on the Basic Channel of the unit + 2, to allow an external synthesizer to play the notes. This allows, for example, a MATRIX-6 and a Matrix-6R to be used together as a single, 12-Voice synthesizer.

The interaction between these two parameters, and what MIDI Output results is as follows:

If Rob Is On, a Voice is picked as the rob "victim." A Voice already playing a note with the same pitch as the one we are trying to assign is selected, if any. If not, the "victim" is picked just as in ROTATE.

Once we have a "victim," its pitch is updated to the new note's value, and a multi (but not single) trigger is generated.

If the note being assigned came from the local keyboard, then a Note Off MIDI Out message is generated for the note being robbed, with the "default" Release Velocity of 64, and a Note On MIDI Out message is generated for the new note that "robbed" the previous note. If the note came from MIDI, then no MIDI messages are generated.

If Rob Is Off, we know that we cannot assign the note internally. The value of M13 SPILLOVER is then checked. If spillover is ON, then a Note On message is generated for the note, but it is transmitted on the MIDI Channel two higher than the unit's Basic Channel. If the Basic Channel is 15 or 16, the output Channel wraps around to 1 or 2, respectively. If spillover is OFF, then the Note On message is generated on the unit's Basic Channel. The message is generated whether or not the note came from the keyboard or from MIDI. This allows units to be cascaded to add more Voices to the Spillover chain.

Note Off / Key Release Handling

When a MIDI Note Off message is received or a key is released on the keyboard, the unit finds the matching Note On message or key press. From this table, it determines where the note was assigned either an internal Voice, or a MIDI Channel.

If the note was assigned to an internal Voice, the gate is removed from the Voice and its Release Velocity updated. If the note originally came from the keyboard, then a MIDI Note Off message is generated to match the previously sent MIDI Note On message.

If the note was assigned to MIDI, then a matching Note Off message is generated on the appropriate MIDI Channel (the Basic Channel if the note was not spilled over, or the Basic Channel + 2 if it was spilled over).

MIDI Mono Mode

In Mono Mode, individual Voices are assigned directly to particular MIDI Channels. This effectively divides up the M-6 into multiple monophonic synthesizers, one per Voice. The Voices are assigned one-to-one with MIDI channels starting at the Basic Channel. Assuming all six Voices are available (if we are in Split Mode, there may be fewer), notes on the Basic Channel would be assigned to Voice 1, notes on the Basic Channel +1 would be assigned to Voice 2, etc. Thus, the unit is receiving on a "band" of Channels six wide. If the Basic Channel selection reaches the end (Channel 16), it wraps around and starts again on Channel 1. Even if the parameter M01 OMNI MODE is ON, the unit will only respond to messages on the appropriate Channel, as described above. Mode3: Omni On/Mono is not supported.

If, while in Mono Mode, a second Note On message is received on the Channel which a Voice is monitoring, and the Voice is already playing a note for which a Note Off has not yet been received, the new note will "rob" the Voice from the old note, and a multi (but not single) trigger will be generated. The Note Off message for the old note will not ungate the Voice. Only the Note Off message which matches the new note will remove the gate.

In Mono Mode, notes played on the keyboard are assigned to internal Voices in exactly the same way they are in Poly Mode. Mono Mode only affects the way that MIDI note messages are assigned to Voices, and the way in which MIDI Out messages for notes played on the keyboard are generated.

Note On / Key Press Handling

When a Note On message is received from MIDI, the Channel on which it was received determines exactly which Voice is to play it. If the Voice is ungated, then the note is assigned to it, the Voice is gated, and a single and multi trigger is generated. If the Voice is currently gated, then the old note is "robbed," the Voice is reassigned to the new note, and a multi (but not single) trigger is generated. Since the note originated with MIDI, no MIDI Out message is generated.

When a key is pressed on the keyboard, it is assigned to internal Voices as described above for Poly mode. If it was successfully assigned to an internal Voice (either through Voices being free or through Rob mode being ON), and we are in Mono/Omni Off, then a MIDI Note On message is generated for the note on the MIDI Channel to which that Voice is assigned (remember that in Mono Mode, every Voice has a Channel assigned to it). If we are in Mono/Omni On mode, or the note was not assigned to an internal Voice, then the Note On is sent on the Basic Channel.

M13 SPILLOVER has no effect in Mono Mode.

Note Off / Key Release Handling

When a MIDI Note Off message is received on a particular MIDI Channel, the gate is removed from the Voice and its Release Velocity and pitch are updated as appropriate. No MIDI out message is generated.

When a key is released, and the matching Note On was assigned to an internal Voice, that Voice becomes ungated. A MIDI Note Off message is generated on the Channel assigned to the Voice.

In Mono Mode, the M-6 supports multiple Note Off messages for a single Note On message; this allows multiple updates of the Release Velocity, as is often done by guitar controllers. The Note Off updates the pitch and Release Velocity of the note, and generates a multi trigger.

Unison Mode

The M-6 has a Unison Mode in which all of the internal Voices (or, if we are in Split mode, all of the Voices assigned to the side of the keyboard which is in Unison Mode) play each note played. Thus, the M-6 is effectively a monophonic synthesizer using six Voices and 12 oscillators. Unison Mode is controlled by parameter P48 KEYBOARD MODE. If this parameter is set to UNISON, all of the Voices will play any incoming note. The Unison Mode priority is strictly Low Note Priority: the lowest note received, on either MIDI or from the keyboard, will be assigned to the Voices, robbing the Voices from the previous note, if any. If a note is robbed, a multi trigger but not a single trigger will be generated.

When in Unison Mode, the unit effectively behaves as if it had only one internal Voice. Thus, Spillover (if parameter M13 SPILLOVER is ON) will occur after the first, not the sixth, note is assigned to internal Voices.

Note that while Unison Mode affects the way in which notes are assigned to Voices, it does not affect the way that incoming MIDI messages are recognized. Thus, if the unit (in addition to being in Unison Mode) is in MIDI Mono/Omni Off mode, note messages on the Basic Channel will be recognized, and assigned to Voices using Last Note Priority. In Mono/Omni On mode, the unit will assign the first MIDI note received on any Channel to all Voices, and will then continue to monitor that channel alone until a matching Note Off message is received, at which point it will revert to monitoring all Channels.

If, while in Mono Mode, a second Note On message is received on the Channel which the Voices are monitoring, and the Voices are already playing a note for which a Note Off has not yet been received, the new note will "rob" the Voices from the old note, and a multi (but not single) trigger will be generated. The Note Off message for the old note will not ungate the Voices; only the Note Off message which matches the new note will remove the gate.

Usually, Unison Mode is used with MIDI Poly mode, with either Omni Off or On.

Controllers

Controllers, from the point of view of the M-6, are internal "registers" which hold values put into them. Controllers can be set by one of two sources: a hardware device, such as the lever labelled "1" on the Matrix-6, or via a MIDI controller change message sent by some other device. Some controllers can only be set by MIDI. For example, the logical controller Lever 2 exists on the Matrix-6R, although it has no physical Lever 2. It is set by another device sending a MIDI controller change message to the appropriate controller number.

Local controllers are always summed with controller values received from MIDI.

With the exception of lever 1, all of the controllers in the M-6 have associated controller numbers which can be changed by the user and their defaults are given on the next page under the heading **Using Controllers.**

Local and Global Controllers

Controllers can be either **Local** or **Global**. A Global Controller is one which same value at all times for all of the Voices of the unit. A Local Controller is one where a separate value is maintained for each Voice. This distinction is only important when Controller values are changed via MIDI. When a Local Controller is changed by its corresponding hardware Controller, the value for all of the Voices is updated.

When a MIDI Controller change message is received on a particular MIDI Channel for a Local Controller, the values which are updated are those which belong to Voices which are receiving on that Channel. When a MIDI Controller change message is received for a Global Controller on any Channel which the unit is listening to, the value for the parameter is updated for all six Voices in the instrument.

Local Controllers are useful for guitar controllers, which can send separate Pitch Bend messages for each string on separate Channels. The M-6 can properly respond to such messages.

In the M-6, Lever 1, Pedal 1, and Pressure are Local. All other Controllers are Global.

Controller Parameters

The parameter M02 CONTROLLERS controls the handling of MIDI Controller change messages. When M02 CONTROLLERS is ON, Controller messages are generated whenever the hardware Controllers on the M-6 are changed, and Controller change messages via MIDI are recognized and used to update the internal Controller values. When M02 CONTROLLERS is OFF, Controller messages are not generated, and no Controller change message are recognized from MIDI.

Using Controllers

Controllers are used by the M-6 as modulation sources for its Voices. Controller information can be received over MIDI from specified Controller numbers, and changes to the M-6's own hardware Controllers can be sent over MIDI as Controller change messages.

The Controllers on the M-6 are:

Controller Name	Controller Number (default Value)	Comments
Pedal 1	7	Continuous, range 0 through 127
Pedal 2	64	Switch, either 0 or 127
Lever 1	BENDER	Cannot be reassigned; range 0 through 255 (8 bits)
Lever 2	1	Controls both lever 2 and 3
Lever 3	2	The "away" half of lever 2

These Controllers are available as modulation sources on both the Matrix-6 and Matrix-6R. On the Matrix-6, Pedal 1 and Pedal 2 correspond to the hardware footpedals and Lever 1 and Lever 2 correspond to the hardware levers. These Controllers produce numeric values as their output which are used to modulate the Voices in the M-6.

Pedal 1, in hardware, is a continuous controller with range 0 through 127. Pedal 2, in hardware, is a switch and is either open or closed. The M-6 automatically adapts to the type of footswitch (normally open or normally closed) when it is first turned on. The "normal" position corresponds to a numeric value of 0; the "active" position always corresponds to a numeric value of 127. Although Pedal 2 is a switch in hardware, and is always transmitted as either 0 or 127, it can be set to any of its 128 possible values by a Controller change command from MIDI.

Lever 1 (the Pitch Bender) produces a continuous range of values, with the lowest value (0) being generated when Lever 1 is pulled all the way towards the user, the median value (127) being generated when the lever is at the center position, and the highest value (255) being generated when the lever is pushed all the way away from the user. Lever 1 on both MIDI transmit and receive is permanently assigned to the Channel Pitch Bend function.

Lever 2 generates its lowest value (0) when at center position, and its largest value when it is pulled all the way towards the user. When it is pushed away from the user, the value of Lever 2 stays at 0, but the value of Lever 3 increases as the physical lever is pushed away. Thus, Lever 3 does not correspond to an actual hardware lever but to the "other half" of the throw of Lever 2.

On the Matrix-6R, all of the Levers and Pedals still exist as modulation sources, but only the Pedals exist in hardware; in order to use the Levers, Controller Change Messages must be received by the Matrix-6R from MIDI.

Pedal 1, Pedal 2, Lever 2 and Lever 3 can be reassigned to different MIDI Channels, although the same Channel is always used for both send and receive; Lever 1 is always assigned to MIDI Channel Pitch Bend on both send and receive.

MIDI Messages Sent
Assuming that parameter M02 CONTROLLERS is ON, a MIDI message for a Controller is sent whenever one of the hardware controllers is changed. The message is always sent on the Basic Channel of the unit, and includes the new Controller value (even in Split mode). The Controller Number can be set by the user, using Master parameters M06 – M09. The default values of these parameters are given in the table on the previous page. Lever 1 is always sent using MIDI Channel Pitch Bend message, with eight bits.

MIDI Messages Recognized
Assuming that parameter M02 CONTROLLERS is ON, MIDI Controller Change messages which apply to a Controller Number to which an internal M-6 Controller is assigned are recognized and updated. Controller change messages are recognized on any of the MIDI channels to which the M-6 is listening. If the Controller being changed is specific to particular Voices rather than Global to the entire instrument, then the Controller Change message will only update the Controller values for those Voices which are listening to the MIDI Channel on which it was received.

Split Mode

The M-6 supports a Split Mode, where the unit can play two Patches simultaneously. The Patches are assigned to the Upper and Lower areas of the keyboard (with a programmable Split Point dividing the keyboard in two). The Patches are called the "Upper" and "Lower" Patches.

When in Split, the unit's six Voices can be assigned as: 6/0 (all Lower, none Upper), 4/2, 2/4, or 0/6 (all Upper, none Lower). Thus, Split mode effectively divides the unit up into two independent synthesizers, each with a certain number of the M-6's six Voices.

When in Split mode, the unit has **two** Basic Channels: the Basic Channel for the overall unit (Channel N) is the Channel on which the Lower keyboard responds to MIDI commands, and the Upper keyboard responds on the next higher MIDI Channel, N + 1. MIDI Output generated by the Lower section goes out on Channel N, and that generated by the Upper area goes out on channel N + 1. Spillover works similarly: Spillover from the Lower goes out on Channel N + 2, and from the Upper on N + 1 + 2, or Channel N + 3.

Controller messages, both primary and secondary, are recognized and transmitted on both Channels. Local Controller messages affect only the Voices assigned to the keyboard on whose MIDI Channel the message was received. Global Controller changes affect all Voices.

If the Splits of the keyboard overlap, notes played in the overlapping section will be treated exactly as two independent notes: one on the Upper Channel, and one on the Lower.

If a keyboard half has no Voices assigned to it (a 6/0 or 0/6 Split), then that section of the keyboard is used only as a controller keyboard generating MIDI Out messages. All key messages from a keyboard section with no Voices transmit on the Channel for that section (N or N + 1). A keyboard area with no Voices **never** spills over.

Remember that Master Edit parameters affect **both** Patches in Split mode, since Master Edit parameters affect all Patches. This includes many important MIDI settings, such as M01 OMNI MODE, M18 MIDI MONO and M13 SPILLOVER.

System Exclusive

The M-6 uses System Exclusive messages to send Patches from one unit to another and to allow one M-6 to be the "front panel" for another when editing Patches and setting parameters. This section describes the System Exclusive message functions and formats.

General Format

All System Exclusive messages generated and recognized by the M-6 have the same general structure. This structure consists of three parts: a Lead-In, which starts and identifies the System Exclusive sequence, an Operation, which contains an opcode and data bytes, and an End of Exclusive status byte. There can only be one operation in the System Exclusive message.

There are two valid formats of the Lead-In sequence. Once is specific to the Matrix-6 and Matrix-6R, and one is a special message used for compatability with the Matrix-12 and Xpander synthesizers. They differ only in the device ID (06H for the M-6, 02H for the Matrix-12 and Xpander). Unless otherwise noted, the M-6 will recognize System Exclusive messages sent with either Lead-In, and will always generate the M-6 specific format on transmission. The format of an entire System Exclusive message is:

Byte	Function
F0H	System Exclusive byte
10H	Oberheim ID code
dd	Device ID, 06H for M-6 specific format, 02H for Matrix-12/Xpander
<opcode>	Opcodes are always in the range 0 through 127, inclusive.
<data bytes>	The number of data bytes is defined by the <opcode>. The data bytes are always in the range 0 through 7FH.
F7H	End of System Exclusive (EOX)

The M-6 always generates an EOX byte to end its System Exclusive transmissions, and will recognize any Status message except Real-time messages as ending a received System Exclusive message. Any System Exclusive message which contains a manufacturer ID other than 10H, or a device ID other than 06H or 02H, or an illegal opcode, is ignored. In addition, the M-6 will always wait 20 mSec after sending an EOX byte before sending any other data. Conversely, System Exclusive data sent to the M6 – particularly Patch dumps – should be separated by at least 20 mSec. The individual operations are described below.

Patch Transmission

The M-6 can both send and receive Patches, Split Patches, and Master Edit parameters via MIDI. Patch transmission can be triggered from the front panel or by a MIDI request for a Patch Dump. The M-6 can also be requested to send all of its Single Patches, Splits and Master parameters at once. The operations are:

Request Patch Dump

This message is used by an external device to request the M-6 to dump one or all of its Patches via MIDI. This is usually used in a "closed loop" MIDI configuration: the MIDI Out of the M-6 goes to the MIDI In of the other device, and the MIDI Out of the other device goes to the MIDI In of the M-6. The format of a Request Patch Dump operation is listed at the top of the following page.

Byte	Function
04H	Opcode
xx	Code indicating what to transmit: 0: Transmit all Single Patches, Splits, and Master parameters 1: Transmit a Single Patch 2: Transmit a Split 3: Transmit Master parameters
pp	Patch Number to transmit, 0 – 99 for Single Patches, 0 – 49 for Splits.

This byte is ignored for Transmit Master Parameters and Transmit All requests, but but must be included to pad out the fixed-length message.

When a DUMP ALL command is received (Code 0), the M-6 will dump all of its internal data as separate Patches, Splits and Master parameter blocks. **This means that each patch in the stream will have its own System Exclusive header and EOX command. If it is desired to transfer this data to a remote data storage device, the user should be required to tell the device when the transfer is done (> 1 second after the "10 SEND ALL" message reappears on the M-6's display) or the device should assume more data will be incoming until a timeout of > 500 mSec with no further incoming data has occurred.**

The total number of bytes transmitted in response to the dump all command is approximately 29K bytes including headers, checksums and EOX marks. It should be noted that all data (excluding headers, checksums and EOX marks) is transmitted nybble-wise, so judicious use of space could store all the transmitted data in as little as 15K bytes.

Note that for downward compatibility with version 1.xx, a Single Patch can also be requested by the sequence F0,10,06,00,pp,F7 where pp is the requested Single Patch Number in the range 0 to 99.

Single Patch Data
This message contains the actual Single Patch data. The opcode is followed by a stream of data bytes containing the Patch information. **When more than one Patch is being transmitted at a time (in a "Send All" operation), each Patch is sent as a separate System Exclusive message.** The form of a Single Patch dump operation is:

Byte	Function
01H	Opcode
pp	Single Patch Number from 0 through 99.
xx . . . yy	Patch or parameter data. Each byte is sent nybble-by-nybble, as follows: Each byte in the (eight bit) Patch data to be transmitted is sent as two bytes. The first sent byte, in its least significant four bits, contains the least significant four bits of the original byte. The second sent byte, also in its least significant four bits, contains the most significant four bits of the original byte. See Appendix A for the exact format of this data.
cc	Checksum. The original (**not** transmitted) data is summed in seven bits ignoring overflows, and the result is put here. If this checksum does not match that calculated while reading the sent Patch, an error has been determined and the Patch is ignored. When the M-6 receives a Patch data message via MIDI, it also checks to see that hardware protect is not on, and the Patch whose number is in the message is not protected. It then replaces the Patch in M-6 Patch storage with the Patch received.

Split Patch Data

This message contains the actual Split Patch data. The opcode is followed by a stream of data bytes containing the Patch information. The form of a Single Patch Dump operation is:

Byte	Function
02H	Opcode
pp	Split Patch Number from 0 through 49.
xx . . . yy	Patch data. Each byte is sent nybble-by-nybble, as above.
cc	Checksum.

Master Parameter Data

This message contains the actual Master parameter data. The opcode is followed by a stream of data bytes containing the parameter information. The form of a Master Parameter Dump operation is:

Byte	Function
03H	Opcode
xx . . . yy	Parameter data. Each byte is send nybble-by-nybble, as above.
cc	Checksum.

Remote Editing

The M-6 has a set of System Exclusive messages which can be used to edit Patch parameters via MIDI. This is an alternative to transmitting the entire Patch in its edited form. The primary differences are that this editing operation can be performed much more quickly than retransmitting the entire Patch, and any currently gated sounds will continue playing through the remote edit operation. This makes it possible to hear a sound change under remote control without regating the sound after each update. The remote editing System Exclusive messages are:

Select Quick Patch Edit

This operation selects the Quick mode of the Patch Edit function on the M-6. The M-6 must be in Patch Edit mode in order to act upon parameter change commands. This command should be used as a prefix to any remote editing commands. The select Quick Patch Edit operation has the format:

Byte	Function
05H	Opcode

Change Parameter

This operation changes the Value of the specified parameter. If the Value specified is out of range for the parameter, the operation is ignored. This operation implicitly selects the specified parameter as the current parameter, just as does the Select Parameter operation. The M-6 must be in QUICK Patch Edit mode to perform this operation.

Byte	Function
06H	Opcode
pp	Parameter Number to change; must be in range 0 through 99, and specify a parameter in the current Page.
vv	New parameter Value; must be within correct range for current parameter. This does not support negative Values.

Remote Mode Change
For compatibility with the Oberheim Xk Keyboard Controller, Matrix-12 and Xpander, the following codes are recognized, although they cannot be generated:

Select **Single Patch** Mode

Byte	Function
F0H	Start of System Exclusive
10H	Oberheim Mfg. ID
02H	Xpander Product Code
0DH	Switch Program Mode opcode
01H	Select Single Patch Mode
F7H	End of System Exclusive

Select **Split** Mode

Byte	Function
F0H	Start of System Exclusive
10H	Oberheim Mfg. ID
02H	Xpander Product Code
0DH	Switch Program Mode opcode
02H	Select "Multi-Patch" (Split) Mode
F7H	End of System Exclusive

Parameters
For any System Exclusive messages to be generated or recognized, the parameter M04 SYSTEM EXCLUSIVE must be ON. The Master Edit page parameter M10 SEND DATA is used to send a Single Patch, Split Patch, or the Master Edit parameter set via MIDI to another device. If the M-6 is in Single Patch mode, M10 SEND DATA sends the currently selected Single Patch. If the M-6 is in Split Patch mode, M10 SEND DATA sends the currently selected Split Patch. The parameter M11 SEND ALL sends all of the M-6's 100 Single Patches, 50 Split Patches, and the set of Master Edit parameters to another device. A SEND ALL operation takes about 12 seconds to complete.

Miscellaneous MIDI Functions

Patch Changes
The M-6 can both send and receive MIDI Patch Change messages. If parameter M03 PATCH CHANGES is ON, a MIDI Patch Change message will be sent any time the current Patch or current Split is changed, either from the M-6's front panel or via MIDI. The current Patch or Split will also be changed any time a Patch Change message is received via MIDI when M03 PATCH CHANGES is ON. The M-6 ignores commands to change to a Patch Number greater than 99 when in Single mode, and greater than 49 in Split mode.

Patch Change messages, both on receive and transmit, are affected by the parameters M15 PATCH MAP and M16 P MAP EDIT. See the next section for details.

Patch Mapping
The M-6 can translate incoming and outgoing Patch Change messages such that a particular incoming message can select a different Patch on the M-6, and that selecting a particular Patch on the M-6 can generate a Patch Change message different from the one selected on the sending unit. The parameters which control this feature are M15 PATCH MAP and M16 P MAP EDIT. When M15 PATCH MAP is ON, this feature enables the translating incoming and outgoing Patches through the Patch Map. When off, all incoming and outgoing Patches are acted on and transmitted exactly as they appear. M16 P MAP EDIT is used to change the values of the Patch Map, and is described in more detail in the Matrix-6R Owner's Manual, 1st Edition.

Echo

If parameter 12 MIDI ECHO is ON, all MIDI data received in the M-6's MIDI In port is immediately retransmitted out the MIDI Out port. MIDI data is retransmitted regardless of what Channel it is being sent on. All of the M-6's internally generated MIDI messages are also transmitted out the MIDI Out port, thus allowing the M-6 to serve as a "MIDI mixer," combining its own data with that from the unit connected to the MIDI In port.

System Exclusive data is not retransmitted out the MIDI Out port, whether or not the System Exclusive data is recognized by the M-6 itself.

Since MIDI Note messages which are received by the M-6 on the M-6's Basic Channel are never transmitted out MIDI Out by the Voice assignment logic, there is no possibility of multiple Note Ons or Offs being generated on the M-6's Basic Channel from a single Note On or Off message received on the MIDI In port in Echo mode. Only the message generated by the Echo will be sent out the MIDI Out port. If the unit is in Spillover mode, however, there is the possibility that a Note On or Off command will be both echoed out on the Basic Channel and transmitted out the MIDI Out port on the Basic Channel + 2 as part of the Spillover operation. Furthermore, as all MIDI data regardless of Channel is echoed, it is possible that the Spillover messages generated on the Basic Channel + 2 might conflict with other messages if the unit or units plugged into the MIDI In port are also transmitting on the Basic Channnel + 2 of the M-6.

To avoid these problems, simply reserve the "band" of four Channels starting with the M-6's Basic Channel to the M-6 and any units set to receive spilled-over notes from it. If Split mode is not going to be used, the band need only be three Channels wide.

Running Status

The M-6 correctly interprets Running ("Implied") Status on all incoming messages, and generates Running Status on all output Channel Voice Messages when possible.

Note Attack/Release Velocity

Since the M-6 supports both Attack and Release Velocity on all notes, the M-6 always transmits a Note Off as a separate Note Off message (8bH 0kkkkkkk 0vvvvvvv) rather than as a Note On with zero Velocity (9bH 0kkkkkkk 0). Both forms are correctly handled as MIDI In messages.

Local Control

Parameter M05 LOCAL CONTROL corresponds to the MIDI Local Control message. If the parameter is ON, then the keyboard on the Matrix-6 sends notes to its own internal Voices as well as out MIDI Out. If the parameter is OFF, then the keyboard only generates MIDI Out notes and does not assign any notes to internal Voices. The internal Voices are controllable only via MIDI. If local control is OFF, then Spillover is disabled as well, regardless of the setting of M13 SPILLOVER.

Parameter M05 LOCAL CONTROL is always set to ON upon power-up and reset.

Active Sensing

The M-6 supports MIDI Active Sensing, both on transmission and reception. If the parameter M14 ACT SENSE is ON, the M-6 does the following: If 240 milliseconds passes with no activity on the M-6's MIDI Out port, the M-6 generates an Active Sense message (status FEH). Once an Active Sense message is received, if 360 milliseconds passes with no activity on the M-6's MIDI In port, the M-6 performs an All Notes Off operation. Any received message on any port counts as "activity" for the purposes of active sensing. If parameter M14 ACT SENSE is OFF, then no Active Sensing messages are generated by the M-6, and the M-6 does not turn off notes until a matching Note On command, or an explicit All Notes Off command, is received.

Tune Request

The M-6 responds to incoming Tune Request messages by tuning the high frequency oscillators (HFOs). It will transmit a Tune Request message whenever the tune command is given via the second column button in MASTER mode. Note that the Tune Request message is not associated with the M-6's CALIBRATE function.

Appendix A: MIDI Implementation Summary

Channel Voice Messages

Status	Data Bytes	Description
1000 xxxx	0nnn nnnn 0vvv vvvv	Note Off (Release Velocity = 1 – 127)
1001 xxxx	0nnn nnnn 0vvv vvvv	Note On (Velocity = 1 – 127, 0 = Note Off: Receive Only)
1011 xxxx	0ccc cccc 0nnn nnnn	Controller Change (If Enabled) (Any controller 0 – 121 can be used; all values are 7 bits only)
1100 xxxx	0nnn nnnn	Program Select (If Enabled) (Range = 0 – 99)
1110 xxxx	0000 000n 0nnn nnnn	Pitch Bend (LSB) Pitch Bend (MSB)

Channel Mode Messages

1011 xxxx	7AH 0ccccccc	Local Control 0 = Off, 127 = On
1011 xxxx	7BH 00H	All Notes Off
1011 xxxx	7CH 00H	OMNI Mode Off
1011 xxxx	7DH 00H	OMNI Mode On (OMNI is assumed off in MONO ON)
1011 xxxx	7EH 06H	Mono Mode On
1011 xxxx	7FH 00H	Mono Mode Off

System Common Messages

1111 0110		Tune Request
1111 0000	10H <dd> <opcode> <data bytes> 0F7H	System Exclusive – Oberheim ID Device ID End of System Exclusive
1111 0111		End of System Exclusive

System Real Time Messages

1111 1110	Active Sensing

Appendix B: System Exclusive Data Format

Device ID = 06H : Matrix-6

Opcode	Byte	Description
00H		Single Patch Request (Receive Only)
	0ppppppp	Patch Number (0 – 99)
01H		Single Patch Data
	0ppppppp	Patch Number (0 – 99)
	<patch data>	See Apendix C
	0ccccccc	Checksum
02H		Split Patch Data
	00pppppp	Split Number (0 – 49)
	<split data>	See Appendix D
	0ccccccc	Checksum
03H		Master Parameter Data
	<parm data>	See Appendix E
	0ccccccc	Checksum
04H		General Data Request (Receive Only)
	000000xx	Code For Request Type 0 = Transmit all Single patches, Splits and Master parameters 1 = Transmit a Single Patch 2 = Transmit a Split Patch 3 = Transmit Master Parameters
	0ppppppp	Patch Number to Transmit 0 – 99 for Single Patches 0 – 49 for Split Patches 0 for Master parameters
05H		Enter Remote Edit Mode (Receive Only)
06H		Change Parameter (Receive Only)
	0ppppppp	Parameter to change
	0vvvvvvv	New Value

Device ID = 02H : Xpander (Compatible with Matrix-12, Xk)

Opcode	Byte	Description
0DH		Mode Change (Receive Only)
02H		Multi Patch Mode (Split Mode)
0DH		Mode Change (Receive Only)
01H		Single Patch Mode

Appendix C: Single Patch Data Format

Statistics: 134 Bytes/Single Patch
= 268 nybbles transmitted + 5 bytes Header + 1 byte Checksum + 1 byte EOX
= 275 total transmitted bytes/Single Patch

Byte	Parameter	# Bits	Description
0 - 7		6 each	Patch Name Each character is represented by the lower 6 bits of it's ASCII representation.
8	48	2	Keyboard Mode 0 = Reassign 1 = Rotate 2 = Unison 3 = Reassign w/ Rob
9	00	6	DCO 1 Initial Frequency LSB = 1 Semitone
10	05	6	DCO 1 Initial Waveshape 0 = Sawtooth 31 = Triangle
11	03	6	DCO 1 Initial Pulse Width
12	07	2	DCO 1 Fixed Modulations Bit 0 = Lever 1 Bit 1 = Vibrato
13	06	2	DCO 1 Waveform Enable Bit 0 = Pulse Bit 1 = Wave
14	10	6	DCO 2 Initial Frequency LSB = 1 Semitone
15	15	6	DCO 2 Initial Waveshape 0 = Sawtooth 31 = Triangle
16	13	6	DCO 2 Initial Pulse Width
17	17	2	DCO 2 Fixed Modulations Bit 0 = Lever 1 Bit 1 = Vibrato
18	16	3	DCO 1 Waveform Enable Bit 0 = Pulse Bit 1 = Wave Bit 2 = Noise
19	12	6 (signed)	DCO 2 Detune

Byte	Parameter	# Bits	Description
20	20	6	Mix
21	08	2	DCO 1 Fixed Modulations Bit 0 = Portamento Bit 1 = Not used
22	09	1	DCO 1 Click
23	18	2	DCO 2 Fixed Modulations Bit 0 = Portamento Bit 1 = Keyboard Tracking Enable
24	19	1	DCO 2 Click
25	02	2	DCO Sync Mode
26	21	7	VCF Initial Frequency LSB = 1 Semitone
27	24	6	VCF Initial Resonance
28	25	2	VCF Fixed Modulations Bit 0 = Lever 1 Bit 1 = Vibrato
29	26	2	VCF Keyboard Modulation Bit 0 = Portamento Bit 1 = Keyboard
30	30	6	VCF FM Initial Amount
31	27	6	VCA 1 (Exponential) Initial Amount
32	44	6	Portamento Initial Rate
33	46	2	Lag Mode 0 = Constant Speed 1 = Constant Time 2 = Exponential 3 = Exponential
34	47	1	Legato Portamento Enable
35	80	6	LFO 1 Initial Speed
36	86	2	LFO 1 Trigger 0 = No Trigger 1 = Single Trigger 2 = Multi Trigger 3 = External Trigger
37	87	1	LFO 1 Lag Enable
38	82	3	LFO 1 Waveshape (see Table 1)
39	83	5	LFO 1 Retrigger point
40	88	5	LFO 1 Sampled Source Number
41	84	6	LFO 1 Initial Amplitude

Byte	Parameter	# Bits	Description
42	90	6	LFO 2 Initial Speed
43	96	2	LFO 2 Trigger See LFO 1 Triggers above
44	97	1	LFO 2 Lag Enable
45	92	3	LFO 2 Waveshape (see Table 1)
46	93	5	LFO 2 Retrigger point
47	98	5	LFO 2 Sampled Source Number
48	94	6	LFO 2 Initial Amplitude
49	57	3	Env 1 Trigger Mode Bit 0 = Reset Bit 1 = Multi Trigger Bit 2 = External Trigger
50	50	6	Env 1 Initial Delay Time
51	51	6	Env 1 Initial Attack Time
52	52	6	Env 1 Initial Decay Time
53	53	6	Env 1 Sustain Level
54	54	6	Env 1 Initial Release Time
55	55	6	Env 1 Initial Amplitude
56	59	2	Env 1 LFO Trigger Mode Bit 0 = Gated Bit 1 = LFO Trigger
57	58	2	Env 1 Mode Bit 0 = DADR Mode Bit 1 = Free Run
58	67	3	Env 2 Trigger Mode See Env 1 Trigger Mode above
59	60	6	Env 2 Initial Delay Time
60	61	6	Env 2 Initial Attack Time
61	62	6	Env 2 Initial Decay Time
62	63	6	Env 2 Sustain Level
63	64	6	Env 2 Initial Release Time
64	65	6	Env 2 Initial Amplitude
65	69	2	Env 2 LFO Trigger Mode See Env 1 LFO Trigger Mode above
66	68	2	Env 2 Mode See Env 1 Mode above
67	77	3	Env 3 Trigger Mode See Env 1 Trigger Mode above
68	70	6	Env 3 Initial Delay Time
69	71	6	Env 3 Initial Attack Time
70	72	6	Env 3 Initial Decay Time
71	73	6	Env 3 Sustain Level
72	74	6	Env 3 Initial Release Time
73	75	6	Env 3 Initial Amplitude

Byte	Parameter	# Bits	Description
74	79	2	Env 3 LFO Trigger Mode (See Env 1 LFO Trigger Mode)
75	78	2	Env 3 Mode (See Env 1 Mode)
76	33	5	Tracking Generator Input Source Code (See Table 2)
77	34	6	Tracking Point 1
78	35	6	Tracking Point 2
79	36	6	Tracking Point 3
80	37	6	Tracking Point 4
81	38	6	Tracking Point 5
82	40	6	Ramp 1 Rate
83	41	2	Ramp 1 Mode 0 = Single Trigger 1 = Multi Trigger 2 = External Trigger 3 = External Gated
84	42	6	Ramp 2 Rate
85	43	2	Ramp 2 Mode (See Ramp 1 Mode)
86	01	7 (Signed)	DCO 1 Freq. by LFO 1 Amount
87	04	7 (Signed)	DCO 1 PW by LFO 2 Amount
88	11	7 (Signed)	DCO 2 Freq. by LFO 1 Amount
89	14	7 (Signed)	DCO 2 PW by LFO 2 Amount
90	22	7 (Signed)	VCF Freq. by Env 1 Amount
91	23	7 (Signed)	VCF Freq. by Pressure Amount
92	28	7 (Signed)	VCA 1 by Velocity Amount
93	29	7 (Signed)	VCA 2 by Env 2 Amount
94	56	7 (Signed)	Env 1 Amplitude by Velocity Amount
95	66	7 (Signed)	Env 2 Amplitude by Velocity Amount
96	76	7 (Signed)	Env 3 Amplitude by Velocity Amount
97	85	7 (Signed)	LFO 1 Amp. by Ramp 1 Amount
98	95	7 (Signed)	LFO 2 Amp. by Ramp 2 Amount
99	45	7 (Signed)	Portamento Rate by Velocity Amount
100	31	7 (Signed)	VCF FM Amount by Env 3 Amount
101	32	7 (Signed)	VCF FM Amount by Pressure Amount
102	81	7 (Signed)	LFO 1 Speed by Pressure Amount
103	91	7 (Signed)	LFO 2 Speed by Keyboard Amount
104		5	Matrix Modulation Bus 0 Source Code (see Table 2)
105		7 (Signed)	MM Bus 0 Amount
106		5	MM Bus 0 Destination code (see Table 3)
107		5	Matrix Modulation Bus 1 Source Code (see Table 2)
108		7 (Signed)	MM Bus 1 Amount
109		5	MM Bus 1 Destination code (see Table 3)
110		5	Matrix Modulation Bus 2 Source Code (see Table 2)
111		7 (Signed)	MM Bus 2 Amount
112		5	MM Bus 2 Destination code (see Table 3)

Byte	Parameter	# Bits	Description
113		5	Matrix Modulation Bus 3 Source Code (see Table 2)
114		7 (Signed)	MM Bus 3 Amount
115		5	MM Bus 3 Destination code (see Table 3)
116		5	Matrix Modulation Bus 4 Source Code (see Table 2)
117		7 (Signed)	MM Bus 4 Amount
118		5	MM Bus 4 Destination code (see Table 3)
119		5	Matrix Modulation Bus 5 Source Code (see Table 2)
120		7 (Signed)	MM Bus 5 Amount
121		5	MM Bus 5 Destination code (see Table 3)
122		5	Matrix Modulation Bus 6 Source Code (see Table 2)
123		7 (Signed)	MM Bus 6 Amount
124		5	MM Bus 6 Destination code (see Table 3)
125		5	Matrix Modulation Bus 7 Source Code (see Table 2)
126		7 (Signed)	MM Bus 7 Amount
127		5	MM Bus 7 Destination code (see Table 3)
128		5	Matrix Modulation Bus 8 Source Code (see Table 2)
129		7 (Signed)	MM Bus 8 Amount
130		5	MM Bus 8 Destination code (see Table 3)
131		5	Matrix Modulation Bus 9 Source Code (see Table 2)
132		7 (Signed)	MM Bus 9 Amount
133		5	MM Bus 9 Destination code (see Table 3)

Table 1

LFO Wave Codes

0 = Triangle
1 = Up Sawtooth
2 = Down Sawtooth
3 = Square
4 = Random
5 = Noise
6 = Sampled Modulation
7 = Not Used

Table 2

Modulation Source Codes
Tracking Generator Inputs

Unused Modulations must have their Sources and Destinations set to 0.

0 = Unused Modulation*
1 = Envelope 1
2 = Envelope 2
3 = Envelope 3
4 = LFO 1
5 = LFO 2
6 = Vibrato
7 = Ramp 1
8 = Ramp 2
9 = Keyboard
10 = Portamento
11 = Tracking Generator
12 = Keyboard Gate
13 = Velocity
14 = Release Velocity
15 = Pressure
16 = Pedal 1
17 = Pedal 2
18 = Lever 1
19 = Lever 2
20 = Lever 3

* The "0 = Unused Modulation" parameter in this Table is found in the Modulation Source list only. This parameter is not a Tracking Generator input and thus will not be displayed in parameter S33 TRACK INPUT.

Table 3

Modulation Destination Codes

Unused Modulations must have their Sources and Destinations set to 0.

0 = Unused Modulation
1 = DCO 1 Frequency
2 = DCO 1 Pulse Width
3 = DCO 1 Waveshape
4 = DCO 2 Frequency
5 = DCO 2 Pulse Width
6 = DCO 2 Waveshape
7 = Mix Level
8 = VCF FM Amount
9 = VCF Frequency
10 = VCF Resonance
11 = VCA 1 Level
12 = VCA 2 Level
13 = Envelope 1 Delay
13 = Envelope 1 Attack
15 = Envelope 1 Decay
16 = Envelope 1 Release
17 = Envelope 1 Amplitude
18 = Envelope 2 Delay
19 = Envelope 2 Attack
20 = Envelope 2 Decay
21 = Envelope 2 Release
22 = Envelope 2 Amplitude
23 = Envelope 3 Delay
24 = Envelope 3 Attack
25 = Envelope 3 Decay
26 = Envelope 3 Release
27 = Envelope 3 Amplitude
28 = LFO 1 Speed
29 = LFO 1 Amplitude
30 = LFO 2 Speed
31 = LFO 2 Amplitude
32 = Portamento Time

Appendix D: Split Patch Data Format

Statistics: 18 Bytes/Split Patch
= 36 nybbles transmitted + 5 bytes header + 1 byte Checksum + 1 byte EOX
= 42 total transmitted bytes/Split Patch

Byte	Parameter	# Bits	Description
0 - 5		6 each	Split Name Each character is represented by the lower 6 bits of it's ASCII representation.
6		6	Not Used
7		6	Not Used
8		7	Lower Patch Number
9		7	Upper Patch Number
10	0	7	Left Zone Limit
11	1	6 (Signed)	Left Zone Transpose
12	2	1	Left Zone MIDI Out Enable
13	3	7	Right Zone Limit
14	4	6 (Signed)	Right Zone Transpose
15	5	1	Right Zone MIDI Out Enable
16	6	6 (Signed)	Left/Right Balance -31 = Left only +31 = Right only
17	7	2	Voice Configuration 0 = 2/4 1 = 4/2 2 = 6/0 3 = 0/6

Appendix E: Global Parameters Data Format

Statistics: 236 bytes/Global Parameters
= 472 nybbles transmitted + 4 bytes Header + 1 byte Checksum + 1 byte EOX
= 477 total transmitted bytes/Global parameters

Byte	Parameter	# Bits	Description
0			Not Used
1	30	6	Vibrato Speed
2	31	3	Vibrato Waveform
3	32	6	Vibrato Amplitude
4	33	2	Vibrato Speed Mod Source Code
5	34	6	Vibrato Speed Modulation Amount
6	35	2	Vibrato Amp Mod Source Code
7	36	6	Vibrato Amp Modulation Amount
8	55	6 (Signed)	Master Tune
9	40	2	Velocity Scale Type
10	41	6	Velocity Sensitivity
11	00	4	MIDI Basic Channel
12	01	1	MIDI OMNI Mode Enable
13	02	1	MIDI Controllers Enable
14	03	1	MIDI Patch Changes Enable
15	04	1	MIDI SysX Enable
16	05	1	MIDI Local Control Enable
17	06	7	MIDI Pedal 1 Controller
18	07	7	MIDI Pedal 2 Controller
19	08	7	MIDI Lever 2 Controller
20	09	7	MIDI Lever 3 Controller
21	42	1	Pedal 2 Invert Enable
22	43	1	Levers Invert Enable
23	53	5	Display Brightness
24	56	1	SQUICK Enable
25	17	1	Patch Map Echo Enable
26	57	1	Stereo Output Enable
27			Not Used
28	44	6	Pressure Standoff
29	13	1	Spillover Enable
30			Not Used
31	14	1	MIDI Active Sensing Enable
32	12	1	MIDI Echo Enable
33	15	1	Patch Map Enable
34			Not Used
35	18	1	MIDI Mono Mode Enable
36-135		6 each	Input Patch Map
136-235		6 each	Output Patch Map