Common Rules for MIDI-CI Profiles

MIDI Association Document: M2-102-U

Document Version 1.1
Draft Date May 11, 2023

Published June 15, 2023

Developed and Published By
The MIDI Association
and
Association of Musical Electronics Industry (AMEI)

PREFACE

MIDI Association Document M2-102-U
Common Rules for MIDI-CI Profiles

The MIDI Capability Inquiry (MIDI-CI) specification defines mechanisms and a set of Universal System Exclusive messages used for Profile Configuration (and much more). However, MIDI-CI does not define the rules for Profile Specifications or devices that implement Profiles. This document, the Common Rules for Profiles, complements MIDI-CI by defining a set of design rules for all Profile Specifications.

http://www.amei.or.jp

https://www.midi.org

Version History

Table 1 Version History

Publication Date	Version	Changes
Feb. 20, 2020	1.0	Initial release
June 15, 2023	1.1	Added Profile Details Inquiry messages. Added Profile Added Report and Profile Removed Report messages. Added Function Block Profiles and made resulting adjustments to Profile Addresses.

Contents

Tables

1 Introduction

1.1 Executive Summary

A Profile is a defined set of rules for how a MIDI device implementing the Profile shall respond to a chosen set of MIDI messages to achieve a particular purpose or to suit a particular application. This specification defines rules which are commonly applied to all Profiles. The rules also define specific details for using the Profile Configuration messages which are defined in the MIDI Capability Inquiry (MIDI-CI) specification ***[MA03]***.

1.2 Background

MIDI-CI allows devices to communicate their capabilities to each other. Devices can use that capabilities information to self-configure their MIDI connections and related settings. Profiles are a beneficial component in enabling intelligent auto-configuration.

A Profile is a defined set of rules for how a MIDI receiver device implementing the Profile shall respond to a chosen set of MIDI messages to achieve a particular purpose or to suit a particular application. In addition to defining response to MIDI messages, a Profile may optionally also define other device functionality requirements. This definition also then implies MIDI implementation of a sender or in some cases may require a defined MIDI implementation of a sender.

The most successful MIDI feature similar to a Profile in the first 3 decades of MIDI has been General MIDI. GM allows devices to "know" that a defined set of sounds is available at particular Program Change locations, that the device receives on all 16 MIDI Channels, that a Drum set is on Channel 10, and that there is defined response to a chosen set of MIDI messages.

This kind of knowledge shared between devices allows those devices to configure a more integrated level of control with increased predictability of the results that will come from sending related MIDI messages. MIDI-CI Profiles are also intended to allow more integrated cooperation between devices.

While GM is the best model of a successful profile concept prior to MIDI-CI, it does not take MIDI-CI or 2-way communication into account. There is a "GM On" message but no reply from the receiver. MIDI-CI Profiles take advantage of the 2-way communication so that a sender can know that receiver will use MIDI messages in the way the Profile specification defines.

This document defines how specific Profile specifications should be written and how devices that are compatible with MIDI-CI Profile Configuration should use Profiles.

This version of the Common Rules for Profiles defines the use of messages defined in version 1.2 of the M2-101-UM MIDI Capability Inquiry (MIDI-CI) specification ***[MA03]***.

References

1.2.1 Normative References

[MA01] ***Complete MIDI 1.0 Detailed Specification***, Document Version 96.1, Third Edition, Association of Musical Electronics Industry, *http://www.amei.or.jp/*, and The MIDI Association, *https://www.midi.org/*

[MA02] ***M2-100-U MIDI 2.0 Specification Overview***, Version 1.1, Association of Musical Electronics Industry, *http://www.amei.or.jp/*, and The MIDI Association, *https://www.midi.org/*

[MA03] ***M2-101-UM MIDI Capability Inquiry (MIDI-CI)***, Version 1.2, Association of Musical Electronics Industry, *http://www.amei.or.jp/*, and The MIDI Association, *https://www.midi.org/*

[MA04] ***M2-102-U Common Rules for MIDI-CI Profiles***, Version 1.1, Association of Musical Electronics Industry, *http://www.amei.or.jp/*, and The MIDI Association, *https://www.midi.org/*

[MA05] ***M2-103-UM Common Rules for Property Exchange***, Version 1.1, Association of Musical Electronics Industry, *http://www.amei.or.jp/*, and The MIDI Association, *https://www.midi.org/*

[MA06] ***M2-104-UM Universal MIDI Packet (UMP) Format and MIDI 2.0 Protocol***, Version 1.1, Association of Musical Electronics Industry, *http://www.amei.or.jp/*, and The MIDI Association, *https://www.midi.org/*

[MA07] ***M2-113-UM MIDI-CI Profile: Default Control Change Mapping***, Version 1.0, Association of Musical Electronics Industry, *http://www.amei.or.jp/*, and The MIDI Association, *https://www.midi.org/*

1.3 Terminology

1.3.1 Definitions

AMEI: Association of Musical Electronics Industry. Authority for MIDI Specifications in Japan.

Device: An entity, whether hardware or software, which can send and/or receive MIDI messages.

Group: A field in the UMP Format addressing some UMP Format MIDI messages (and some UMPs comprising any given MIDI message) to one of 16 Groups. See the M2-104-UM Universal MIDI Packet (UMP) Format and MIDI 2.0 Protocol specification ***[MA06]***.

Initiator: One of two MIDI-CI Devices with a bidirectional communication between them. Initiator has the management role of setting and negotiating parameters for interoperability between the two Devices. The primary goal of Initiator is usually (but not strictly required to be) configuring two Devices for subsequent communication from Initiator as MIDI transmitter to Responder as MIDI receiver. The role of Initiator and Responder may alternate between the two MIDI-CI Devices. Either MIDI-CI Device may initiate a MIDI Transaction (act as Initiator) at any time. Also see Responder.

Inquiry: A message sent by an Initiator to begin a Transaction.

MA: See MIDI Association.

MIDI 1.0 Protocol: Version 1.0 of the MIDI Protocol as originally specified in ***[MA01]*** and extended by MA and AMEI with numerous additional MIDI message definitions and Recommended Practices. The native format for the MIDI 1.0 Protocol is a byte stream, but it has been adapted for many different Transports. MIDI 1.0 messages can be carried in UMP packets. The UMP format for the MIDI 1.0 Protocol is defined in the M2-104-UM Universal MIDI Packet (UMP) Format and MIDI 2.0 Protocol specification ***[MA06]***.

MIDI 1.0 Specification: Complete MIDI 1.0 Detailed Specification, Document Version 96.1, Third Edition ***[MA01]***.

MIDI 2.0: The MIDI environment that encompasses all of MIDI 1.0, MIDI-CI, Universal MIDI Packet (UMP), MIDI 2.0 Protocol, MIDI 2.0 messages, and other extensions to MIDI as described in AMEI and MA specifications.

MIDI 2.0 Protocol: Version 2.0 of the MIDI Protocol. The native format for MIDI 2.0 Protocol messages is UMP as defined in M2-104-UM Universal MIDI Packet (UMP) Format and MIDI 2.0 Protocol specification ***[MA06]***.

MIDI Association: Authority for MIDI specifications worldwide except Japan. See MIDI Manufacturers Association.

MIDI-CI: MIDI Capability Inquiry ***[MA03]***, a specification published by The MIDI Association and AMEI.

MIDI-CI Device: A Device that has the ability to act as a Responder that replies to inquiries received from an Initiator. The ability to act as an Initiator is recommended but optional.

MIDI-CI Transaction: A Transaction using a set of MIDI-CI messages that includes an Inquiry sent by an Initiator and a reply to the Inquiry returned by the Responder. The Responder's reply to an Inquiry might be a single message that satisfies the Inquiry, a set of multiple messages that satisfy the Inquiry, or an error message. See also Transaction.

MIDI Endpoint: A Device which is an original source of MIDI messages or final consumer of MIDI messages.

MIDI In: A hardware or software MIDI connection used by a MIDI Device to receive MIDI messages from a MIDI Transport.

MIDI Out: A hardware or software MIDI connection used by a MIDI Device to transmit MIDI messages to a MIDI Transport.

MIDI Manufacturers Association: A California nonprofit 501(c)6 trade organization, and the legal entity name of the MIDI Association.

MIDI Port: The source or sink of stream of MIDI data in the MIDI 1.0 data format.

MIDI Transport: A hardware or software MIDI connection used by a Device to transmit and/or receive MIDI messages to and/or from another Device.

MMA: See MIDI Manufacturers Association.

MUID (MIDI Unique Identifier): A 28-bit random number generated by a Device used to uniquely identify the Device in MIDI-CI messages sent to or from that Device.

NRPN: Non-Registered Parameter Number, a type of controller message defined in the MIDI 1.0 Protocol ***[MA01]***. NRPNs have equivalent messages in the MIDI 2.0 Protocol, called Assignable Controllers (see ***[MA06]***).

PE: Property Exchange.

Port: See MIDI Port.

Profile: An MA/AMEI specification that includes a set of MIDI messages and defined responses to those messages. A Profile is controlled by MIDI-CI Profile Negotiation Transactions. A Profile may have a defined minimum set of mandatory messages and features, along with some optional or recommended messages and features. See the MIDI-CI specification ***[MA03]*** and the Common Rules for MIDI-CI Profiles ***[MA04]***.

Property: A JSON key-value pair used by Property Exchange, for example `"channel": 1`.

Property Exchange: A set of MIDI-CI Transactions by which one device may access Property Data from another device.

Protocol: There are two defined MIDI Protocols: the MIDI 1.0 Protocol and the MIDI 2.0 Protocol, each with a data structure that defines the semantics for MIDI messages. See ***[MA01]*** and ***[MA06]***.

Receiver: A MIDI Device which has a MIDI Transport connected to its MIDI In.

Resource: A defined collection of one or more PE Properties with an associated inquiry to access its Properties.

Responder: One of two MIDI-CI Devices with a bidirectional communication between them. The Responder is the Device that receives an Inquiry message from an Initiator Device as part of a MIDI-CI Transaction and acts based on negotiation messages managed by the Initiator Device. Also see Initiator.

RPN: Registered Parameter Number, a type of controller message defined in the MIDI 1.0 Protocol ***[MA01]***. RPNs have equivalent messages in the MIDI 2.0 Protocol, called Registered Controllers (see ***[MA06]***).

Sender: A MIDI Device which transmits MIDI messages to a MIDI Transport which is connected to its MIDI Out or to its MIDI Thru port.

Source: A Source is a Sender which originates or generates MIDI messages. A Source does not include a Sender which is retransmitting messages which originated in another MIDI Device.

Transaction: An exchange of MIDI messages between two MIDI Devices with a bidirectional connection. All the MIDI messages in a single Transaction are associated and work together to accomplish one function. The simplest Transaction generally consists of an inquiry sent by one MIDI Device and an associated reply returned by a second MIDI Device. A Transaction may also consist of an inquiry from one MIDI Device and several associated replies from a second MIDI Device. A Transaction may be a more complex set of message exchanges, started by an initial inquiry from one MIDI Device and multiple, associated replies exchanged between the first MIDI Device and a second MIDI Device. Also see MIDI-CI Transaction.

UMP: Universal MIDI Packet, see ***[MA06]***.

UMP Format: Data format for fields and messages in the Universal MIDI Packet, see ***[MA06]***.

UMP MIDI 1.0 Device: any Device that sends or receives MIDI 1.0 Protocol messages using the UMP ***[MA06]***. Such Devices may use UMP Message Types that extend the functionality beyond Non-UMP MIDI 1.0 Systems.

Universal MIDI Packet (UMP): The Universal MIDI Packet is a data container which defines the data format for all MIDI 1.0 Protocol messages and all MIDI 2.0 Protocol messages. UMP is intended to be universally applicable, i.e., technically suitable for use in any transport where MA/AMEI elects to officially support UMP.

For detailed definition see M2-104-UM Universal MIDI Packet (UMP) Format and MIDI 2.0 Protocol specification *[MA06]*.

1.3.2 Reserved Words and Specification Conformance

In this document, the following words are used solely to distinguish what is required to conform to this specification, what is recommended but not required for conformance, and what is permitted but not required for conformance:

Table 2 Words Relating to Specification Conformance

Word	Reserved For	Relation to Specification Conformance
shall	Statements of requirement	**Mandatory** A conformant implementation conforms to all 'shall' statements.
should	Statements of recommendation	**Recommended but not mandatory** An implementation that does not conform to some or all 'should' statements is still conformant, providing all 'shall' statements are conformed to.
may	Statements of permission	**Optional** An implementation that does not conform to some or all 'may' statements is still conformant, providing that all 'shall' statements are conformed to.

By contrast, in this document, the following words are never used for specification conformance statements; they are used solely for descriptive and explanatory purposes:

Table 3 Words Not Relating to Specification Conformance

Word	Reserved For	Relation to Specification Conformance
must	Statements of unavoidability	Describes an action to be taken that, while not required (or at least not directly required) by this specification, is unavoidable. Not used for statements of conformance requirement (see 'shall' above).
will	Statements of fact	Describes a condition that as a question of fact is necessarily going to be true, or an action that as a question of fact is necessarily going to occur, but not as a requirement (or at least not as a direct requirement) of this specification. Not used for statements of conformance requirements (see 'shall' above).
can	Statements of capability	Describes a condition or action that a system element is capable of possessing or taking. Not used for statements of conformance permission (see 'may' above).
might	Statements of possibility	Describes a condition or action that a system element is capable of electing to possess or take. Not used for statements of conformance permission (see 'may' above).

2 Profile Mechanisms

This section defines device design for using Profiles, implementing MIDI-CI Profile Configuration messages for enabling and disabling Profiles.

Table 4 Overview: Steps to Implement and Use a Profile

	Device Design	**Profile Configuration** (Setting up the device to use the Profile)	**Using the Profile**
Understand / Implement the Profile in a Device	Conform to MIDI-CI and a Profile Specification		
Get Profiles (with Level w. Minimum Required)		Profile Inquiry Profile Added Report Profile Removed Report	
Get Optional Features Supported		Profile Details Inquiry	
Get Channels for Profile		Profile Details Inquiry	
Set and Report Channels for Profile		Set Profile On / Enabled Report	
Enable/Disable Profile		Set Profile On/Off	
Send MIDI data in usage of the Profile.			Send Channel Voice Messages
			Send System Messages
			Send SysEx, Sysex8, and MDS messages
			Send Profile Specific Data Message
Get list of Controllers Supported with Values			Process Inquiry: MIDI Message Report
Get list of Controllers Supported with Destination Parameter			Property Exchange Controller Resources
Get/Set Complex Parameters			New Property Exchange Resource specific to Profile

2.1 Discovering and Configuring Supported Profiles

MIDI-CI inquiries allow an Initiator to ask for a list of Profiles that a Responder supports. The Initiator may use this information to auto-configure the connection for greater interoperability of specific applications. The Initiator manages the configuration by enabling and disabling particular Profiles on the Responder. When a Responder changes its Profiles in response to an event that is not managed by the Initiator, the Responder informs the Initiator.

For example: a software sound editing application might configure a specific set of on-screen controls to send associated MIDI messages that control parameters of any device that it discovers which conforms to the Drawbar Organ Profile. It would configure a different set of on-screen controls and MIDI messages if it found that the device conforms to an Electric Piano Profile.

To discover Profiles supported by a Responder, the Initiator sends a Profile Inquiry. The Responder replies with one or more Reply to Profile Inquiry messages to report supported Profiles. These inquiries and replies have various Address field values to declare Profiles supported at different locations in the device. See Section 2.3. The final reply from the Responder is the declaration of Profiles available at Address field value 0x7F.

2.2 Profile Identifier (ID)

The MMA and AMEI shall maintain the list of Standard Defined Profiles and assign an official Profile ID to each Standard Defined Profile.

Standardized Profiles and Manufacturers' proprietary Profiles are identified in MIDI-CI Profile messages in a set of 5 bytes:

Table 5 Profile Identifier Data Format

Byte Number	Standard Defined Profiles	Manufacturer Specific Profiles
Profile ID Byte 1	0x7E Standard Defined Profile	Manufacturer SysEx ID 1
Profile ID Byte 2	Profile Number Bank	Manufacturer SysEx ID 2
Profile ID Byte 3	Profile Number	Manufacturer SysEx ID 3
Profile ID Byte 4	Profile Version	Manufacturer-Specific Info 1
Profile ID Byte 5*	Profile Level*	Manufacturer-Specific Info 2

**See [MA03] for more details of Profile ID Byte 5, Profile Level.*

Standardized Profiles defined by AMEI and MMA use a Profile ID as defined in the Common Rules for MIDI Profiles (this specification) and by each Profile Specification. The value of the Profile ID Byte 1 for standardized Profiles is 0x7E (Universal). Values for Bytes 2 and 3 are assigned by MMA and AMEI. Byte 4 is defined by each Profile specification. Byte 5 is defined by the "Profile Level and Minimum Requirements" section of this specification and each Profile specification.

Manufacturers may use MIDI-CI to control Profiles of their own proprietary design by placing their own 3-byte System Exclusive ID in Profile ID Byte 1, Profile ID Byte 2, and Profile ID Byte 3. For System Exclusive ID values that are only 1 byte in length, the System Exclusive ID value is placed in the Profile ID Byte 1 and the remaining 2 ID Bytes are filled with zeroes: ID 00 00. Then every manufacturer defines their own usage of Profile ID Byte 4 and Profile ID Byte 5.

2.3 Devices, Function Blocks, Groups, Ports, and MIDI Channels

Some MIDI-CI inquiries allow a device to report Profiles that are supported on a single Channel, multiple Channels, on per Group basis and/or on a per Function Block basis.

MIDI-CI Profile Configuration messages can be addressed on a Channel, Group wide or Function Block wide basis. Addressing is done using the Device ID of the Universal System Exclusive messages:

0x7F = Send to/from Function Block

0x7E = Send to/from the Group (or Port)

0x00-0x0F = Send to/from Channels 1-16

There are four MIDI Channel structures for Profiles, each of which can be enabled using those MIDI-CI addresses, as defined in Sections 2.3.1 through 2.3.4.

See the MIDI-CI specification v1.2 ***[MA03]*** for more about addressing MIDI-CI messages.

2.3.1 Single Channel Profile

Some Profiles define Device functions that use only one MIDI Channel for the Profile. Piano is a typical example.

> A MIDI-CI Initiator asks what Profiles are supported using a target Channel Device ID address 0x00 – 0x0F. The Responder replies with the same Device ID. Initiator sends Profile Enable message or Profile Disable message with the Device ID address of the target MIDI Channel 0x00-0x0F. Responder sends Profile Enabled message or Profile Disabled message with the Device ID address of the MIDI Channel 0x00-0x0F.

2.3.2 Group Profiles: All Channels in a Single Group

All Channel Profiles: Some Profiles define Device functions that use all 16 Channels of a Group or MIDI Port. General MIDI is a typical example.

> A MIDI-CI Initiator asks what Profiles are supported using the Group Device ID address 0x7E. The Responder replies with the Device ID address 0x7E. Initiator sends Profile Enable message or Profile Disable message with the Device ID address 0x7E. Responder sends Profile Enabled message or Profile Disabled message with the Device ID address 0x7E.

2.3.3 Function Block Profiles: All Channels in a Function Block

All Channel Profiles: Some Profiles define Device functions that use all Channels of a Function Block which may span more than one Universal MIDI Packet Group.

> A MIDI-CI Initiator asks what Profiles are supported using the Device ID address 0x7F. The Responder replies with the Device ID address 0x7F. Initiator sends Profile Enable message or Profile Disable message with the Device ID address 0x7F. Responder sends Profile Enabled message or Profile Disabled message with the Device ID address 0x7F.

2.3.4 Multi-Channel Profiles: Two or More Channels

Some Profiles define Device functions that use multiple MIDI Channels for the Profile but is not a Group Profile or a Function Block Profile. Guitar is a typical example that might be defined to use 7 Channels: 6 Member Channels for the 6 strings plus a Manager Channel for controlling total output volume. Enabling and Disabling the Profile is also performed on the Manager Channel. See Section 2.3.4.1"Manager Channel and Member Channels"

> A MIDI-CI Initiator asks what Profiles are supported using a target Manager Channel Device ID address 0x00-0x0F. The Responder replies with the same Device ID. Initiator sends Profile Enable message or Profile Disable message with the Device ID address of the target Manager Channel, 0x00-0x0F. Responder sends Profile Enabled message or Profile Disabled message with the Device ID address of the Manager Channel, 0x00-0x0F.

If a Multi-Channel Profile does not define a fixed number of Channels, then the Profile Details Inquiry mechanisms should be used to determine the number of Channels that a Responder will use when the Profile is enabled. See Section 2.5.1.

2.3.4.1 Manager Channel and Member Channels

Profile Specifications which define the use of 2 or more Channels, as described in Section 2.3.4 shall define a Manager Channel. The Manager Channel shall be the lowest numbered Channel in a series, on the lowest numbered Group in the series. MIDI-CI Profile Configuration messages are sent on the Manager Channel.

All other active Channels in the Profile are considered Member Channels. The Profile Specification shall define how Member Channels used by the Profile are identified in relation to the Manager Channel.

In most cases a Manager Channel will be independent from the Member Channels. Typical Manager Channel and relationship to Member Channels is often as described in this example:

A hypothetical five-string Banjo Profile might define that the five Member Channels in sequence starting above the Manager Channel are used for messages related to the five strings. Main volume and tone of the whole Banjo and other similar messages are typically sent on the Manager Channel. Note related control data such as Note On/Off and Pitch Bend for each of the five strings are sent on the Member Channels.

If the Banjo Profile Responder has declared a Function Block which only has a single Group or has not declared a Function Block, then the Manager Channel may be assigned between Channel 1 and Channel 11. MIDI-CI Profile Configuration messages are sent on the Manager Channel (1 to 11), and it is known by the definition in the Profile Specification that the five strings are on the five Channels sequentially higher than that Manager Channel.

In some cases, the Manager Channel may also be a Member Channel, only if Manager functions do not overlap or conflict with Member Channel functions. One typical use case would be a Profile which defines no Manager Functions other than MIDI-CI messages for Profile Configuration. The Profile specification shall clearly define which functions on that Channel are Manager controls and which are Member controls. For example:

When the Manager Channel is also a Member Channel:

1. If the Volume message (CC#7) on the Manager Channel affects total output volume of all Member Channels, then all the Member Channels shall ignore Volume messages (CC#7) on their own Channels.

Or

2. the Volume messages (CC#7) on Member Channels controls the volume of that one Channel, then the Manager Channel shall use Volume message (CC#7) as the volume of that one Channel and not use it for total volume of all Channels.

2.4 Common Profile Configuration Messages (MIDI-CI Messages)

All Profiles Specifications and compatible Devices should support the Common Profile Messages that are defined in the MIDI Capabilities Inquiry specification.

Also see the MIDI-CI Specification ***[MA03]*** for more details of MIDI-CI Profile Configuration messages.

Table 6 MIDI-CI Profile Configuration Messages

Universal System Exclusive Sub ID #1: MIDI-CI	Universal System Exclusive Sub ID #2: MIDI-CI Message Number	Function	Device Implementation Requirement
0x0D	0x20	Profile Inquiry	Mandatory
0x0D	0x21	Reply to Profile inquiry	Mandatory
0x0D	0x26	Profile Added Report	Optional
0x0D	0x27	Profile Removed Report	Optional
0x0D	0x28	Profile Details Inquiry	Optional
0x0D	0x29	Reply to Profile Details Inquiry	Optional
0x0D	0x22	Set Profile On	Mandatory
0x0D	0x23	Set Profile Off	Mandatory
0x0D	0x24	Profile Enabled Report	Mandatory
0x0D	0x25	Profile Disabled Report	Mandatory

0x20 Profile Inquiry

A device (Initiator) sends this to request a list of Profiles that are supported by a connected device (Responder). The Initiator may use this information to auto-configure the connection between the devices for increased interoperability. The message may be addressed to:

- An individual Channel (addresses 0x00-0x0F) to find the Channel Profiles available on that Channel and the Multi-Channel Profiles which can have a Manager Channel on that Channel, or
- to a Group (address 0x7E) to find the Group Profiles available, or
- to the Function Block (address 0x7F) to discover all Profiles on the Function Block, including Function Block Profiles, Group Profiles, Multi-Channel Profiles, and Single Channel Profiles. (See Section *2.3* for more information about addressing)

> *Note: While the Profile Inquiry addressed to 0x7F requests all Profiles regardless of MIDI Channel structure (See Section* ***2.3****), a Reply to Profile Inquiry message (below) addressed to 0x7F only declares Function Block Profile types. Other Profile types are declared in separate replies.*

0x21 Reply to Profile Inquiry

When a Responder receives the Profile Inquiry message it shall reply with one or more Reply to Profile inquiry messages to report a list of Profiles the Responder supports.

The Profiles supported by a Device at the address requested are declared as 2 lists as shown in the following table:

Table 7 Lists of Profiles Supported

Currently Enabled Profiles Supported (Profiles which are active)	**Currently Disabled Profiles Supported** (Profiles which are available but NOT active)
Profile A Profile B etc.	Profile C Profile D Profile E Profile F etc.
	Note: • *Profiles may be added to this list by the Profile Added Report message.* • *Profiles may be removed from this list by the Profile Removed Report message.*
Note: • *Profiles may be changed from Currently Disabled to Currently Enabled by the Set Profile On and Profile Enabled messages.* • *Profiles may be changed from Currently Enabled to Currently Disabled by the Set Profile Off and Profile Disabled messages.*	

When a Responder receives the Profile Inquiry message on a specific Channel address 0x00-0x0F, the Responder sends this Reply to Profile Inquiry message addressed with that Channel to report the Channel Profiles supported on the requested Channel (See Section 2.3.1) and the Multi-Channel Profiles which can have a Manager Channel on that Channel. If the Responder does not support any Profiles on that Channel, then it shall reply with no Profiles reported.

When a Responder receives the Profile Inquiry message on a specific Group address 0x7E, the Responder sends this Reply to Profile Inquiry message addressed with that Group to report the Group Profiles supported on the requested Group (See Section 2.3.2). If the Responder does not support any Group Profiles on that Group, then it shall reply with no Profiles reported.

When a Responder receives the Profile Inquiry message on the Function Block address 0x7F, it shall reply with one or more Reply to Profile Inquiry messages in the following order:

1. If the Responder supports Single Channel Profiles and/or Multi-Channel Profiles which can have a Manager Channel on that Channel within the Function Block, then it sends Reply to Profile Inquiry messages on those Channels first.
2. If the Responder supports any Group Profiles within the Function Block, then it sends Reply to Profile Inquiry messages with address 0x7E to report Group Profiles supported on each Group of the Function Block.
3. As a final reply, the Responder shall send a Reply to Profile Inquiry message with address 0x7F to report Function Block Profiles that are supported by the Device. If the Responder does not support any Function Block Profiles or does not support any Profiles at all, it shall send this reply on the address 0x7F with no Profiles reported to indicate that no further replies will follow.

0x26 Profile Added Report

While a Device is active, it might change its state such that a Profile is newly added to its capabilities. If the Device has previously acted as a Responder by sending a Reply to Profile Inquiry message, then the Device should send a Profile Added Report message to update the list of Currently Disabled Profile Supported declared in the previous Reply to Profile Inquiry message. See Table 7 Lists of Profiles Supported.

The Profile Added Report message shall be sent with the Destination MUID set to Broadcast (0x7F 7F 7F 7F).

If the newly added Profile is enabled in the Device, then the Device shall then send a Profile Enabled Report message for that Profile immediately following the Profile Added Report.

0x27 Profile Removed Report

While a Device is active, it might change its state such that it is no longer capable of supporting a Profile. If the Device has previously acted as a Responder by sending a Reply to Profile Inquiry message, then the Device should send a Profile Removed Report message to update the list of Currently Disabled Profile Supported declared in the previous Reply to Profile Inquiry message. See Table 7 Lists of Profiles Supported.

The Profile Removed Report message shall be sent with the Destination MUID set to Broadcast (0x7F 7F 7F 7F).

If the Device is removing a Profile which is currently enabled, then the Device shall send a Profile Disabled Report message for that Profile before sending the Profile Removed Report message.

0x22 Set Profile On

An Initiator sends this to enable a Profile on a Responder.

0x23 Set Profile Off

An Initiator sends this to disable a Profile on a Responder.

0x24 Profile Enabled Report

A device sends this message if it has enabled a Profile.

This is an acknowledgement upon receipt of a Set Profile On message.

This is an informative message if an event local to the device enables a Profile.

> Example: A change of patch might trigger a change of Profile.

0x25 Profile Disabled Report

A device sends this message if it has disabled a Profile.

This is an acknowledgement upon receipt of a Set Profile Off message.

This is an informative message if an event local to the device disables a Profile.

Example: A change of patch might trigger a change of Profile.

0x28 Profile Details Inquiry Message

An Initiator sends this to discover details of the Profile implementation of a Responder. An Initiator may make this inquiry to retrieve useful information from the Responder before instructing the Responder to enable a Profile. This message may also be used to discover details about a Responder's Profile implementation while it is enabled.

For example, an Initiator might discover:

1. How many Channels may be used or assigned by the Responder for a Multi-Channel Profile.
2. Which optional Profile features the Responder is capable of supporting.

The details which are being requested are determined by the Inquiry Target field. See Section 2.5.

The details which may be discovered are defined by the specification for the requested Profile Id.

If a Responder which supports Profile Configuration receives a Profile Details Inquiry message addressed to a Profile Identifier which is not supported by the Responder, the Responder should reply by sending a MIDI-CI NAK message with a Status Code value 0x04.

0x29 Reply to Profile Details Inquiry Message

When a Responder receives the Profile Details Inquiry message it shall send this message with data appropriate to the Inquiry Target in the Profile Details Inquiry message.

If a Target Data definition allows it, the Reply to Profile Details message for that Target Data may also be sent without prior receipt of a Profile Details Inquiry message. It is then used as a Notification message from a device for the Target Data (See Section 2.5).

2.5 Profile Details Inquiry: Inquiry Target

The Profile Details Inquiry mechanism allows an Initiator to get information from the Responder about its Profile-related implementation. There are two types of information which may be requested (the target):

1. Registered Target Data: the format of the reply is defined by MIDI Association and AMEI in this Common Rules for Profiles specification and designed for use by many Profiles.
2. Profile Specific Target Data: the format of the reply is defined by the Profile specification and is unique to that Profile.

Registered Target Data definitions in this Common Rules for Profiles specification shall define whether a Reply to Profile Details Inquiry Message may also be used as a Notification for that Target Data.

A Profile specification shall define whether a Reply to Profile Details Inquiry Message may be also used as a Notification for that Target Data.

Following are the definitions of Registered Target Data. In this revision of the Common Rules for Profiles there is one Inquiry Target defined. Other Registered Target Data may be defined in future revisions of this specification.

2.5.1 Profile Details Inquiry Target Data: Number of MIDI Channels

Inquiry Target field = 0x00

This Registered Target Data allows an Initiator and Responder to negotiate the number of Channels which will be allocated when enabling a multichannel Profile which may have a variable number of channels as described in Section 2.3.4 Multi-Channel Profiles: 2-256 Channels.

A Reply to Profile Details Inquiry Message with this Target Data shall not be sent as a Notification.

When the Responder receives a Profile Details Inquiry message with the Inquiry Target field set to 0x00, the inquiry is Number of MIDI Channels. The reply data has 2 defined fields which contain the data requested:

Table 8 Reply to Profile Details Inquiry Message, Maximum Number of Channels

Value	Parameter
Other defined MIDI-CI message fields here.	
5 bytes	Profile Id
0x00	Inquiry Target = Number of MIDI Channels
0x04 0x00	Inquiry Target Data length = 4
2 bytes	The number of Channels currently in use by this Profile. Value = Total Number of Channels, including Manager and Member Channels. (LSB First). If the Profile is not currently enabled, set to 0x00 0x00.
2 bytes	Maximum Number of Channels (available for use by this Profile). Value = Total number of Channels, including Manager and Member Channels. (LSB first)
0xF7	End Universal System Exclusive

After the Initiator receives the Reply to Profile Details Inquiry Message with Maximum Number of MIDI Channels and when the Initiator sends an Enable Profile message. The Initiator shall not set the Number Channels Requested field in the Enable Profile message to a number higher than the Responder declared in the Maximum Number of Channels field.

The complete steps to negotiate the Number of MIDI Channels used by a Profile are as follows:

Step 1: Initiator Sends Profile Details Inquiry Message

Table 9 Negotiating Number of Channels Step 1

Value	Parameter
F0	System Exclusive Start
7E	Universal System Exclusive
1 byte	Device ID: Source or Destination (depending on type of message): 00–0F: To/from MIDI Channels 1-16 (set to desired Manager Channel)
0D	Universal System Exclusive Sub-ID#1: MIDI-CI
0x28	Universal System Exclusive Sub-ID#2: Inquiry: Profile Details Inquiry Message
1 byte	MIDI-CI Message Version/Format
4 bytes	Source MUID (LSB first)
4 bytes	Destination MUID (LSB first)
5 bytes	Profile Id

0x00	Inquiry Target = Number of MIDI Channels
F7	End Universal System Exclusive

Step 2: Responder Sends Reply to Profile Details Inquiry Message

Table 10 Negotiating Number of Channels Step 2

Value	Parameter
F0	System Exclusive Start
7E	Universal System Exclusive
1 byte	Device ID: Source or Destination (depending on type of message): 00–0F: To/from MIDI Channels 1-16 (set to requested Manager Channel)
0D	Universal System Exclusive Sub-ID#1: MIDI-CI
0x29	Universal System Exclusive Sub-ID#2: Inquiry: Reply to Profile Details Message
1 byte	MIDI-CI Message Version/Format
4 bytes	Source MUID (LSB first)
4 bytes	Destination MUID (LSB first)
5 bytes	Profile Id
0x00	Inquiry Target = Number of MIDI Channels
0x04 0x00	Inquiry Target Data length = 4
2 bytes	The number of Channels currently in use by this Profile. Value = Total Number of Channels, including Manager and Member Channels. (LSB First). If the Profile is not currently enabled, set to 0x00 0x00.
2 bytes	Maximum Number of Channels (available for use by this Profile). Value = Total number of Channels, including Manager and Member Channels. (LSB first)
F7	End Universal System Exclusive

Step 3: Initiator Sends Set Profile On Message

Table 11 Negotiating Number of Channels Step 3

Value	Parameter
0xF0	System Exclusive Start
0x7E	Universal System Exclusive
1 byte	Destination 00–0F: To/from MIDI Channels 1-16 (set to desired Manager Channel)
0x0D	Universal System Exclusive Sub-ID#1: MIDI-CI
0x22	Universal System Exclusive Sub-ID#2: Set Profile On

1 byte	MIDI-CI Message Version/Format
4 bytes	Source MUID (LSB first)
4 bytes	Destination MUID (LSB first)
5 bytes	Profile ID of Profile to be Set to On (to be enabled)
The following fields (except F7 End) were added in MIDI-CI Message Version 2	
2 bytes	Number Channels Requested (LSB First) to assign to this Profile when it is enabled
0xF7	End Universal System Exclusive

The value of the Number of Channels field shall not be higher than the Maximum Number of Channels declared by the Responder in Step 2.

Step 4: Responder Sends Profile Enabled Message

Table 12 Negotiating Number of Channels Step 4

Value	Parameter
0xF0	System Exclusive Start
0x7E	Universal System Exclusive
1 byte	Destination 00–0F: To/from MIDI Channels 1-16 (set to enabled Manager Channel)
0x0D	Universal System Exclusive Sub-ID#1: MIDI-CI
0x24	Universal System Exclusive Sub-ID#2: Inquiry: Profile Enabled
0x02	MIDI-CI Message Version/Format
4 bytes	Source MUID (LSB first)
4 bytes	Destination MUID (LSB first)
5 byte	Profile Id
The following fields (except F7 End) were added in MIDI-CI Message Version 2	
2 bytes	Number Channels enabled on this Profile. (Manager + Member Channels, LSB first)
0xF7	End Universal System Exclusive

Step 5: Profile Enabled

Initiator knows that the Profile is enabled and how many Channels have been allocated.

2.6 Enabling Profiles

A device shall enable any Profile it supports after it receives the associated Set Profile On Message. After the device has enabled the Profile, it shall reply with a Profile Enabled message.

If any action local to a device enables a particular Profile, the device shall send the associated Profile Enabled message.

If a device receives a Program Change message that causes a specific Profile to be enabled, the device shall send the associated Profile Enabled message.

Reply When Responder Cannot Enable or Does Not Support a Profile:

- If a device receives a Set Profile On message for a Profile that it supports but the device is unable to enable that Profile, then the device shall reply with a Profile Disabled Report.
- If a device receives a Set Profile On message for a Profile that it does not support, then it should reply with a MIDI-CI NAK Message.

2.7 Disabling Profiles

A device should disable a Profile after it receives the associated Set Profile Off Message from an Initiator. After the device has disabled the Profile, it shall reply with a Profile Disabled message. However, if a device is not able to disable a particular Profile, then it shall reply with a Profile Enabled message.

If any action local to a device disables an enabled Profile, the device shall send the associated Profile Disabled message.

If a device receives a Program Change message that causes a specific Profile to be disabled, the device shall send the associated Profile Disabled message.

Reply When Device Cannot Disable or Does Not Support a Profile:

- If a device receives a Set Profile Off message for a Profile that it supports but the device is unable to disable that Profile, then the device shall reply with a Profile Enabled Report.
- If a device receives a Set Profile Off message for a Profile that it does not support, then it should reply with a MIDI-CI NAK Message.

2.8 Mutually Exclusive Profiles

Some combination of Profiles can be enabled simultaneously on one MIDI Channel. But in some cases, a device might not be able to support particular combinations of simultaneous Profiles. A device may report that it cannot support the simultaneous combination of 2 requested Profiles.

- Example 1: For many devices it would be reasonably possible to allow Electric Piano Profile and Guitar Effects Profile to exist simultaneously on the same MIDI Channel, perhaps for a sound of an Electric Piano going through Phaser + Overdrive.
- Example 2: For many devices it would not be feasible to allow Electric Piano Profile and Drawbar Organ Profile to exist simultaneously on the same MIDI Channel. Many devices may not be able to do both at the same time. On the other hand, some devices may allow this combination, perhaps if the Notes played on that Channel trigger a layered sound of Electric Piano + Drawbar Organ.

If a device has a Profile (Profile A) enabled and it receives a Profile On message for another Profile (Profile B) that cannot be simultaneously supported with Profile A, the device should switch to the new Profile and send 2 replies to the Profile On message:

1. Profile Disabled Report for Profile A
2. Profile Enabled Report for Profile B

2.9 Avoiding Excessive Profile Configuration Reports

Some Devices allow a user to quickly scroll through a list of many Programs. If such a Device changes Profiles based on the choice of Programs, the Device should avoid sending Profile Configuration messages for every program in the list which is scrolled.

Such a Device should reasonably limit Profile Configuration messages to those which match a user’s intended selection. One potential implementation is to wait some short time after a program has been selected before sending Profile Enabled Report, Profile Disabled Report, Profile Added Report, and Profile Removed Report.

3 Rules for Profile Definitions

Each Standard Defined Profile is defined in a separate Defined Profile specification document for that Profile. Each Profile specification defines the factors that allow a high level of interoperability between devices that share support for that Profile.

3.1 Profile Specifications

A Profile Specification defines the Receiver device implementation of specific MIDI messages including but not limited to:

- Response to Note On/Off messages
- Profile Specific Note On/Off Attributes
- Response to messages that influence note tuning
- Response to Control Change messages
- Response to specific Registered Controller (RPN) messages
- Response to Bank Select and Program Change messages
- Response to System messages
- Any applicable Universal System Exclusive messages
- Any Profile-specific System Exclusive messages

A Profile Specification may also define other Receiver implementation requirements including but not limited to:

- Device functional entities and topology
- Minimum Polyphony required
- Number of MIDI Channels supported (See Sections 2.3 and 2.5.1)
- Message response times or other performance standards
- Other Non-MIDI data types supported (example: audio samples)
- Specific sounds or programs

3.1.1 Control Change, Registered Controllers (RPN), and Assignable Controllers (NRPN)

Control Change messages should not gain new definitions within a Profile but should be reserved for legacy applications.

As a general rule (but not a hard requirement) new Profiles should define Registered Controller (RPN) messages to control parameters that have not previously been defined.

Manufacturer-Specific Profiles should use Assignable Controllers (NRPN) messages to control any function that does not have a Control Change or Registered Controller (RPN) defined for that function.

3.2 Profile Support Level and Minimum Requirements

A Profile may define several levels of compatibility or compliance.

Profile specifications shall clearly define the set of properties or features that are the minimum requirements for a device to claim that it supports that Profile.

Profile specifications may also define extended properties or features that are optional, available for common use but not absolutely required for a device to claim that it conforms to that Profile.

Profile Specifications define which properties or features belong to defined levels of support that are indicated by "Profile ID Byte 5, Profile Level" in the 6 Common Profile Messages of MIDI-CI. See the following table.

Table 13 Levels of Profile Supported

Profile ID Byte 5 Profile Level	Profile Support Level
0x00	Partial
0x01	Minimum Required
0x02-7E	Minimum Required plus optional, Extended Feature Sets as defined by the Profile specification.
0x7F	Highest Possible

Partial

Reported by a Receiver that it generally supports the Profile but lacks some part(s) of the defined Minimum Required. The manufacturer believes the device offers valuable features of the Profile or that primary application of the Profile functionality is delivered even if Profile compliance is not 100%.

If a Receiver reports that it has Partial Support Level, then after Enabling the Profile, the Sender should send messages to the Receiver as though the Receiver meets the Minimum Required. Sender or Receiver may choose to display a notice to the user that complete Profile function might not be achieved.

Minimum

Reported by a Receiver that fully implements all the Profile features that are defined as the Minimum Required.

Extended Feature Sets

Profile specification may define specific extended feature sets in other Profile Levels of support. The Profile Support Level of each extended feature of the Profile should be clearly defined. Profile designers should give careful consideration whether implementation with some extended features should be defined as an Extended Feature Set or whether there is sufficient functionality to warrant a separate Profile specification. If there are credible use cases for the Extended Features without using the Base Level features, then a separate Profile specification is probably warranted.

Highest Possible

A Sender uses the value 0x7F for the Profile Level Support field when sending a Set Profile On message. When a Receiver receives a Set Profile On message with Highest Possible Level value set, the Receiver should enable all features of the Profile that it supports.

3.2.1 Device Instrument Definitions and Channel Structures

A Profile shall define the use of only one of the following MIDI Channel structures:

- Single Channel Profile as defined in Section 2.3.1
- Group Profile as defined in Section 2.3.2
- Function Block Profile as defined in Section 2.3.3
- Multi-Channel Profile as defined in Section 2.3.4

However, some device types may be defined for more than one of those structures above.

> Hypothetical Example 1: A classic Drawbar Organ is a 3-manual instrument that would usually use 4 MIDI Channels (Manager, Upper, Lower, Pedals). But there are many single manual Drawbar organs and many multi-Channel synthesizers which can assign a Drawbar Organ to just 1 Channel.
>
> Hypothetical Example 2: The original General MIDI specification was defined for 16 Channel applications, but MIDI-CI allows a GM profile to be enabled on a single Channel.

Multiple Profiles, with separate Profile Identifier values, are used to differentiate between the different MIDI Channel structures.

> Hypothetical Example 1: A Single Manual Drawbar Organ vs. a 3-Manual Drawbar Organ, each having its own Profile ID Number:
>
> > Three Manual Drawbar Organ Profile (3-Channels with Manager Channel on Swell/Upper Manual)
> >
> > Single Manual Drawbar Organ Profile
>
> Hypothetical Example 2: There could be 3 separate but related Profiles for General MIDI Level 2, each with a unique Profile ID Number:
>
> > General MIDI Level 2 Profile (16 Channels)
> >
> > General MIDI Level 2 Single Channel Melodic Profile
> >
> > General MIDI Level 2 Single Channel Drums Profile

In such cases, a single Profile specification document may define more than one Profile Id because there is high commonality of the messages used by the Profiles.

3.3 Profile Details Inquiry

If a Profile is a Multi-Channel Profile as defined in Section 2.3.4 and does not define a fixed number of Channels, then the Profile specification shall define that the use of the Profile Details Inquiry mechanism should be used to negotiate the number of Channels to be allocated (see Section 2.5.1).

A Profile specification may define the use of the Profile Details Inquiry with Registered Targets applicable to the Profile

A Profile specification may define optional, extended features that a Receiver may support. Then the Profile shall clearly define the Profile Specific Target(s) used and the format of the targeted data in a reply.

3.4 Property Exchange

A Profile specification may define Property Exchange Resources which are useful to realize the goals of the Profile. Those Resources may be common Resources which have already been defined for general use. Or the Profile may define new Resources which are uniquely defined to serve the purposes of the Profile.

3.5 Commonality of Profile Properties/Parameters

As much as practicably possible, Profile definitions should use the same message to control any function that is already defined elsewhere.

> For Example: GM2 already defines Reverb Level on CC#91. In most cases it is beneficial for another Profile that contains controls for reverb to use that same controller for its reverb level.

However, this is just a guideline for commonality across Profiles. It may not be reasonable or feasible to observe this guideline in all parameters of all Profile definitions.

3.6 Note On/Off with Attribute Type

A Profile may define the use of specific Attribute Types and the associated Attribute Data fields in Note On and Note Off messages.

When a Profile defines the use of Attribute Type 0x02 (Profile Specific), then the format of the Attribute Data fields for those Note On and Note Off messages shall be defined by the Profile specification. This definition of the Attribute Data is only known by devices which understand the Profile. Devices that do not understand the currently active Profile should ignore the Attribute Data when the Attribute Type is set to 0x02. See the M2-104-UM Universal MIDI Packet (UMP) Format and MIDI 2.0 Protocol specification ***[MA06]***, for more details of the Attribute Types and the associated Attribute Data fields.

3.6.1 Control Change (CC) Messages

MIDI 1.0 has defined default assignment of Control Change messages to specific parameters. Profiles shall observe those legacy assignments. A list of Control Change messages that Profiles should use if they have the associated parameter is attached as Appendix A. At the time of enabling a Profile, the device shall reset these Control Change destination assignments. A Profile might also define that certain parameters should or shall be reset to their default values. For a list of standardize default values see M2-113-UM MIDI-CI Profile: Default Control Change Mapping specification ***[MA07]***.

Refer to the latest MIDI specifications for any further updates to the list of Control Change Messages.

3.6.2 Registered Controllers (RPN)

For new control parameters that are not already assigned by MIDI 1.0 to specific Control Change messages, Profiles should assign the parameter to a Registered Controller (RPN). Profiles may also assign a Registered Controller for parameters normally addressable by a Control Change where increased functionality of the message is required (for example, for increased resolution or for relative control).

Assigning a function to a Registered Controller requires approval of the MMA and AMEI.

3.6.3 Manufacturer Specific Profiles and Assignable Controllers

Manufacturer Specific Profiles that define unique parameters that are not addressed by defined Control Change or Registered Controller (RPN) messages may be controlled by Assignable Controllers (NRPN) messages without specific approval.

3.7 Mode Messages

MIDI Mode Messages (CC#120-127) have special functionality. All Profiles shall define the response to Reset All Controllers (CC#121). A Profile for any device that uses Note-On/Off messages shall define the response to all of these, even if the response is the default MIDI 1.0 meaning. Some might be defined as ignored. Some Mode change messages might trigger All Notes Off while ignoring the rest of the default mechanisms (See General MIDI Level 2 as an example).

All Sound Off (CC#120)

Reset All Controllers (CC#121)

Local On/Off (CC#122)

All Notes Off (CC#123)

Omni Mode Off (CC#124)

Omni Mode On (CC#125)

Mono Mode On (CC#126)

Poly Mode On (CC#127)

3.8 Profile Categories

There are several common categories of Profiles. The category of a Profile is used to help organize or group similar Profiles into related sets. In particular, this is used to assign Registered Controllers (RPN) numbers to parameters defined by each Profile.

Profiles are organized into 3 categories:

- Feature Profiles or Ancillary Profiles (General MIDI, Show Control, MIDI Visual Control, Note Tuning, MPE, Sample Dump, Mixed Data Types)
- Instrument Profiles (Orchestral Strings, Piano, Organ, Drums)
- Effect Profiles (Reverb, Delay, Chorus, Overdrive, Rotary, etc.)

3.8.1 Feature Profiles

Some Feature Profiles define features and MIDI implementation requirements that apply across a wide range of musical instrument types. Examples might include Tuning Tables, Real Time Direct Pitch Control, Per Note Expression (like MPE), Layer/Zone Key Configuration, etc.

Some Feature Profiles define features and MIDI implementation requirements that apply to devices that record, edit, or modify MIDI data such as Arpeggiators, Sequencers or Rhythm/Music Style engines.

Some Feature Profiles define features and MIDI implementation requirements for devices that are not a musical instrument or an audio effects processor. Examples might include Lighting Control, Drone Flight Control, Video Effects Control, Mixer Control and Automation, DAW Software Control, etc.

3.8.2 Instrument Profiles

Instrument Profiles define features and MIDI implementation requirements for various musical instrument types or devices that use Note On/Off messages to represent musical notes. Typical Instrument Profiles might include Piano, Organ, Orchestral Strings, Guitar, Brass Instruments, Drums, Subtractive Synthesizer, FM Synthesizer, etc.

3.8.3 Effect Profiles

Effect Profiles define features and MIDI implementation requirements for various audio processor types. Typical Effect Profiles might include Reverb, Chorus, Compressor, Overdrive, Rotary, Delay, Audio Mixer, etc.

3.9 Protocol Differences

Profiles can be supported by devices using the MIDI 1.0 Protocol and/or the MIDI 2.0 Protocol. Some Profiles might be written for only one protocol. Profile specifications shall state which MIDI protocols are supported.

In the case of Profiles supporting multiple protocols, it is not mandated that performance must be equal for all protocols. MIDI 2.0 has some messages and features that do not have equivalent messages and features in MIDI 1.0. If Profile performance is not equal on all supported MIDI protocols, the Profile specification should clearly delineate the differences.

In some cases, a Profile might define a different mechanism in each protocol and define a unique translation of those mechanisms between different protocols when the mechanisms are not accomplished by equivalent messages. Standard Translator devices may not be able to perform that unique translation. But specialized translators might be available to support the unique features of such Profiles.

Notwithstanding the differences explained above, Profiles shall support the default assignment of Control Change messages as defined in Section 3.4.1. whether using MIDI 1.0 Protocol or MIDI 2.0 Protocol.

Appendix A: Defined Control Change Messages Common to All Profiles

A list of Control Change messages that Profiles should use if they define the use of the associated parameter. Profiles are not required to support all these messages.

Table 14 Control Changes and Mode Changes

Control Change Number			Control Function
Decimal	Binary	Hex	
0	00000000	00	Bank Select
1	00000001	01	Modulation Wheel or Lever
2	00000010	02	Breath Controller
4	00000100	04	Foot Controller
5	00000101	05	Portamento Time
6	00000110	06	Data Entry MSB
7	00000111	07	Channel Volume (formerly Main Volume)
8	00001000	08	Balance
10	00001010	0A	Pan
11	00001011	0B	Expression Controller
12	00001100	0C	Effect Control 1
13	00001101	0D	Effect Control 2
32-63	00100000	20	LSB for CC # 0-31
64	01000000	40	Damper Pedal on/off (Sustain)
65	01000001	41	Portamento On/Off
66	01000010	42	Sostenuto On/Off
67	01000011	43	Soft Pedal On/Off
68	01000100	44	Legato Footswitch
69	01000101	45	Hold 2
70	01000110	46	Sound Controller 1 (default: Sound Variation)
71	01000111	47	Sound Controller 2 (default: Timbre/Harmonic Intensity)
72	01001000	48	Sound Controller 3 (default: Release Time)
73	01001001	49	Sound Controller 4 (default: Attack Time)
74	01001010	4A	Sound Controller 5 (default: Brightness)
75	01001011	4B	Sound Controller 6 (default: Decay Time - see MMA RP-021)

76	01001100	4C	Sound Controller 7 (default: Vibrato Rate - see MMA RP-021)
77	01001101	4D	Sound Controller 8 (default: Vibrato Depth - see MMA RP-021)
78	01001110	4E	Sound Controller 9 (default: Vibrato Delay - see MMA RP-021)
79	01001111	4F	Sound Controller 10 (default undefined - see MMA RP-021)
84	01010100	54	Portamento Control
88	01011000	58	High Resolution Velocity Prefix
91	01011011	5B	Effects 1 Depth (default: Reverb Send Level - see MMA RP-023) (formerly External Effects Depth)
92	01011100	5C	Effects 2 Depth (formerly Tremolo Depth)
93	01011101	5D	Effects 3 Depth (default: Chorus Send Level - see MMA RP-023) (formerly Chorus Depth)
94	01011110	5E	Effects 4 Depth (formerly Celeste [Detune] Depth)
95	01011111	5F	Effects 5 Depth (formerly Phaser Depth)
96	01100000	60	Data Increment (Data Entry +1) (see MMA RP-018)
97	01100001	61	Data Decrement (Data Entry -1) (see MMA RP-018)
98	01100010	62	Non-Registered Parameter Number (NRPN) - LSB
99	01100011	63	Non-Registered Parameter Number (NRPN) - MSB
100	01100100	64	Registered Parameter Number (RPN) - LSB*
101	01100101	65	Registered Parameter Number (RPN) - MSB*
Controller numbers 120-127 are reserved for Channel Mode Messages, which rather than controlling sound parameters, affect the Channel's operating mode.			
120	01111000	78	[Channel Mode Message] All Sound Off
121	01111001	79	[Channel Mode Message] Reset All Controllers (See MMA RP-015)
122	01111010	7A	[Channel Mode Message] Local Control On/Off
123	01111011	7B	[Channel Mode Message] All Notes Off
124	01111100	7C	[Channel Mode Message] Omni Mode Off (+ all notes off)
125	01111101	7D	[Channel Mode Message] Omni Mode On (+ all notes off)
126	01111110	7E	[Channel Mode Message] Mono Mode On (+ poly off, + all notes off)
127	01111111	7F	[Channel Mode Message] Poly Mode On (+ mono off, +all notes off)

http://www.amei.or.jp

https://www.midi.org