Roland

MT-32 MIDI Implementation

Roland Exclusive Messages

1. Data Format for Exclusive Messages

Roland's MIDI implementation uses the following data format for all exclusive messages (type IV) :

Byte	Description
F0H	Exclusive status
41H	Manufactures ID (Roland)
DEV	Device ID
MDL	Model ID
CMD	Command ID
[BODY]	Maindata
F7H	End of exclusive

 # MIDI status : F0H, F7H

An exclusive message must be flanked by a pair of status codes, starting with a Manufactures-ID immediately after F0H (MIDI version1.0).

 # Manufactures-ID : 41H

The Manufactures-ID identifies the manufacturer of a MIDI instrument that triggeres an exclusive message. Value 41H represents Roland's Manufactures-ID.

 # Device-ID : DEV

The Device-ID contains a unique value that identifies the individual device in the multiple implementation of MIDI instruments. It is usually set to 00H - 0FH, a value smaller by one than that of a basic channel, but value 00H - 1FH may be used for a device with multiple basic channels.

 # Model-ID : MDL

The Model-ID contains a value that uniquely identifies one model from another. Different models, however, may share an identical Model-ID if they handle similar data.

The Model-ID format may contain 00H in one or more places to provide an extended data field. The following are examples of valid Model-IDs, each representing a unique model :

01H
02H
03H
00H, 01H
00H, 02H
00H, 00H, 01H

 # Command-ID : CMD

The Command-ID indicates the function of an exclusive message. The Command-ID format may contain 00H in one or more places to provide an extended data field. The following are examples of valid Command-IDs, each representing a unique function :

01H
02H
03H
00H, 01H
00H, 02H
00H, 00H, 01H

 # Main data : BODY

This field contains a message to be exchanged across an interface. The exact data size and contents will vary with the Model-ID and Command-ID.

2. Address-mapped Data Transfer

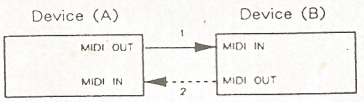
Address mapping is a technique for transferring messages conforming to the data format given in Section 1. It assigns a series of memory-resident records--waveform and tone data, switch status, and parameters, for example--to specific locations in a machine-dependent address space, thereby allowing access to data residing at the address a message specifies.

Address-mapped data transfer is therefore independent of models and data categories. This technique allows use of two different transfer procedures : one-way transfer and handshake transfer.

 # One-way transfer procedure (See Section3 for details.)

This procedure is suited for the transfer of a small amount of data. It sends out an exclusive message completely independent of a receiving device status.

Connection Diagram

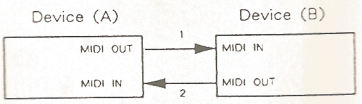


Connectionat point2 is essential for "Request data" procedures. (See Section3.)

 # Handshake-transfer procedure (See Section4 for details.)

This procedure initiates a predetermined transfer sequence (handshaking) across the interface before data transfer takes place. Handshaking ensures that reliability and transfer speed are high enough to handle a large amount of data.

Connection Diagram



Connectionat points1 and 2 is essential.

*There are separate Command-IDs for different transfer procedures.
*DevicesA and B cannot exchange data unless they use the same transfer procedure, share identical Device-ID and Model ID, andare ready for communication.

3. One-way Transfer Procedure

This procedure sends out data all the way until it stops when the messages are so short that answerbacks need not be checked.
For long messages, however, the receiving device must acquire each message in time with the transfer sequence, which inserts intervals of at least 20milliseconds in between.

Types of Messages

Message	Command ID
Request data 1	RQ1 (11H)
Data set 1	DT1 (12H)

 # Request data # 1 : RQ1 (11H)

This message is sent out when there is a need to acquire data from a device at the other end of the interface. It contains data for the address and size that specify designation and length, respectively, of data required.
On receiving an RQ1 message, the remote device checks its memory for the data address and size that satisfy the request.

If it finds them and is ready for communication, the device will transmit a "Data set#1 (DT1)" message, which contains the requested data. Otherwise, the device will send out nothing.

Byte	Description
F0H	Exclusive status
41H	Manufactures ID (Roland)
DEV	Device ID
MDL	Model ID
11H	Command ID
aaH	Address MSB
⋮	⋮
	LSB
ssH	Size MSB
⋮	⋮
	LSB
sum	Check sum
F7H	End of exclusive

*The size of the requested data does not indicate the number of bytes that will make up a DT1 message, but represents the address fields where the requested data resides.
*Some models and data are subject to limitations in data format used for a single transaction. Requested data, for example, may have a limit in length or must be divided into predetermined address fields before it is exchanged across the interface.
*The same number of bytes comprises address and size data, which, however, vary with the Model-ID.
*The error checking process uses a checksum that provides a bit pattern where lower seven bits are zero when values for an address, size, and that checksum are summed.

Data set # 1 : DT1 (12H)

This message corresponds to the actual data transfer process. Because every byte in the data is assigned a unique address, a DT1 message can convey the starting address (es) of one or more data as well as a series of data formatted in an address- dependent order.

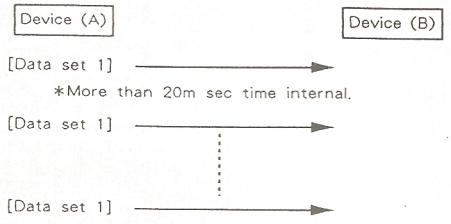
Although the MIDI standards inhibit non-real time messages from interrupting an exclusive one, some devices support a "soft- through" mechanism for such interrupts. To maintaincompatibility with such devices, Roland has limited the DT1 to 256bytes so that an excessively long message is sent out in separate segments.

Byte	Description
F0H	Exclusive
41H	Manufactures ID (Roland)
DEV	Device ID
MDL	Model ID
12H	Command ID
aaH	Address MSB
⋮	⋮
	LSB
ddH	Data
⋮	⋮
sum	Check sum
F7H	End of exclusive

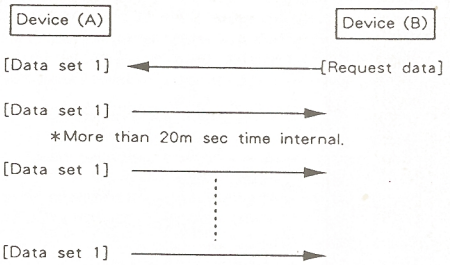
*A DT1 message is capable of providing only the valid data among those specified by an RQ1 message.
*Some models and data are subject to limitations in data format used for a single transaction. Requested data, for example, may have a limit in length or must be divided into predetermined address fields before it is exchanged across the interface.
*The number of bytes comprising address data varies from one Model-ID to another.
*The error checking process uses a checksum that provides a bit pattern where lower seven bits are zero when values for an address, size, and that checksum are summed.

Example of Message Transactions

● Device A sending data to Device B
Transfer of a DT1 message is all that takes place.



● Device B requesting data from Device A
Device B sends an RQ1 message to Device A. Checking the message, Device A sends a DT1 message back to Device B.



4. Handshake- Transfer Procedure

Handshaking is an interactive process where two devices exchange error checking signals before a message transaction takes place, thereby increasing data reliability. Unlike one-way transfer that inserts a pause between message transactions, handshake transfer allows much speedier transactions becausedata transfer starts once the receiving device returns a ready signal.

When it comes to handling large amounts of data--sampler waveforms and synthesizer tones over the entire range, for example--across a MIDI interface, handshaking transfer is morecefficient than one-way transfer.

Types of Messages

Message	Command ID
Want to send data	WSD (40H)
Request data	RQD (41H)
Data set	DAT (42H)
Acknowledge	ACK (43H)
End of data	EOD (45H)
Communication error	ERR (4EH)
Rejection	RJC (4FH)

Want to send data : WSD (40H)

This message is sent out when data must be sent to a device at the other end of the interface. It contains data for the address and size that specify designation and length, respectively, of the data to be sent.

On receiving a WSD message, the remote device checks its memory for the specified data address and size which will satisfy the request. If it finds them and is ready for communication, the device will return an "Acknowledge (ACK)" message.
Otherwise, it will return a "Rejection (RJC)" message.

Byte	Description
F0H	Exclusive status
41H	Manufactures ID (Roland)
DEV	Device ID
MDL	Model ID
40H	Command ID
aaH	Address MSB
⋮	⋮
	LSB
ssH	Size MSB
⋮	⋮
	LSB
sum	Check sum
F7H	End of exclusive

*The size of the data to be sent does not indicate the number of bytes that make up a "Data set (DAT)" message, but represents the address fields where the data should reside.
*Some models and data are subject to limitations in data format used for a single transaction. Requested data, for example, may have a limit in length or must be divided into predetermined address fields before it is exchanged across the interface.
*The same number of bytes comprises address and size data, which, however, vary with the Model-ID.
*The error checking process uses a checksum that provides a bit pattern where lower seven bits are zero when values for an address, size, and that checksum are summed.

Request data : RQD (41H)

This message is sent out when there is a need to acquire data from a device at the other end of the interface. It contains data for the address and size that specify designation and length, respectively, of data required.

On receiving an RQD message, the remote device checks its memory for the data address and size which satisfy the request. If it finds them and is ready for communication, the device will transmit a "Data set (DAT)" message, which contains the requested data. Otherwise, it will return a "Rejection (RJC)" message.

Byte	Description
F0H	Exclusive status
41H	Manufactures ID (Roland)
DEV	Device ID
MDL	Model ID
41H	Command ID
aaH	Address MSB
⋮	⋮
	LSB
ssH	Size MSB
⋮	⋮
	LSB
sum	Check sum
F7H	End of exclusive

*The size of the requested data does not indicate the number of bytes that make up a "Data set (DAT)" message, but represents the address fields where the requested data resides.
*Some models and data are subject to limitations in data format used for a single transaction. Requested data, for example, may have a limit in length or must be divided into predetermined address fields before it is exchanged across the interface.
*The same number of bytes comprises address and size data, which, however, vary with the Model-ID.
*The error checking process uses a checksum that provides a bit pattern where lower seven bits are zero when values for an address, size, and that checksum are summed.

Data set : DAT (42H)

This message corresponds to the actual data transfer process. Because every byte in the data is assigned a unique address, the message can convey the starting address (es) of one or more data as well as a series of data formatted in an address-dependent order.

Although the MIDI standards inhibit non-real time messages from interrupting an exclusive one, some devices support a "soft- through" mechanism for such interrupts. To maintaincompatibility with such devices, Roland has limited the DAT to 256bytes so that an excessively long message is sent out in separate segments.

Byte	Description
F0H	Exclusive status
41H	Manufactures ID (Roland)
DEV	Device ID
MDL	Model ID
42H	Command ID
aaH	Address MSB
⋮	⋮
	LSB
ddH	Data
⋮	⋮
sum	Check sum
F7H	End of exclusive

*A DAT message is capable of providing only the valid data among those specified by an RQD or WSD message.
*Some models and data are subject to limitations in data format used for a single transaction. Requested data, for example, may have a limit in length or must be divided into predetermined address fields before it is exchanged across the interface.
*The number of bytes comprising address data varies from one model ID to another.
*The error checking process uses a checksum that provides a bit pattern where lower seven bits are zero when values for an address, size, and that checksum are summed.

Acknowledge : ACK (43H)

This message is sent out when no error was detected on reception of a WSD, DAT, "End of data (EOD)", or some other message and a requested setup or action is complete. Unless it receives an ACK message, the device at the other end will not proceed to the next operation.

Byte	Description
F0H	Exclusive status
41H	Manufactures ID (Roland)
DEV	Device ID
MDL	Model ID
43H	Command ID
F7H	End of exclusive

End of data : EOD (45H)

This message is sent out to inform a remote device of the end of a message. Communication, however, will not come to an end unless the remote device returns an ACK message even though an EOD message was transmitted.

Byte	Description
F0H	Exclusive status
41H	Manufactures ID (Roland)
DEV	Device ID
MDL	Model ID
45H	Command ID
F7H	End of exclusive

Communications error : ERR (4EH)

This message warns the remote device of a communications fault encountered during message transmission due, for example, to a checksum error. An ERR message may be replaced with a "Rejection (RJC)" one, which terminates the current message transaction in midstream.

When it receives an ERR message, the sending device may either attempt to send out the last message a second time or terminate communication by sending out an RJC message.

Byte	Description
F0H	Exclusive status
41H	Manufactures ID (Roland)
DEV	Device ID
MDL	Model ID
4EH	Command ID
F7H	End of exclusive

Rejection : RJC (4FH)

This message is sent out when there is a need to terminate communication by overriding the current message. An RJC message will be triggered when :

a WSD or RQD message has specified an illegal data address or size, or the device is not ready for communication.

an illegal number of addresses or data has been detected.

data transfer has been terminated by an operator.

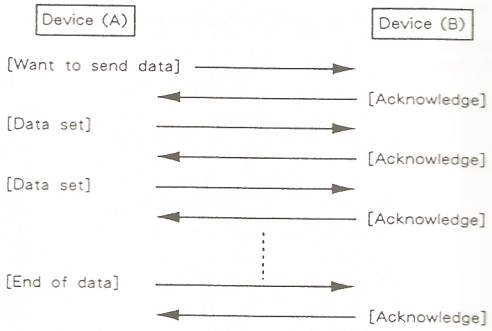
a communications error has occurred.

An ERR message may be sent out by a device on either side of the interface. Communication must be terminated immediately when either side triggers an ERR message.

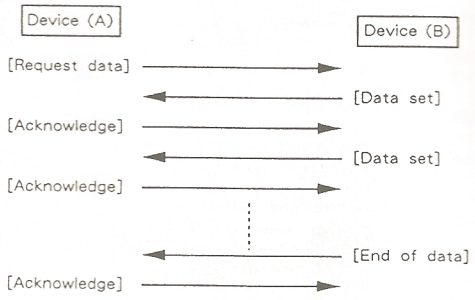
Byte	Description
F0H	Exclusive status
41H	Manufactures ID (Roland)
DEV	Device ID
MDL	Model ID
4FH	Command ID
F7H	End of exclusive

Example of Message Transactions

● Data transfer from device (A) to device (B).

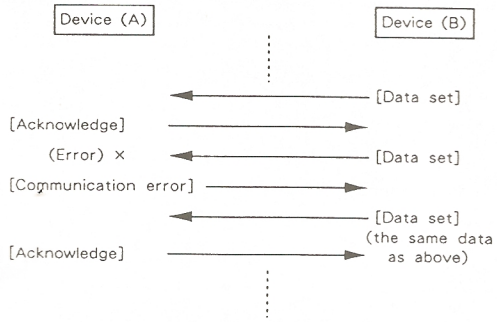


● Device (A) requests and receives data from device (B).

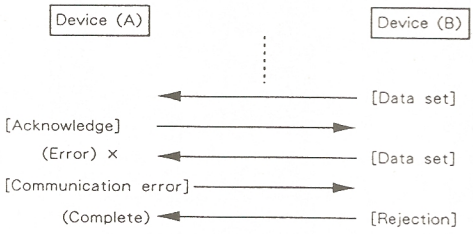


● Error occurs while device (A) is receiving data from device (B).

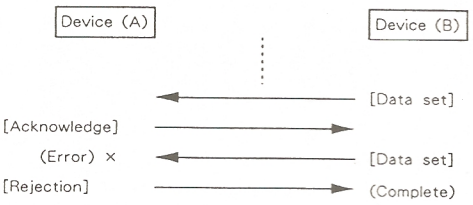
1) Data transfer from device (A) to device (B).



2) Device (B) rejects the data re-transmitied, and completes data transfer.



3) Device (A) immediately completes data transfer.



MULTI TIMBRE SOUND MODULE
MODEL MT−32

MIDI Impremention

Date : Sep. 19. 1987
Version : 1.01

1. TRANSMITTED DATA

Bypassed message

In Overflow Assign mode, the following MIDI In messages are sent to MIDI Outas

- Channel Voice messages except Note On
- System Exclusive message whose manufacturer ID# is 41H
- Odd Note On (s) left unassigned any voice because all assignable voices are engaged.

Created message

System exclusive

Status
F0H : System Exclusive
F7H : EOX (End of System Exclusive)

See "3.EXCLUSIVE COMMUNICATIONS" for details.

2. RECOGNIZED DATA

Note event

Note off

Status	Second	Third
8nH	kkH	vvH
9nH	kkH	00H

kkH : Note number 0CH−6CH (12−108)
vvH : ignored

Note on

Statu	Second	Third
9nH	kkH	vvH

kkH : Note number 0CH−6CH (12−108)
vvH : Velocity 1H−7FH (1−127)

Control change

Continuous controller (14 bits)

Status	Second	Third
BnH	mmH	vvH
Modulation	mmH=01H	vvH=0H−7FH (0−127)
Volume	mmH=07H	vvH=0H−7FH (0−127)
Panpot	mmH=0AH	vvH=0H−7FH (0−127)
Expression	mmH=0BH	vvH=0H−7FH (0−127)

Continuous controller (7 bits)

Status	Second	Third
BnH	mmH	vvH
Hold 1	mmH=40H	vvH=0H−3FH (0−63) OFF 40H−7FH (64−127) ON
Resets all controllers	mmH=79h	vvH=0

Program change

Status	Second
CnH	ppH

ppH : Program number 0H−7FH (0−127)

Program Change changes Patch.

Pitch bender

Status	Second	Third
EnH	llH	mmH

llH : 0H−7FH (0−127)
mmH : 0H−7FH (0−127)

Channel mode message

Status	Second	Third
BnH	mmH	00H

mmH : All Notes Off 7BH (123)
Omni Off 7CH (124)
Omni On 7DH (124)
Mono On 7EH (124)
Poly On 7FH (128)

Recognized as only All Notes Off.
MT−32 does not change mode, but remains in mode 3 (Omni off, Poly).

Active sensing

Status
FEH

System exclusive

Status
F0H : System Exclusive
F7H : EOX (End of System Exclusive)

3. EXCLUSIVE COMMUNICATIONS

Model−ID# of MT−32 is 16H.

MT−32 can receive/send some of the EXCLUSIVE MESSAGEs in the D−50 (Roland synthesizer) format.

Model−ID# of D−50 is 14H.

Device−ID# is the basic channel# of the each part or Unit# of the MT−32

Unit# can be changed in "UNIT# SETUP MODE".
Device ID numbers, 0−31, are displayed on the LCD as 1−32, respectively.

One way communication

Request RQ1 11H

When the RQ1 received contains a start address listed in Parameter base address, and address size is 1 or more, MT−32 sends the corresponding data.

In Overflow Assign mode, MT−32 does not recoginze RQ1, but passes the message to MIDI OUT.

MT−32 won't transmit RQ1 in the default mode.

Byte	Description	
F0H	Exclusive status	
41H	Roland−ID	
DEV	Device−ID	
16H (14H)	Model−ID (MT−32 (D−50))	*3−1
11H	Command−ID (RQ1)	
aaH	Address MSB	*3−2
aaH	Address	
aaH	Address LSB	
ssH	Size MSB	
ssH	Size	
ssH	Size LSB	
sum	Checksum	
F7H	EOX (End of Exclusive)	

Data set DT1 12H

When the DT1 contains a start address as defined in RQ1 above, MT−32 stores the data into that memory location.

MT−32 sends this message upon receiving RQ1 in the default mode.

Additional function in Overflow Assign mode :
MT−32 retransmits DT1 while it processes the DT1 data as necessary.

Byte	Description	
F0H	Exclusive status	
41H	Roland−ID	
DEV	Device−ID	
16H (14H)	Model−ID (MT−32 (D−50))	*3−1
12H	Command−ID (DT1)	
aaH	Address MSB	*3−2
aaH	Address	
aaH	Address LSB	
ddH	Data	*3−3
:		
sum	Checksum	
F7H	EOX (End of Exclusive)	

Handshaking communication

Want to send data WSD 40H

Upon receiving WSD, MT−32 sends ACK and waits for DATA SET message. However, if any part is reproducing sound, MT−32 sends RJC.

In Overflow Assign mode, MT−32 relays this message to downstream.

In the default mode, MT−32 won't send this message.

Byte	Description	
F0H	Exclusive status	
41H	Roland－ID	
DEV	Device－ID	
16H	Model－ID (MT－32)	
40H	Command－ID (WSD)	
aaH	Address MSB	*3－2
aaH	Address	
aaH	Address LSB	
ssH	Size MSB	
ssH	Size	
ssH	Size LSB	
sum	Checksum	
F7H	EOX (End of Exclusive)	

Request data　　RQD 41H

When the RQD contains a start address as defined in RQ1 above, MT－32 stores the data into that memory location. However, if any part is reproducing sound, MT－32 sends RJC.

In Overflow Assign mode, MT－32 relays this message to downstream without recognizing it.

In the default mode, MT－32 won't send this message.

Byte	Description	
F0H	Exclusive status	
41H	Roland－ID	
DEV	Device－ID	
16H	Model－ID (MT－32)	
41H	Command－ID (RQD)	
aaH	Address MSB	*3－2
aaH	Address	
aaH	Address LSB	
ssH	Size MSB	
ssH	Size	
ssH	Size LSB	
sum	Checksum	
F7H	EOX (End of Exclusive)	

Data set　　DAT 42H

When the DAT contains a start address as defined in RQ1 above, MT－32 stores the data into that memory location. However,if any part is reproducing sound, MT－32 sends RJC.

In the default mode, MT－32 sends this data upon receipt of RQD.

In Overflow Assign mode, MT－32 relays this message to downstream without recognizing it.

Byte	Description	
F0H	Exclusive status	
41H	Roland－ID	
DEV	Device－ID	
16H	Model－ID (MT－32)	
42H	Command－ID (DAT)	
aaH	Address MSB	*3－2
aaH	Address	
aaH	Address LSB	
ddH	Data	*3－3
:		
sum	Checksum	
F7H	EOX (End of Exclusive)	

Acknowledge　　ACK 43H

When MT－32 receives this message after sending DAT, it sends the next data. When MT－32 receives this message after sending EOD, it ends the current handshaking.

MT－32 sends ACK when it receives WSD, RQD or DAT in the default mode with no part reproducing sound and with data checksum proves correct.

Byte	Description
F0H	Exclusive status
41H	Roland－ID
DEV	Device－ID
16H	Model－ID (MT－32)
43H	Command－ID (ACK)
F7H	EOX (End of Exclusive)

End of data　　EOD 45H

Upon receiving this message, it sends ACK and ends the current handshaking.

After finishing the data set (DAT) transmission in the default mode, MT－32 sends this message.

In Overflow Assign mode, MT－32 relays this message to downstream without recognizing the contents.

Byte	Description
F0H	Exclusive status
41H	Roland－ID
DEV	Device－ID
16H	Model－ID (MT－32)
45H	Command－ID (EOD)
F7H	EOX (End of Exclusive)

Communication error　　ERR 4EH

If checksum doesn't agree (failure in data reception), MT－32 sends this message.

When MT－32 receives this message in the default mode, it sends the latest message again.

In Overflow Assign mode, MT－32 relays this message to downstream without recognizing it.

Byte	Description
F0H	Exclusive status
41H	Roland－ID
DEV	Device－ID
16H	Model－ID (MT－32)
4EH	Command－ID (ERR)
F7H	EOX (End of Exclusive)

Rejection　　RJC 4FH

If MT－32 receives WSD while it is reproducing sound, it sends RJC.

When MT－32 receives this message, it ends the current handshaking.

In Overflow Assign mode, MT－32 relays this message to downstream without recognizing it.

Byte	Description
F0H	Exclusive status
41H	Roland－ID
DEV	Device－ID
16H	Model－ID (MT－32)
4FH	Command－ID (RJC)
F7H	EOX (End of Exclusive)

Notes :

*3－1 Both model－IDs are supported. Addresses & parameters are described in section 4 for model－ID 16H (MT－32) and in section 5 for model－ID 14H (D－50, PG－1000).

*3－2 Address & Size should be the address where data exist.

*3－3 If the data is Partial Reserve Parameter, received data must comprise all the parameters for being recognized.

4. Address mapping of parameters

Addresses are shown in Hexa－decimal, while numbers are given in 7 bits.

Address	MSB		LSB
binary	0aaa aaaa	0bbb bbbb	0ccc cccc
7 bit Hex	AA	BB	CC

The actual address of a parameter in a block is the sum of the start address of each block and one or more offset address. That is, parameters marked by *4－1, *4－2 have two offset addresses : one in the table under NOTE *4－1, *4－2 and the other in Rhythm Setup table, in Common parameter table or in Partial parameter table.

■ Parameter base address

Temporary area (Accessible on each basic channel)

Start address	Description	
00 00 00	Patch Temp Area (part 1－8)	
01 00 00	Setup Temp Area (rhythm part)	*4－1
02 00 00	Timbre Temp Area (part 1－8)	*4－2

Whole part (Accessible on UNIT#)

Start address	Description	
03 00 00	Patch Temp Area (part 1)	
03 00 10	Patch Temp Area (part 2)	
:		
03 00 60	Patch Temp Area (part 7)	
03 00 70	Patch Temp Area (part 8)	
03 01 10	Setup Temp Area (rhythm part)	
04 00 00	Timbre Temp Area (part 1)	*4－2
04 01 76	Timbre Temp Area (part 2)	*4－2
:		
04 0b 44	Timbre Temp Area (part 7)	*4－2
04 0d 3a	Timbre Temp Area (part 8)	*4－2
05 00 00	Patch Memory #1	
05 00 08	Patch Memory #2	
:		
05 07 70	Patch Memory #127	
05 07 78	Patch Memory #128	
08 00 00	Timbre Memory #1	*4－2
08 02 00	Timbre Memory #2	*4－2
:		
08 7C 00	Timbre Memory #63	*4－2
08 7E 00	Timbre Memory #64	*4－2
10 00 00	System area	
20 00 00	Display	*4－3
7F xx xx	All parameter reset	*4－4

Common parameter *4-5

Offset address	Description		
00H	0aaa aaaa	TONE NAME 1	32-127 (ASCII)
:			
09H	0aaa aaaa	TONE NAME 10	
0AH	0000 aaaa	Structure of Partial# 1&2	0-12 (1-13)
0BH	0000 aaaa	Structure of Partial# 3&4	0-12 (1-13)
0CH	0000 aaaa	PARTIAL MUTE	0-15 (0000-1111)
0DH	0000 000a	ENV MODE	0-1 (Normal,No sustain)
Total size		00 00 0EH	

Partial parameter *4-5

Offset address	Description		
00 00H	0aaa aaaa	WG PITCH COARSE	0-96 (C1,C#1,-C9)
00 01H	0aaa aaaa	WG PITCH FINE	0-100 (-50-+50)
00 02H	0000 aaaa	WG PITCH KEYFOLLOW	0-16 (-1,-1/2, -1/4,0,1/8, 1/4,3/8,1/2, 5/8,3/4,7/8, 1,5/4,3/2,2,s1, s2)
00 3H	0000 000a	WG PITCH BENDER SW	0-1 (OFF,ON)
00 04H	0000 000a	WG WAVEFORM	0-1 (SQU,SAW)
00 05H	0aaa aaaa	WG PCM WAVE #	0-127 (1-128)
00 06H	0aaa aaaa	WG PULSE WIDTH	0-100
00 07H	0000 aaaa	WG PW VELO SENS	0-14 (-7-+7)
00 08H	0000 aaaa	P-ENV DEPTH	0-10
00 09H	0aaa aaaa	P-ENV VELO SENS	0-100
00 0AH	0000 0aaa	P-ENV TIME KEYF	0-4
00 0BH	0aaa aaaa	P-ENV TIME 1	0-100
00 0CH	0aaa aaaa	P-ENV TIME 2	0-100
00 0DH	0aaa aaaa	P-ENV TIME 3	0-100
00 0EH	0aaa aaaa	P-ENV TIME 4	0-100
00 0FH	0aaa aaaa	P-ENV LEVEL 0	0-100 (-50-+50)
00 10H	0aaa aaaa	P-ENV LEVEL 1	0-100 (-50-+50)
00 11H	0aaa aaaa	P-ENV LEVEL 2	0-100 (-50-+50)
00 12H	0aaa aaaa	P-ENV SUSTAIN LEVEL	0-100 (-50-+50)
00 13H	0aaa aaaa	END LEVEL	0-100 (-50-+50)
00 14H	0aaa aaaa	P-LFO RATE	0-100
00 15H	0aaa aaaa	P-LFO DEPTH	0-100
00 16H	0aaa aaaa	P-LFO MOD SENS	0-100
00 17H	0aaa aaaa	TVF CUTOFF FREQ	0-100
00 18H	000a aaaa	TVF RESONANCE	0-30
00 19H	0000 aaaa	TVF KEYFOLLOW	0-14 (-1,-1/2, -1/4,0,1/8, 1/4,3/8,1/2, 5/8,3/4,7/8, 1,5/4,3/2,2)
00 1AH	0aaa aaaa	TVF BIAS POINT/DIR	0-127 (<1A-<7C >1A->7C)
00 1BH	0000 aaaa	TVF BIAS LEVEL	0-14 (-7-+7)
00 1CH	0aaa aaaa	TVF ENV DEPTH	0-100
00 1DH	0aaa aaaa	TVF ENV VELO SENS	0-100
00 1EH	0000 0aaa	TVF ENV DEPTH KEYF	0-4
00 1FH	0000 0aaa	TVF ENV TIME KEYF	0-4
00 20H	0aaa aaaa	TVF ENV TIME 1	0-100
00 21H	0aaa aaaa	TVF ENV TIME 2	0-100
00 22H	0aaa aaaa	TVF ENV TIME 3	0-100
00 23H	0aaa aaaa	TVF ENV TIME 4	0-100
00 24H	0aaa aaaa	TVF ENV TIME 5	0-100
00 25H	0aaa aaaa	TVF ENV LEVEL 1	0-100
00 26H	0aaa aaaa	TVF ENV LEVEL 2	0-100
00 27H	0aaa aaaa	TVF ENV LEVEL 3	0-100
00 28H	0aaa aaaa	TVF ENV SUSTAIN LEVEL	0-100
00 29H	0aaa aaaa	TVA LEVEL	0-100
00 2AH	0aaa aaaa	TVA VELO SENS	0-100
00 2BH	0aaa aaaa	TVA BIAS POINT 1	0-127 (<1A-<7C >1A->7C)
00 2CH	0000 aaaa	TVA BIAS LEVEL 1	0-12 (-12-0)
00 2DH	0aaa aaaa	TVA BIAS POINT 2	0-127 (<1A-<7C >1A->7C)
00 2EH	0000 aaaa	TVA BIAS LEVEL 2	0-12 (-12-0)
00 2FH	0000 0aaa	TVA ENV TIME KEYF	0-4
00 30H	0000 0aaa	TVA ENV TIME V_FOLLOW	0-4
00 31H	0aaa aaaa	TVA ENV TIME 1	0-100
00 32H	0aaa aaaa	TVA ENV TIME 2	0-100
00 33H	0aaa aaaa	TVA ENV TIME 3	0-100
00 34H	0aaa aaaa	TVA ENV TIME 4	0-100
00 35H	0aaa aaaa	TVA ENV TIME 5	0-100
00 36H	0aaa aaaa	TVA ENV LEVEL 1	0-100
00 37H	0aaa aaaa	TVA ENV LEVEL 2	0-100
00 38H	0aaa aaaa	TVA ENV LEVEL 3	0-100
00 39H	0aaa aaaa	TVA ENV SUSTAIN LEVEL	0-100
Total size		00 00 3AH	

System area

Offset address	Description		
00 00H	0aaa aaaa	MASTER TUNE	0-127 (432.1Hz-457.6Hz)
00 01H	0000 00aa	REVERB MODE	0-3 (Room,Hall,Plate,Tap9 delay)
00 02H	0000 0aaa	REVERB TIME	0-7 (1-8)
00 03H	0000 0aaa	REVERB LEVEL	0-7
00 04H	00aa aaaa	PARTIAL RESERVE (Part 1)	0-32 *4-6
00 05H	00aa aaaa	PARTIAL RESERVE (Part 2)	0-32 *4-6
00 06H	00aa aaaa	PARTIAL RESERVE (Part 3)	0-32 *4-6
00 07H	00aa aaaa	PARTIAL RESERVE (Part 4)	0-32 *4-6
00 08H	00aa aaaa	PARTIAL RESERVE (Part 5)	0-32 *4-6
00 09H	00aa aaaa	PARTIAL RESERVE (Part 6)	0-32 *4-6
00 0AH	00aa aaaa	PARTIAL RESERVE (Part 7)	0-32 *4-6
00 0BH	00aa aaaa	PARTIAL RESERVE (Part 8)	0-32 *4-6
00 0CH	00aa aaaa	PARTIAL RESERVE (Part R)	0-32 *4-6
00 0DH	000a aaaa	MIDI CHANNEL (Part 1)	0-16 (1-16,OFF)
00 0EH	000a aaaa	MIDI CHANNEL (Part 2)	0-16 (1-16,OFF)
00 0FH	000a aaaa	MIDI CHANNEL (Part 3)	0-16 (1-16,OFF)
00 10H	000a aaaa	MIDI CHANNEL (Part 4)	0-16 (1-16,OFF)
00 11H	000a aaaa	MIDI CHANNEL (Part 5)	0-16 (1-16,OFF)
00 12H	000a aaaa	MIDI CHANNEL (Part 6)	0-16 (1-16,OFF)
00 13H	000a aaaa	MIDI CHANNEL (Part 7)	0-16 (1-16,OFF)
00 14H	000a aaaa	MIDI CHANNEL (Part 8)	0-16 (1-16,OFF)
00 15H	000a aaaa	MIDI CHANNEL (Part R)	0-16 (1-16,OFF)
00 16H	0aaa aaaa	MASTER VOLUME	0-100
Total size		00 00 17H	

Rhythm setup

Offset address	Description		
00 00H	0aaa aaaa	TIMBRE	0-94 (M1-M64,R1-R30,OFF)
00 01H	0aaa aaaa	OUTPUT LEVEL	0-100
00 02H	0000 aaaa	PANPOT	0-14 (R-L)
00 03H	0000 000a	REVERB SWITCH	0-1 (OFF,ON)
Total size		00 00 04H	

Patch temp

Offset address	Description		
00 00H	0000 00aa	TIMBRE GROUP	0-3 (GROUP A,GROUP B,MEMORY,RHYTHM)
00 01H	00aa aaaa	TIMBRE NUMBER	0-63 (1-64)
00 02H	00aa aaaa	KEY SHIFT	0-48 (-24-+24)
00 03H	0aaa aaaa	FINE TUNE	0-100 (-50-+50)
00 04H	000a aaaa	BENDER RANGE	0-24
00 05H	0000 00aa	ASSIGN MODE	0-3 (POLY 1,POLY 2,POLY 3,POLY 4)
00 06H	0000 000a	REVERB SWITCH	0-1 (OFF,ON)
00 07H	0xxx xxxx	dummy	
00 08H	0aaa aaaa	OUTPUT LEVEL	0-100
00 09H	0000 aaaa	PANPOT	0-14 (R-L)
00 0AH	0xxx xxxx	dummy	
:			
00 0FH	0xxx xxxx		
Total size		00 00 10H	

■ Patch memory

Offset address	Description		
00 00H	0000 00aa	TIMBRE GROUP (GROUP A,GROUP B,MEMORY,RHYTHM)	0−3
00 01H	00aa aaaa	TIMBRE NUMBER	0−63
00 02H	00aa aaaa	KEY SHIFT	0−48 (−24−+24)
00 03H	0aaa aaaa	FINE TUNE	0−100 (−50−+50)
00 04H	000a aaaa	BENDER RANGE	0−24
00 05H	0000 00aa	ASSIGN MODE (POLY 1,POLY 2,POLY 3,POLY 4)	0−3
00 06H	0000 000a	REVERB SWITCH	0−1 (OFF,ON)
00 07H	0xxx xxxx	dummy	
Total size		00 00 08H	

■ DISPLAY

Offset address	Description		
00H	0aaa aaaa	DISPLAYED LETTER	32−127 (ASCII)
:			
13H	0aaa aaaa	DISPLAYED LETTER	
Total size		00 00 14H	

Notes :

*4−1

Structure of "Setup Temp" area is as follows.

Offset address	Description
00 00 00	Rhythm Setup (for Key# 24)
00 00 04	Rhythm Setup (for Key# 25)
00 00 08	Rhythm Setup (for Key# 26)
00 00 0C	Rhythm Setup (for Key# 27)
00 00 10	Rhythm Setup (for Key# 28)
:	:
:	:
:	:
00 01 78	Rhythm Setup (for Key# 86)
00 01 7C	Rhythm Setup (for Key# 87)

*4−2

Structure of "Timbre Temp/Memory" area is as follows.

Sub start address	Description
00 00 00	Common parameter
00 00 0E	Partial parameter (for Partial# 1)
00 00 48	Partial parameter (for Partial# 2)
00 01 02	Partial parameter (for Partial# 3)
00 01 3C	Partial parameter (for Partial# 4)

*4−3

The data sent to this address are recognized as the string of letters in ASCII CODE, and displayed on MT−32 LCD.
Cannot be called on RQ1 or RQD.

*4−4

All parameters will be initialized by sending data to this address.
Cannot be called on RQ1 or RQD.

*4−5

This parameter can be modified from D−50 (PG−1000) and results in accessing the address "02−00−00 (Timbre Temp Area (part))" of MT−32

*4−6

Partial Reserves should be simultaneously assigned to all the 9 parts by one Exclusive message without the total number of the Partial Reserves exceeding 32.

5. ADDRESS MAPPING OF PARAMETERS

〈compatible with D−50 (PG−1000)〉

■ Parameter base address

Start address	Description	
00−00−00	Partial 3	(0−53)
00−00−40	Partial 4	(64−117)
00−01−0A	Upper Common	(138−175)
00−01−40	Partial 1	(192−245)
00−02−00	Partial 2	(256−309)
00−02−4A	Lower Common	(330−367)

■ Partial parameters

Offset address	Description		
00 00H	0aaa aaaa	WG PITCH COARSE	0−72 (C1,C#1,−C7)
00 01H	0aaa aaaa	WG PITCH FINE	0−100 (−50−+50)
00 02H	0000 aaaa	WG PITCH KEYFOLLOW	0−16 (−1,−1/2, −1/4,0,1/8, 1/4,3/8,1/2, 5/8,3/4,7/8, 1,5/4,3/2,2,s1, s2)
00 03H	0xxx xxxx	dummy	
00 04H	0xxx xxxx	dummy	
00 05H	0000 000a	WG PITCH BENDER SW	0−1 (OFF,ON)
00 06H	0000 000a	WG WAVEFORM	0−1 (SQU,SAW)
00 07H	0aaa aaaa	WG PCM WAVE #	0−99 (1−100)
00 08H	0aaa aaaa	WG PULSE WIDTH	0−100
00 09H	0000 aaaa	WG PW VELO SENS	0−14 (−7−+7)
00 0AH	0xxx xxxx	dummy	
00 0BH	0xxx xxxx	dummy	
00 0CH	0xxx xxxx	dummy	
00 0DH	0aaa aaaa	TVF CUTOFF FREQ	0−100
00 0EH	000a aaaa	TVF RESONANCE	0−30
00 0FH	0000 aaaa	TVF KEYFOLLOW	0−14 (−1,−1/2, −1/4,0,1/8, 1/4,3/8,1/2, 5/8,3/4,7/8, 1,5/4,3/2,2)
00 10H	0aaa aaaa	TVF BIAS POINT/DIR	0−127 (<1A−<7C >1A−>7C)
00 11H	0000 aaaa	TVF BIAS LEVEL	0−14 (−7−+7)
00 12H	0aaa aaaa	TVF ENV DEPTH	0−100
00 13H	0aaa aaaa	TVF ENV VELO SENS	0−100
00 14H	0000 0aaa	TVF ENV DEPTH KEYF	0−4
00 15H	0000 0aaa	TVF ENV TIME KEYF	0−4
00 16H	0aaa aaaa	TVF ENV TIME 1	0−100
00 17H	0aaa aaaa	TVF ENV TIME 2	0−100
00 18H	0aaa aaaa	TVF ENV TIME 3	0−100
00 19H	0aaa aaaa	TVF ENV TIME 4	0−100
00 1AH	0aaa aaaa	TVF ENV TIME 5	0−100
00 1BH	0aaa aaaa	TVF ENV LEVEL 1	0−100
00 1CH	0aaa aaaa	TVF ENV LEVEL 2	0−100
00 1DH	0aaa aaaa	TVF ENV LEVEL 3	0−100
00 1EH	0aaa aaaa	TVF ENV SUSTAIN LEVEL	0−100
00 1FH	0xxx xxxx	dummy	
:			
00 22H	0xxx xxxx	dummy	
00 23H	0aaa aaaa	TVA LEVEL	0−100
00 24H	0aaa aaaa	TVA VELO SENS	0−100
00 25H	0aaa aaaa	TVA BIAS POINT 1	0−127 (<1A−<7C >1A−>7C)
00 26H	0000 aaaa	TVA BIAS LEVEL 1	0−12 (−12−0)
00 27H	0aaa aaaa	TVA ENV TIME 1	0−100
00 28H	0aaa aaaa	TVA ENV TIME 2	0−100
00 29H	0aaa aaaa	TVA ENV TIME 3	0−100
00 2AH	0aaa aaaa	TVA ENV TIME 4	0−100
00 2BH	0aaa aaaa	TVA ENV TIME 5	0−100
00 2CH	0aaa aaaa	TVA ENV LEVEL 1	0−100
00 2DH	0aaa aaaa	TVA ENV LEVEL 2	0−100
00 2EH	0aaa aaaa	TVA ENV LEVEL 3	0−100
00 2FH	0aaa aaaa	TVA ENV SUSTAIN LEVEL	0−100
00 30H	0xxx xxxx	dummy	
00 31H	0000 0aaa	TVA ENV TIME V_FOLLOW	0−4
00 32H	0000 0aaa	TVA ENV TIME KEYF	0−4
00 33H	0xxx xxxx	dummy	
00 34H	0xxx xxxx	dummy	
00 35H	0xxx xxxx	dummy	
Total size		00 00 36H	

■ Lower common parameter

Offset address	Description		
00 00H	0000 aaaa	Structure of Partial# 1&2	0–12 (1–13)
00 01H	0aaa aaaa	P–ENV VELO SENS (Partial#1)	0–100
00 02H	0000 0aaa	P–ENV TIME KEYF (Partial#1)	0–4
00 03H	0aaa aaaa	P–ENV TIME 1 (Partial#1)	0–100
00 04H	0aaa aaaa	P–ENV TIME 2 (Partial#1)	0–100
00 05H	0aaa aaaa	P–ENV TIME 3 (Partial#1)	0–100
00 06H	0aaa aaaa	P–ENV TIME 4 (Partial#1)	0–100
00 07H	0aaa aaaa	P–ENV LEVEL 0 (Partial#1)	0–100 (–50–+50)
00 08H	0aaa aaaa	P–ENV LEVEL 1 (Partial#1)	0–100 (–50–+50)
00 09H	0aaa aaaa	P–ENV LEVEL 2 (Partial#1)	0–100 (–50–+50)
00 0AH	0aaa aaaa	P–ENV SUS LEVEL (Partial#1)	0–100 (–50–+50)
00 0BH	0aaa aaaa	END LEVEL (Partial#1)	0–100 (–50–+50)
00 0CH	0xxx xxxx	dummy	
00 0DH	0aaa aaaa	P–LFO MOD SENS (Partial#1)	0–100
00 0EH	0aaa aaaa	P–LFO MOD SENS (Partial#2)	0–100
00 0FH	0xxx xxxx	dummy	
00 10H	0aaa aaaa	P–LFO RATE (Partial#1)	0–100
00 11H	0aaa aaaa	P–LFO DEPTH (Partial#1)	0–100
00 12H	0xxx xxxx	dummy	
00 13H	0xxx xxxx	dummy	
00 14H	0aaa aaaa	P–LFO RATE (Partial#2)	0–100
00 15H	0aaa aaaa	P–LFO DEPTH (Partial#2)	0–100
00 16H	0xxx xxxx	dummy	
:			
00 23H	0xxx xxxx	dummy	
00 24H	0000 00aa	PARTIAL MUTE (Partial# 1&2)	0–3 (00–11)
00 25H	0xxx xxxx	dummy	
Total size	00 00 26H		

■ Upper common parameter

Offset address	Description		
00 00H	0000 aaaa	Structure of Partial# 1&2	0–12 (1–13)
00 01H	0aaa aaaa	P–ENV VELO SENS (Partial#3)	0–100
00 02H	0000 0aaa	P–ENV TIME KEYF (Partial#3)	0–4
00 03H	0aaa aaaa	P–ENV TIME 1 (Partial#3)	0–100
00 04H	0aaa aaaa	P–ENV TIME 2 (Partial#3)	0–100
00 05H	0aaa aaaa	P–ENV TIME 3 (Partial#3)	0–100
00 06H	0aaa aaaa	P–ENV TIME 4 (Partial#3)	0–100
00 07H	0aaa aaaa	P–ENV LEVEL 0 (Partial#3)	0–100 (–50–+50)
00 08H	0aaa aaaa	P–ENV LEVEL 1 (Partial#3)	0–100 (–50–+50)
00 09H	0aaa aaaa	P–ENV LEVEL 2 (Partial#3)	0–100 (–50–+50)
00 0AH	0aaa aaaa	P–ENV SUS LEVEL (Partial#3)	0–100 (–50–+50)
00 0BH	0aaa aaaa	END LEVEL (Partial#3)	0–100 (–50–+50)
00 0CH	0xxx xxxx	dummy	
00 0DH	0aaa aaaa	P–LFO MOD SENS (Partial#3)	0–100
00 0EH	0aaa aaaa	P–LFO MOD SENS (Partial#4)	0–100
00 0FH	0xxx xxxx	dummy	
00 10H	0aaa aaaa	P LFO RATE (Partial#3)	0–100
00 11H	0aaa aaaa	P–LFO DEPTH (Partial#3)	0–100
00 12H	0xxx xxxx	dummy	
00 13H	0xxx xxxx	dummy	
00 14H	0aaa aaaa	P–LFO RATE (Partial#4)	0–100
00 15H	0aaa aaaa	P–LFO DEPTH (Partial#4)	0–100
00 16H	0xxx xxxx	dummy	
:			
00 23H	0xxx xxxx	dummy	
00 24H	0000 00aa	PARTIAL MUTE (Partial# 3&4)	0–3 (00–11)
00 25H	0xxx xxxx	dummy	
Total size		00 00 26H	

MULTI TIMBRE SOUND MODULE
MODEL MT−32

MIDI Implementation Chart

Date : Apr. 30. 1987
Version : 1.00

Function...		Transmitted	Recognized	Remarks
Basic Channel	Default		2−10	
	Changed		1−8, 10	
Mode	Default		Mode 3	
	Messages			
	Altered	*********		
Note Number		* 0−127	0−127	
	True Voice	*********	12−108	
Velocity	Note ON	*	○ v=1−127	
	Note OFF	*	×	
After Touch	Key's	*	×	
	Ch's	*	×	
Pitch Bender		*	○ 0−24 semi	
Control Change	1	*	○	Modulation
	7	*	○	Part Volume
	10	*	○	Panpot
	11	*	○	Expression
	12			
	:	*	×	
	63			
	64	*	○	Hold1
	65			
	:	*	×	
	120			
	121	*	○	Reset all controllers
Prog Change		*	○ 0−127	
	True #		0−127	
System Exclusive		○ *	○	
System Common	Song Pos	×	×	
	Song sel	×	×	
	Tune	×	×	
System Real Time	Clock	×	×	
	Commands	×	×	
Aux Message	Local ON/OFF	×	×	
	All Notes OFF	×	○ (123−127)	
	Active Sense	×	○	
	Reset	×	×	
Notes		* in OVERFLOW MODE received message goes thru MIDI OUT.		

Mode 1 : OMNI ON, POLY
Mode 2 : OMNI ON, MONO
Mode 3 : OMNI OFF, POLY
Mode 4 : OMNI OFF, MONO

○ : Yes
× : No

Roland

'87-12-A4-5I