General MIDI Lite

And Guidelines for Use In Mobile Applications

Version 1.0
October 5, 2001

Published By:
The MIDI Manufacturers Association
Los Angeles, CA

NOTE: GML Specification Version 1.0 (the release version of GML) includes a change to the recommended MIDI Pan formula from previous (unpublished) versions of the document.

General MIDI Lite Specification and Guidelines for Use In Mobile Applications
RP-033

Copyright ©2001 MIDI Manufacturers Association Incorporated

ALL RIGHTS RESERVED. NO PART OF THIS DOCUMENT MAY BE REPRODUCED OR TRANSMITTED IN ANY FORM OR BY ANY MEANS, ELECTRONIC OR MECHANICAL, INCLUDING INFORMATION STORAGE AND RETRIEVAL SYSTEMS, WITHOUT PERMISSION IN WRITING FROM THE MIDI MANUFACTURERS ASSOCIATION.

Printed 2001

MMA
PO Box 3173
La Habra CA 90632-3173

TABLE OF CONTENTS

(c) 2001 MIDI Manufacturers Association Incorporated All Rights Reserved

(c) 2001 MIDI Manufacturers Association Incorporated All Rights Reserved

1 INTRODUCTION

1.1 Background

The MIDI protocol was originally designed with the assumption that the Transmitter and Receiver are located and controlled together in a single location. This is not the case within the portable device market area, which includes products such as cellular telephones with Ringing Tone Melodies. In this market area, the Transmitter and Receiver may be widely separated and the Receiver (e.g. cell phone) may have only limited control over the Transmitter (e.g. remote music server). Under these circumstances it can be particularly difficult to ensure compatibility, so it is essential that standards exist for song data content and playback performance.

As the administrators of the MIDI Standard, AMEI and the MMA are obligated to promote the correct application of MIDI, SMF (Standard MIDI File) and related specifications, and to establish standards and recommended practices appropriate for such markets. This document describes a platform for mobile MIDI communication that has been approved and adopted as a standard by the MIDI industry (represented by AMEI and MMA). Additional specifications are also planned for development by AMEI and MMA in the near future.

1.2 About this Document

This document has three primary components:

- A specification called **General MIDI Lite (GM Lite)**, which defines a new level of tone generation (sound module) device;
- Authoring guidelines for music data in SMF (Standard MIDI File) format that is intended for playback on GM Lite devices;
- Implementation guidelines for GM Lite file players .

This document was initially developed by the Mobile MIDI working group of the Association of Musical Electronics Industry (“AMEI”), to address the cellular telephone ringer market in Japan. The document was subsequently refined by the MIDI Manufacturers Association (“MMA”) and AMEI to enable broader use of General MIDI Lite in mobile communications and in other portable applications.

Note: The General MIDI Lite specification defines a fixed-polyphony MIDI device, intended to meet a particular set of current and future market needs. AMEI and MMA are well aware that this fixed-polyphony approach poses a number of problems, not only for current software-based Tone Generators with lower polyphony, but also for future higher-polyphony devices. Therefore, AMEI and the MMA are also developing a specification for flexible polyphony MIDI devices and content, that is intended to complement General MIDI Lite. Developers of GM Lite players are strongly advised to keep as much flexibility as possible in how their players handle channel priorities, drum channels and other System messages. This will make it far easier for their products to be compatible with song data authored for the pending scalable MIDI specification.

1.3 About General MIDI Lite

1. General MIDI Lite Sound is intended for equipment that does not have the capability to support GM (General MIDI) 1.0, on the assumption that a lower level of performance may be acceptable in some applications. GM Lite represents just one standardized set of performance capabilities for portable applications — other performance levels are likely to be standardized in the future.
2. It is necessary to comply with all three parts of this document (General MIDI Lite Sound Module Specification, Authoring Guidelines For GM Lite Content and Player Guidelines for GM Lite Content) to ensure compatibility and proper operation of a GM Lite device.

(c) 2001 MIDI Manufacturers Association Incorporated All Rights Reserved

3. Features that are optional but recommended are indicated as [*recommended*]. All features without such indication must be considered to be [*required*].
4. GM Lite playback systems must assume the SMF format, which is hereafter approved as the official GM Lite input file format by AMEI / MMA.
5. The Authoring Guidelines for GM Lite Content are based on the assumption that the song content data is produced using currently available commercial MIDI sequencing software.

1.4 Guide To Technical Terms

- **MIDI**
  MIDI (Musical Instrument Digital Interface) is essentially standardized as a communications protocol for the transmission of music as electronic data and for communication between a transmitter and receiver using MIDI messages. Originally, MIDI was designed as the communication protocol for the 5 pin DIN MIDI cable, however, MIDI is often used in other environments and with other transports. The MIDI protocol has been enhanced and expanded in many ways, and the MIDI specification now includes sub-specifications and recommended practices such as:

  - Standard MIDI Files
  - General MIDI
  - MIDI Show Control
  - MIDI Time Code
  - MIDI Machine Control
  - Downloadable Sounds

- **AMEI**
  JMSC (Japan MIDI Standard Committee was formed in 1983. It later became AMEI (Association of Musical Electronics Industry) and was authorized in 1996 as a corporation by The Minister of International Trade and Industry (Japan) with the goal to contribute to the growth of the Japanese national economy and to improve the life of the people by promoting the development of the musical electronics industry and related industries with such activities as the production, distribution and researching of products for the musical electronics industry; accumulating and presenting information; and forming plans and promoting MIDI as a standard.
  http://www.amei.or.jp/index.html

- **MMA**
  The MMA (MIDI Manufacturers Association) is a non-profit corporation formed to maintain the MIDI specification as an open standard. The MMA provides forums for discussion of proposals aimed at improving and standardizing the capabilities and marketability of MIDI-related products, and provides a process for adoption and subsequent publication of any enhancements or clarifications resulting from these activities. The MMA members include large and small companies from every application of audio and MIDI technology, including stage and theater, musical performance and recording, computing, telecommunications, film and broadcast. Membership is open to anyone commercially involved in the design or manufacture of MIDI hardware or software.
  http://www.midi.org/

- **RP**
  AMEI/MMA RP (Recommended Practice) documents supplement the MIDI specification by providing references for the standardized use of MIDI for or in specific applications. RPs are defined by AMEI and MMA to extend the capabilities of MIDI and to ensure interoperability and correct performance of MIDI applications and devices. SMF (Standard MIDI File) and General MIDI are two examples of RPs.

(c) 2001 MIDI Manufacturers Association Incorporated All Rights Reserved

- **SMF**
  The SMF (Standard MIDI File) format was created in 1988 for exchanging song data among MIDI devices and software programs. SMF stores sequenced performance data with time stamps, much like a musical score. Since the specific instrumentation is not included in the SMF data, SMF files alone do not guarantee identical performance on different sound sources.

- **GM**
  GM (General MIDI) was introduced in 1991 to improve sound module compatibility with Standards MIDI File data. GM's functions were enhanced with GM 2 (General MIDI 2) by AMEI and MMA in 1999.

- **Music Content**
  Music Content means music for any type of portable devices such as ringing tone with melodies, newly released music, game music or BGM (background music) for websites. It is often stored in a downloadable site or in the device itself.

- **MIDI Sound Module and Keyboard Controller**
  A MIDI Sound Module (receiver) and Keyboard Controller (transmitter) provide sound by transmitting and receiving MIDI messages according to the MIDI protocol.

- **Velocity**
  Velocity refers to how slow or fast a piano key moves when struck (or released), due to the intensity with which it is struck (or released). A high velocity is achieved by a heavy (pounding) touch, while a lower velocity is achieved by a lighter (and more legato) touch. In MIDI, the velocity message is used to determine the appropriate sound level and timbral makeup of each note that is played, just as it would in a piano.

- **Same Key Number on Same Channel**
  For every note in MIDI there is a Note Off message which effectively determines the duration of the note. "Same Key Number on Same Channel" refers to the transmission of duplicated Note-On messages without a Note Off message between them — a situation that would never actually occur in a piano, for example, and for which there is no defined response in MIDI. To prevent incompatibility among GM Lite devices, this practice is prohibited in GM Lite.

- **MIDI Sequencer**
  A software or hardware system which include facilities to perform input, edit or/and playback of MIDI Data (which is played back "sequentially"). Most MIDI sequencers can store files in SMF format.

- **Bar (measure)**
  An integer quantity of consecutive "beats" (the numerator of the time signature), each of the length specified by the denominator of the time signature. The starting time in a MIDI sequence is often represented by [Bar Beat Tick], where Bar numbers and beat numbers are 1-based, but Tick numbers are 0-based.

- **Chase**
  "Chase" signifies to process by transmitting sequence data to the aimed point. No tone generation is made during the chase, Note Messages should not be transmitted. When there are multiple messages of the same type (e.g. volume and expression) in the interval of the chase, only the last message should be transmitted.

- **Note Stealing**
  If there is an unused sound generator, a new note on message on any MIDI channel will use it. If not, a still-sounding note may have to be silenced in order to re-use its sound generator.

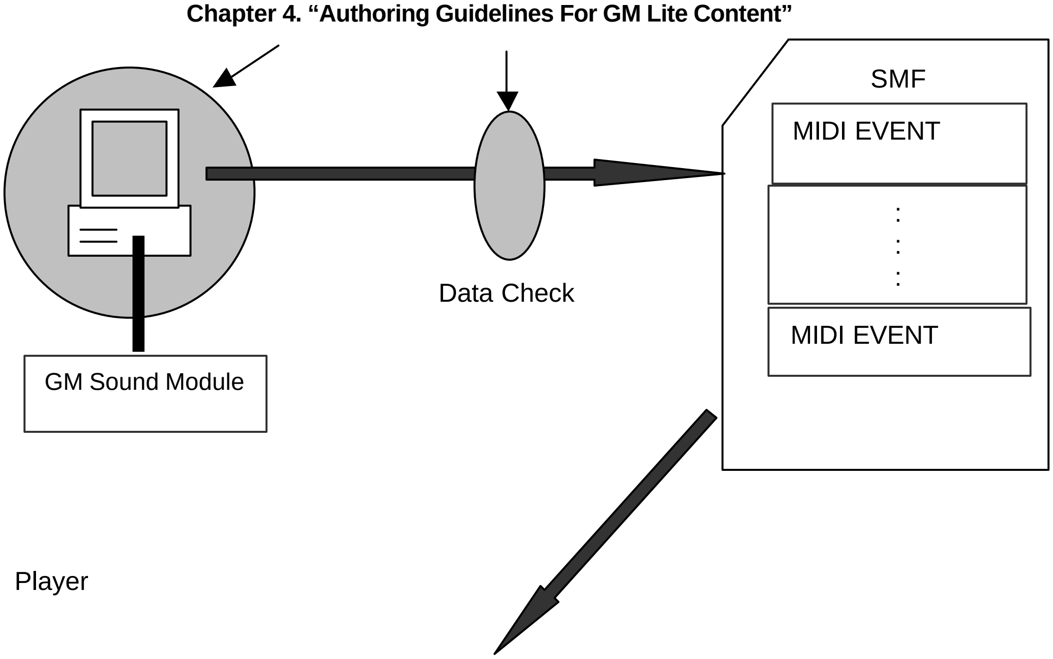
(c) 2001 MIDI Manufacturers Association Incorporated All Rights Reserved

2 MOBILE MIDI OVERVIEW

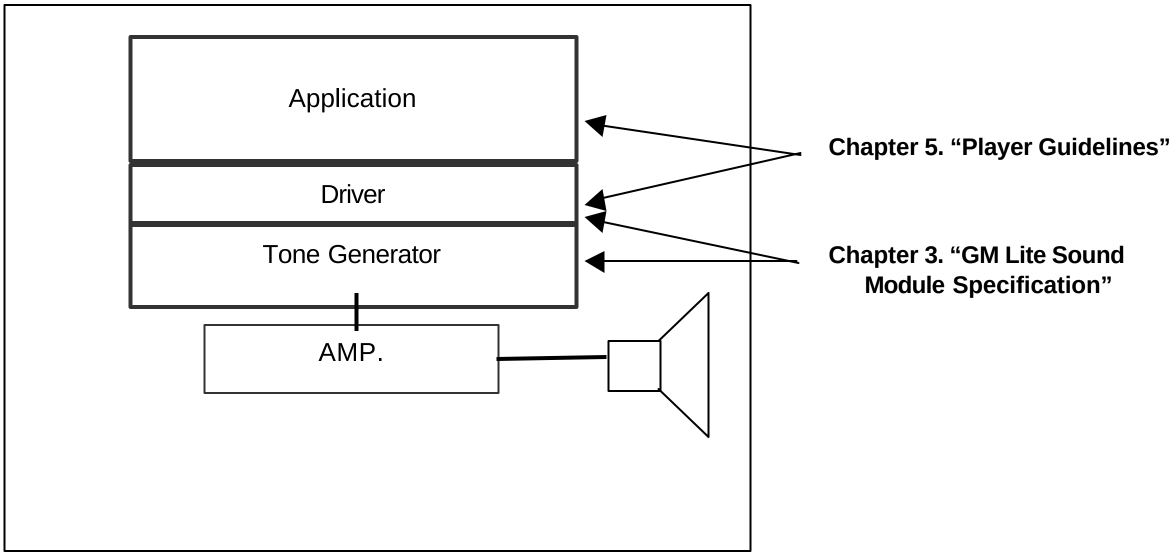
2.1 Functional Model Map

The Functional Model and the relation between each section is shown in the diagram below.

1) Music Contents



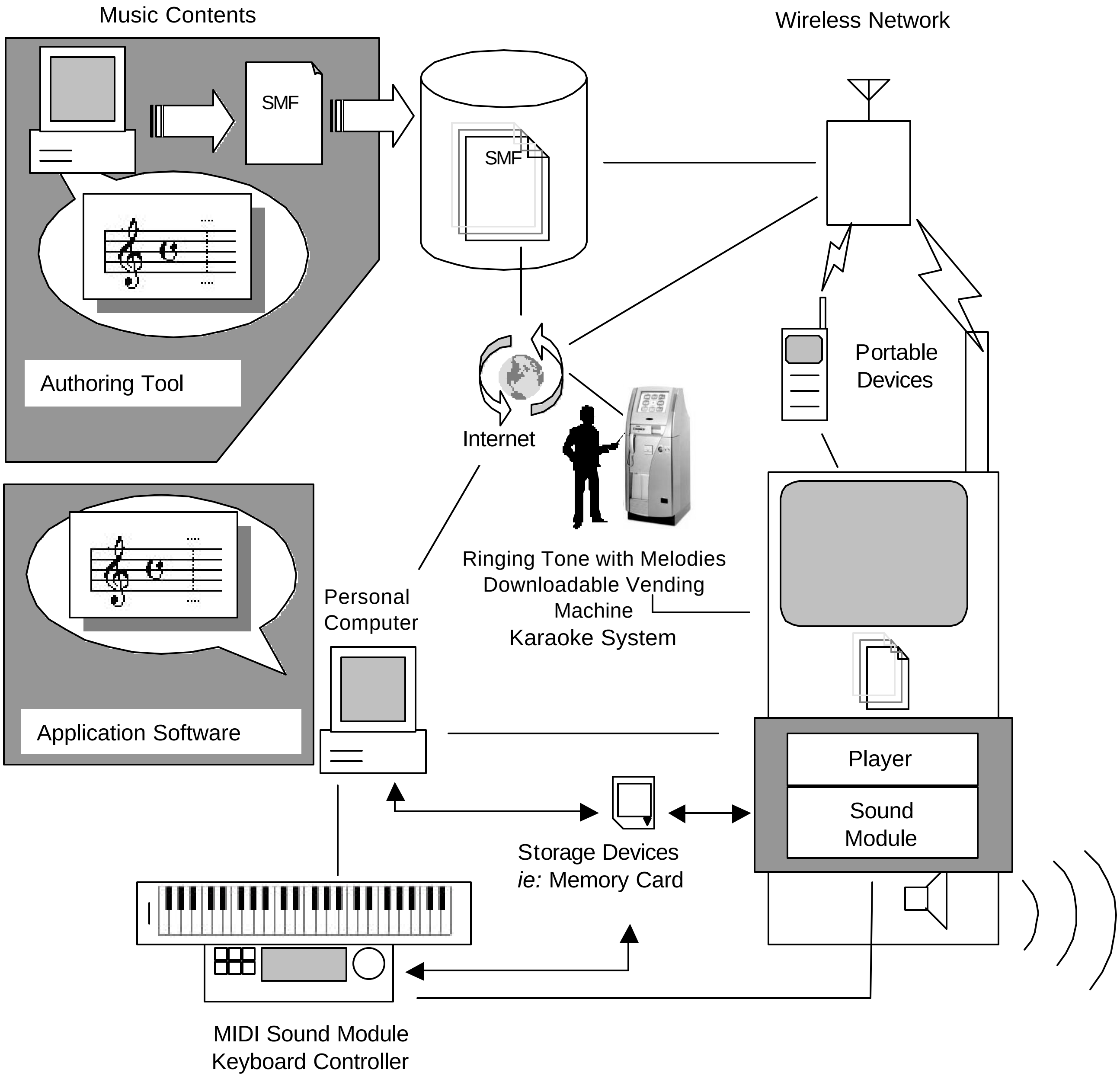
2) Player


Application
Driver
Tone Generator
AMP.
Chapter 5. "Player Guidelines"
Chapter 3. "GM Lite Sound Module Specification"

(c) 2001 MIDI Manufacturers Association Incorporated All Rights Reserved

2.2 Service Model Functional Map

The screened part ▨ indicates the applicable range of this document.



(c) 2001 MIDI Manufacturers Association Incorporated All Rights Reserved

2.3 Future Enhancements

Additional functions useful in mobile applications such as content security, identification of copyrights, and simultaneous playback of picture and text will be considered for standardization by AMEI/MMA in the future. These functions are not covered in this document.

(c) 2001 MIDI Manufacturers Association Incorporated All Rights Reserved

3 GENERAL MIDI LITE SOUND MODULE SPECIFICATION

3.1 General Requirements

3.1.1 Sound Source Type

Undefined.

Each manufacture can choose the most appropriate technology, as long as it meets the requirements of this document.

3.1.2 Number of Notes (Polyphony)

The sound engine must be capable of generating 16 notes simultaneously, allocated between any desired combination of Melody and Rhythm Channels. The Rhythm Channel must be capable of generating up to 8 simultaneous notes. Therefore, the sound engine must be capable of generating 8 Melody Channel and 8 Rhythm Channel notes, 16 Melody Channel notes, or any combination in between (subject to the Rhythm Channel polyphony limit).

3.1.3 MIDI Channels

All 16 MIDI channels must be addressable simultaneously.

[Note] MIDI channels in consumer documentation are normally one-based, although they are stored in the MIDI status byte as zero-based.

3.1.4 Melody Channels and Rhythm Channels

Channels 1 to 9 and 11 to 16 are Melody Channels (used for melody, accompaniment, bass and other pitched instrumental parts). A Melody Channel may select timbres or sounds from the Melody Channel Sound Set (3.4.1).

Channel 10 is the Rhythm Channel (used for percussion and drum parts).
The Rhythm Channel may select timbres from the Rhythm Channel Sound Set (3.4.2).

3.1.5 Modes

The initial mode for all MIDI Channels is Mode 3 (OMNI OFF, POLY). Each Channel can provide variable polyphony, within the limits established for that channel by the instrument (and this specification). When the number of available sound generators is exceeded, the module's voice assignment algorithm will affect the overall sound. See section 3.1.8 for details.

3.1.6 Timbres

All timbres described in both the Melody Channel Sound Set (3.4.1) and the Rhythm Channel Sound Set (3.4.2) must be provided.

[Note] It is recommended, but not required, that each of the approximately 175 timbres should be (a) audibly distinct from other timbres in these two sound sets, and (b) reasonably similar to timbres provided by common GM 1 synthesizers. However, a compliant instrument may employ a smaller set of distinct timbres, as long as such a reduced timbre set is mapped appropriately to the full set of Program Change and Note Numbers specified for the Melody Channel Sound Set and Rhythm Channel Sound Set.

3.1.7 Pitch

3.1.7.1 Melody Channels (Tuned Instruments)

3.1.7.1.1 Pitched Instrument Sounds

Middle C Note = Note number 60 (3CH).
A above Middle C, Note number 69 (45H) is tuned to 440Hz (when pitch bend is set to center).

3.1.7.1.2 Effect Sounds

On MIDI Channels 1-9 and 11-16, programs 116-128, pitch and temperament are undefined.

3.1.7.2 Rhythm Channel (Drum and Percussion Instruments)

On MIDI Channel 10, a specific rhythm timbre is assigned to each note number.
(See 3.4.2 Rhythm Channel Sound Set)

3.1.8 Sound Generator Assignment

If there is an unused sound generator, a new note on message on any MIDI channel will use it. “Note stealing” will occur if there are no unused sound generators.

Manufacturers can decide how to assign sound generators for the following situations:

1. When a new Note On is received while all the sound generators of the sound module are already in use.
2. When one note number has multiple generators, which are active simultaneously.

If no unused sound generators are available and a Channel-based generator assignment algorithm is used, the following Channel priority scheme should be used:
(highest priority) 10ch. > 1ch. > 2ch. >… > 9ch. >11ch. >...>16ch *(lowest priority)*.

3.1.8.1 Rhythm Channels

Note Off messages on the Rhythm Channel are ignored. All sounds in the Rhythm Channel Sound Set (3.4.2) are fixed-duration (but sounds which are mutually exclusive may be cut short).

[Required] The three Hi-hat sounds (Note numbers 42/44/46) must use mutually exclusive assignment. For example, if a Note On message with key number 42 is received while Note number 46 is sounding, Note number 46 must be promptly muted and Note number 42 should then sound.

[Recommended] The following note pair combinations should use the same type of mutually-exclusive assignment as the Hi-Hat:

Note number 71 (Short Whistle) and Note number 72 (Long Whistle)
Note number 73 (Short Guiro) and Note number 74 (Long Guiro)
Note number 78 (Mute Cuica) and Note number 79 (Open Cuica)
Note number 80 (Mute Triangle) and Note number 81 (Open Triangle)

(c) 2001 MIDI Manufacturers Association Incorporated All Rights Reserved

3.2 MIDI Channel Messages

3.2.1 Note On / Note Off

All notes (all sounds/timbres) shall respond to velocity.

3.2.2 Program Change Message

Default Value: 1 (00H)

Sets the timbre for the specified Channel, except Channel 10.

[Note] Program Change messages in consumer documentation are normally one-based; therefore, the decimal value of 1 presented here as the default is equivalent to 00H.

3.2.3 Control Change Messages

3.2.3.1 CC#1: Modulation Depth

Default Value: 0 (00H)

Changes the vibrato (LFO pitch modulation) depth of the specified Channel. The waveform of the LFO shall be a triangle wave or a sine wave. The depth of change at the maximum value conforms to the value, which is set by +/- 50 cents as the standard, however it is possible that different timbres will have differing depths.

3.2.3.2 CC#7: Channel Volume

Default Value: 100 (64H)

Changes the volume of all sounds on the specified MIDI Channel and thus the relative volume balance among the Channels.

The following formula is required:

$$\text{Gain[dB]} = 20*\log\ (\ (\text{CC\#7})\ ^{2}/127^{2})$$

3.2.3.3 CC#10: Pan

Default Value: 64 (40H)

Sets the stereo position for notes of the specified Melody Channel. This message causes subsequent notes on that Melody Channel to be positioned anywhere in the stereo field from hard left (value 0) to hard right (value 127).

[Note] 64 (40H) is Center. (Since MIDI controller values range from 0 to 127, the exact center of the range, 63.5, cannot be represented. Therefore, the effective range for CC#10 is modified to be 1 to 127, and values 0 and 1 both pan hard left.)

It is not necessary to pan a Note that is currently sounding. However, if a currently sounding Note is panned, the panning shall be done without audible artifacts, clicks or "zipper" noise.

The following formulas are recommended:

If CC#10 > 0, subtract 1 from CC#10 prior to applying the gain formulas.
Left Channel Gain [dB] = 20*log (cos (Pi /2* (CC#10)/126)
Right Channel Gain [dB] = 20*log (sin (Pi /2* (CC#10)/126)

(c) 2001 MIDI Manufacturers Association Incorporated All Rights Reserved

CC#11: Expression

Default Value: 127 (7FH)

Modifies the volume set by Channel Volume on the specified Channel.

[Note] CC#7 and CC#11 are used for different purposes. Channel Volume (CC#7) should be used to set the overall volume of the Channel prior to music data playback as well as for mixdown fader-style movements, while Expression (CC#11) should be used during music data playback to attenuate the programmed MIDI volume (CC#7) data.

The following formula is required:

$$\text{Gain [dB]} = 20*\log\left((\text{CC\#11})^{2}/127^{2}\right)$$

3.2.3.5 CC#6/38: Data Entry

Default Value: 0/0 (00H/00H)

Data Entry (MSB/LSB) is used on the specified Channel to adjust the value of the RPN that is selected using CC#100/101.

3.2.3.6 CC#64: Hold 1 (Damper)

Default Value: 0 (00H)

Turns Damper ON or OFF for the specified Channel. (Also known as "sustain pedal") Damper values between 0 and 63 are recognized as OFF, and values between 64 and 127 are recognized as ON.

3.2.3.7 CC#100/101: RPN LSB / MSB

Default Value: 127/127 (7FH/7FH) NULL

Selects parameter numbers for the RPN on the specified Channel. Send MSB followed by LSB.

3.2.4 RPN (Registered Parameter Numbers)

00H/00H: Pitch Bend Sensitivity

Default Value: 2/0 (02H/00H)

Sets the sensitivity of Pitch Bend. The MSB of Data Entry represents the sensitivity in semitones and the LSB of Data Entry represents the sensitivity in cents. LSB must be set to 0.
For example, a value of MSB=01, LSB=00 means +/- 1 semitone (a total range of 2 semitones).
The device shall be able to accommodate at least +/- 12 semitones.

3.2.5 Channel Mode Messages

3.2.5.1 CC#120: All Sound Off

Value: 0 (00H)

Mute all sound generated on the specified Channel.

(c) 2001 MIDI Manufacturers Association Incorporated All Rights Reserved

3.2.5.2 CC#121: Reset All Controllers

Value: 0 (00H)

When value is 00H, this message resets the status of controllers and other messages in the table below on the specific Channel as follows:

Controller	Message	Value	Comment
1	Modulation	00H	OFF
11	Expression	7FH	MAX
64	Hold 1	00H	OFF
100	RPN LSB	7FH	NULL
101	RPN MSB	7FH	NULL
	Pitch Bend Change	40H/00H	Center

The Program Change, the Channel Volume and the Pan are NOT reset.

3.2.5.3 CC#123: All Notes Off

Value: 0 (00H)

Turns off all Notes sounding on the specified Channel.

3.2.6 Pitch Bend Change

Default Value: Center (40H/00H)

Adjusts the Pitch up or down on the specified Channel. Default sensitivity (range) is +/- 2 semitones. 00H/00H specifies maximum pitch bend down. 7FH/7FH specifies maximum pitch bend up. Pitch Bend Sensitivity can be adjusted using RPN 00H/00H.

3.3 System Messages

3.3.1 GM Lite Reset

	Description
F0H 7EH	Universal Non-Real Time System Exclusive header
<device ID>	ID of target device (7FH:ALL)
09H	sub ID number #1=GM Message
01H	sub ID number #2=GM1 On
F7H	End of Exclusive

The “GM1 System On” message is used for resetting a GM Lite sound module.
When this message is received, all currently sounding Notes immediately mute without producing a click, and the device is reset to the default values specified in this chapter.

The reset operation shall be completed within 100 milliseconds after receiving GM1 System On.

(c) 2001 MIDI Manufacturers Association Incorporated All Rights Reserved

3.4 Sound Sets

3.4.1 Melody Channel Sound Set

PC#	Timbre	Recommended Key Range
1(00H)	Acoustic Grand Piano	21-108
2(01H)	Bright Acoustic Piano	21-108
3(02H)	Electric Grand Piano	21-108
4(03H)	Honky-tonk Piano	21-108
5(04H)	Electric Piano1	28-103
6(05H)	Electric Piano 2	28-103
7(06H)	Harpsichord	41-89
8(07H)	Clavi	36-96
9(08H)	Celesta	60-108
10(09H)	Glockenspiel	72-108
11(0AH)	Music Box	60-84
12(0BH)	Vibraphone	53-89
13(0CH)	Marimba	48-84
14(0DH)	Xylophone	65-96
15(0EH)	Tubular Bells	60-77
16(0FH)	Dulcimer	60-84
17(10H)	Drawbar Organ	36-96
18(11H)	Percussive Organ	36-96
19(12H)	Rock Organ	36-96
20(13H)	Church Organ	21-108
21(14H)	Reed Organ	36-96
22(15H)	Accordion	53-89
23(16H)	Harmonica	60-84
24(17H)	Tango Accordion	53-89
25(18H)	Acoustic Guitar (nylon)	40-84
26(19H)	Acoustic Guitar (steel)	40-84
27(1AH)	Electric Guitar (jazz)	40-86
28(1BH)	Electric Guitar (clean)	40-86
29(1CH)	Electric Guitar (muted)	40-86
30(1DH)	Overdriven Guitar	40-86
31(1EH)	Distortion Guitar	40-86
32(1FH)	Guitar Harmonics	40-86
33(20H)	Acoustic Bass	28-55
34(21H)	Electric Bass (finger)	28-55
35(22H)	Electric Bass (pick)	28-55
36(23H)	Fretless Bass	28-55
37(24H)	Slap Bass 1	28-55
38(25H)	Slap Bass 2	28-55
39(26H)	Synth Bass 1	28-55
40(27H)	Synth Bass 2	28-55
41(28H)	Violin	55-96
42(29H)	Viola	48-84
43(2AH)	Cello	36-72
44(2BH)	Contrabass	28-55
45(2CH)	Tremolo Strings	28-96
46(2DH)	Pizzicato Strings	28-96
47(2EH)	Orchestral Harp	23-103
48(2FH)	Timpani	36-57
49(30H)	String Ensembles 1	28-96
50(31H)	String Ensembles 2	28-96
51(32H)	Synth Strings 1	36-96
52(33H)	Synth Strings 2	36-96
53(34H)	Choir Aahs	48-79
54(35H)	Voice Oohs	48-79
55(36H)	Synth Voice	48-84
56(37H)	Orchestra Hit	48-72
57(38H)	Trumpet	58-94
58(39H)	Trombone	34-75
59(3AH)	Tuba	29-55
60(3BH)	Muted Trumpet	58-82
61(3CH)	French Horn	41-77
62(3DH)	Brass Section	36-96
63(3EH)	Synth Brass 1	36-96
64(3FH)	Synth Brass 2	36-96

PC# = Program Change Number

(c) 2001 MIDI Manufacturers Association Incorporated All Rights Reserved

PC#	Timbre	Recommended Key Range	PC#	Timbre	Recommended Key Range
65(40H)	Soprano Sax	54-87	97(60H)	FX 1 (rain)	36-96
66(41H)	Alto Sax	49-80	98(61H)	FX 2 (soundtrack)	36-96
67(42H)	Tenor Sax	42-75	99(62H)	FX 3 (crystal)	36-96
68(43H)	Baritone Sax	37-68	100(63H)	FX 4 (atmosphere)	36-96
69(44H)	Oboe	58-91	101(64H)	FX 5 (brightness)	36-96
70(45H)	English Horn	52-81	102(65H)	FX 6 (goblins)	36-96
71(46H)	Bassoon	34-72	103(66H)	FX 7 (echoes)	36-96
72(47H)	Clarinet	50-91	104(67H)	FX 8 (sci-fi)	36-96
73(48H)	Piccolo	74-108	105(68H)	Sitar	48-77
74(49H)	Flute	60-96	106(69H)	Banjo	48-84
75(4AH)	Recorder	60-96	107(6AH)	Shamisen	50-79
76(4BH)	Pan Flute	60-96	108(6BH)	Koto	55-84
77(4CH)	Blown Bottle	60-96	109(6CH)	Kalimba	48-79
78(4DH)	Shakuhachi	55-84	110(6DH)	Bag pipe	36-77
79(4EH)	Whistle	60-96	111(6EH)	Fiddle	55-96
80(4FH)	Ocarina	60-84	112(6FH)	Shanai	48-72
81(50H)	Lead 1 (square)	21-108	113(70H)	Tinkle Bell	72-84
82(51H)	Lead 2 (sawtooth)	21-108	114(71H)	Agogo	60-72
83(52H)	Lead 3 (calliope)	36-96	115(72H)	Steel Drums	52-76
84(53H)	Lead 4 (chiff)	36-96	116(73H)	Woodblock	*
85(54H)	Lead 5 (charang)	36-96	117(74H)	Taiko Drum	*
86(55H)	Lead 6 (voice)	36-96	118(75H)	Melodic Tom	*
87(56H)	Lead 7 (fifths)	36-96	119(76H)	Synth Drum	*
88(57H)	Lead 8 (bass + lead)	21-108	120(77H)	Reverse Cymbal	*
89(58H)	Pad 1 (new age)	36-96	121(78H)	Guitar Fret Noise	*
90(59H)	Pad 2 (warm)	36-96	122(79H)	Breath Noise	*
91(5AH)	Pad 3 (polysynth)	36-96	123(7AH)	Seashore	*
92(5BH)	Pad 4 (choir)	36-96	124(7BH)	Bird Tweet	*
93(5CH)	Pad 5 (bowed)	36-96	125(7CH)	Telephone Ring	*
94(5DH)	Pad 6 (metallic)	36-96	126(7DH)	Helicopter	*
95(5EH)	Pad 7 (halo)	36-96	127(7EH)	Applause	*
96(5FH)	Pad 8 (sweep)	36-96	128(7FH)	Gunshot	*

PC# = Program Change Number

3.4.2 Rhythm Channel Sound Set

Note	Timbre	Pan
24		
25		
26		
27		
28		
29		
30		
31		
32		
33		
34		
35	Acoustic Bass Drum	64
36	Bass Drum 1	64
37	Side Stick	64
38	Acoustic Snare	64
39	Hand Clap	54
40	Electric Snare	64
41	Low Floor Tom	34
42	Closed Hi-hat [EXC1]	84
43	High Floor Tom	46
44	Pedal Hi-hat [EXC1]	84
45	Low Tom	58
46	Open Hi-hat [EXC1]	84
47	Low-Mid Tom	70
48	High Mid Tom	82
49	Crash Cymbal 1	84
50	High Tom	94
51	Ride Cymbal 1	44
52	Chinese Cymbal	44
53	Ride Bell	44
54	Tambourine	74
55	Splash Cymbal	54
56	Cowbell	84
57	Crash Cymbal 2	44
58	Vibra-slap	29
59	Ride Cymbal 2	44
60	High Bongo	99
61	Low Bongo	99
62	Mute Hi Conga	39
63	Open Hi Conga	39
64	Low Conga	44
65	High Timbale	84
66	Low Timbale	84
67	High Agogo	29
68	Low Agogo	29
69	Cabasa	29
70	Maracas	24
71	Short Whistle [EXC2]	99
72	Long Whistle [EXC2]	99
73	Short Guiro [EXC3]	94
74	Long Guiro [EXC3]	94
75	Claves	84
76	Hi Wood Block	99
77	Low Wood Block	99
78	Mute Cuica [EXC4]	44
79	Open Cuica [EXC4]	44
80	Mute Triangle [EXC5]	24
81	Open Triangle [EXC5]	24
82		
83		
84		
85		
86		
87		

[Note] EXC means mutually exclusive. Instruments that have the same EXC numbers do not sound simultaneously. Playing one such sound will immediately shut off any other sound with the same EXC number. See section 3.1.8.1.

4 AUTHORING GUIDELINES FOR GM LITE CONTENT

4.1 Restrictions on GM Lite Song Data Content Production

GM Lite Song Data Content may be prepared by using any commercially available MIDI sequencer that conforms to the MIDI standard. The completed data must be saved as a Format 0 Standard MIDI File data image. (A Format 1 SMF cannot be played until the entire SMF data image is received, whereas a Format 0 SMF can be played while the data is downloading.)

4.1.1 Number of Notes (Polyphony)

GM Lite Song Data should be developed for a target tone generator capable of producing 16 simultaneous notes on up to 16 channels (Multiple notes can be used on each channel as long as the maximum number of simultaneous notes does not exceed 16. On the Rhythm Channel, up to 8 simultaneous notes may be specified in the song data.).

[Note] The expected order of note priority for the GM Lite Sound Module, from highest to lowest priority, is MIDI Channel: 10ch. > 1ch. > 2ch. >… > 9ch. >11ch. >...>16ch. (10ch.is fixed as a Rhythm Channel)

Because the Rhythm Channel has the highest priority for note generation, the content author should be aware that, when a variety of percussion notes are used, it is possible that concurrent notes on other channels may not be generated when the song data is processed by a GM Lite Player. Therefore, data on Channel 10 should be authored with polyphony limitations taken into account.

4.1.2 Channel Assignment

Three specific channel assignments are recommended for use in GM Lite Content. Other channels may be used at the discretion of the content author. The recommended channel assignments are as follows:

[*Recommended*]

Ch.	Recommended Part
1	Main-Melody
2	Sub-Melody
3	Bass

4.1.3 Available Pitch Range

See 3.4 Sound Sets for details.

4.1.4 Available Message Types

When preparing existing MIDI SMF data for use as GM Lite Content, the content author must delete any messages (events) that are not supported by the GM Lite specification. This will lower the processing overhead on the player.

Messages which are supported by GM Lite are listed in section 3.2 Response to MIDI Channel Messages and section 3.3 System Messages.

(c) 2001 MIDI Manufacturers Association Incorporated All Rights Reserved

4.1.5 Prohibitions on Certain Messages

(a) Do not use two or more simultaneous notes with the same key number on the same channel. This would create the potential for incorrect playback operation by the sound module, since the module could not correctly match Note Ons and Note Offs for the overlapping notes.

(b) Do not use the LSB value for Pitch-Bend-Sensitivity. The LSB value should always be set to zero.

4.1.6 Available Meta-Events

[required]
Time Signature, Tempo, and End Of Track meta-events must be supported.

[recommended]
All SMF meta-events are available.

4.1.7 Set-Up Bar

The first measure of the sequence shall be used as the Set-Up Bar. The Set-Up Bar is required for all MIDI sequences that comply with this RP. It must be present in order for portable devices to recognize the sequence as a valid GM Lite sequence.
The Set-Up Bar must adhere to the following constraints:

- Time Signature = 1/4 (meta-event)
- Tempo=240 (meta-event)
- GM1 System On system exclusive message must be present at time index 1 1 000.
- MIDI Note data must not be present in the Set-Up Bar. Careful treatment is necessary to ensure this constraint is met if the music starts with an upbeat or grace note.
- Program Change and Control Change messages may be present. If present, multiple messages should not be sent at the same time index. Also, they must be delayed by at least 125 milliseconds (1/8 note at 240 BPM) from the GM1 System On message.

The Time Signature and Tempo of the music must be indicated at the start of measure 2, even if the music continues with Time Signature 1/4 and Tempo 240.

When playback is repeated, the Set-Up Bar should not be repeated. IF REPEATED PLAYBACK IS DESIRED when the end of the sequence is reached, the player should immediately repeat playback by a) first "chasing" the song data in the Set-Up Bar at high speed, and then b) resuming normal playback beginning at the start of the second bar, time index 2 1 000. (Note: the GM1 System On message should ONLY be sent before the first time the song data is played. It should not be sent if playback is repeated. See 5.3.3 Dealing with the Set-Up Bar for further discussion.)

[*required*] No special order is observed

- Set-Up / Beat
  Time Index Data (Hex)
  1 1 000 FF 58 04 01 02 24 08 Meta-event: Beat =1/4
- Set-Up / Set Tempo
  Time Index Data (Hex)
  1 1 000 FF 51 03 03 D0 90 Meta-event: Set Tempo (Tempo)=240
- Set-Up / System Exclusive
  Time Index Data (Hex)
  1 1 000 F0 7E 7F 09 01 F7 GM1 System On System Exclusive

[*recommended*] Example of program change and control change messages sent in the Set-Up Bar.

Time Index	MIDI Msg./Meta Event	Set-Up Value	Data Range
1 1 000	Time Signature = 1/4	FF 58 04 01 02 24 08	---
1 1 000	Tempo = 240	FF 51 03 03 D0 90	---
1 1 000	GM1 System On	F0 7E 7F 09 01 F7	---
1 1 240	PC# Program Change	00H (1)	1-128*
1 1 260	CC#7 (Volume)	64H (100)	0-127
1 1 280	CC#10 (Pan)	40H (64)	0-127
2 1 000	Time Signature = 4/4	FF 58 04 04 02 24 08	---
2 1 000	Tempo = 120	FF 51 03 07 A1 20	---

[Note] * Program Change messages in consumer documentation are normally one-based; therefore, the decimal value of 1 presented here as the default is equivalent to 00H.

This example uses a time base of 480 PPQ. If another time base is used, the time index values should be scaled appropriately. For example, with a time base of 240 PPQ, the index values would be scaled by 1/2, so the first message sent would be at time index 1 1 120.

[*required*]
Time Signature and Tempo Events setting the actual time signature and tempo used at the beginning of the music must be included at Time Index 2 1 000. This requirement must be followed even if the actual music happens to begin with a time signature of 1/4 or a tempo of 240.

[Note] If you use an RPN in the setup bar, you must reset controllers #101 and #100 to value of 7FH (NULL).

[example]

Controller	Message	Value	Comment
101	RPN MSB	00H	Pitch Bend Sensitivity MSB
100	RPN LSB	00H	Pitch Bend Sensitivity LSB
6	Data Entry MSB	03H	3 Semitones Range
32	Data Entry LSB	00H	No additional Cents Range (optional)
101	RPN MSB	7FH	NULL MSB (reset)
100	RPN LSB	7FH	NULL MSB (reset)

See 5.3.4 Tempo and Calculating Delta Time (Elapsed Time) for calculation of absolute time.

(c) 2001 MIDI Manufacturers Association Incorporated All Rights Reserved

4.2 Recommended Steps for Content Production

This example provides a model for authoring GM Lite song content data.

<Step1>
Create music (or sound effects) by using commercially available MIDI sequencers.
The content author must follow the conditions defined in section 4.

Of course, each content author will have their own methods and procedures for creating content. Any approach is acceptable as long as the authoring guidelines are followed.

[Note] The content author may use any MIDI sequencer for creating content. However, if the MIDI sequencer used does not comply with the requirements mentioned above, the content author must add and/or modify the Set-Up Bar portion of the song data content using another MIDI sequencer which is compliant.

<Step2>
Verify that the created Content complies with the authoring guidelines
In order to confirm that the guidelines have been followed, the content author should check the data using editors in the MIDI sequencer application (e.g. tabular event list, graphical music editor). The author should also audition the songs on the actual target sound modules. Because of differences between various target portable devices, the resulting timbres and balance between various musical parts may not be quite the same as intended by the content author.

To achieve consistent sound playback experience over a wide range of devices, the content author should attempt to use, as much as possible, only those features and sounds which are supported by the greatest number of target devices.

To achieve the most consistent sound playback experience on all devices, content should be created using only those features and specifications which are available on every target device.

<Step3>
Mastering
The verified data in SMF Type 0 format is applicable as the master data file.

When song data content is prepared for a number of different portable devices, the data must be checked on all intended target devices.

4.3 Points to keep in mind during Content authoring

- No more than 16 notes may be used simultaneously within GM Lite song data content. The content author must confirm that this maximum limit is complied with, because there is a great possibility that the sound module used for authoring the Content will support more than 16 note polyphony. Checking the actual song data (perhaps using a software tool) is useful but not sufficient. For example, even if data inspection shows that only 16 notes are used simultaneously, the actual sound module could be attempting to play more than 16 notes simultaneously, because previously triggered notes could still be sounding the release portion of their envelopes after the Note Off messages have been received. The content author must audition the actual content data using a GM Lite Player in order to confirm that the accompaniment parts are not accidentally muted.

- Here are several tips for reducing the number of simultaneous notes:
  - If two parts are “doubled” to get a fat sound, such as unison or octave unison of strings, consider using only one part.
  - If the two “doubled” parts use different timbres (e.g. distortion guitar doubling clean guitar at a lower volume), try to choose a single timbre which will create the desired effect.

- Since increasing the simultaneous polyphony will also increase the load on the player’s CPU, it is important to carefully consider polyphony in order to avoid perceptible delays and latency in the sound module.

  Here is a concrete example:
  - The best way of avoiding perceptible playback delays is to reduce the number of notes that sound at the same time. However, when events such as Control Change messages are closely packed in various parts at the same time, it is necessary to put some time between them. (Better results are generally achieved using a wide interval such as 1 beat, rather than a short interval such as 5 ticks or 10 ticks.)

- Depending on speaker quality, it is often difficult to recognize the pitch of lower notes. It may be better to create GM Lite song data content that has been transposed up in pitch somewhere between a perfect 4th and one octave higher than the original key of the song. Recommended: check the GM Lite song data on speakers which are similar to those of the target playback device, and then make decisions on the actual transposition interval that should be applied to the song data.

- Drum parts tend to be masked by the other instrumental parts, so the recommended settings for Drum parts are: (a) set both volume and expression at the maximum of 127, and (b) bass drum and snare drum notes should use the maximum velocity value of 127.

- If volume, expression, and/or pan Control Changes are used during a sounding note, it is necessary to audition the GM Lite song data on actual target devices, in order to confirm that there is no zipper noise or other undesired sound during playback.

- GM Lite song data content should be created with the understanding that looped playback is often a function of the portable player, and that the end of the one performance should smoothly connect back to the beginning. Moreover, make sure that there are no notes sounding at the location of the End Of Track meta-event. There must be no note-on messages without matching note-off messages. Enough time should be allowed at the end of the song data so that the release envelope of the final note off can be completed before the End Of Track (EOT) meta-event occurs.

5 PLAYER GUIDELINES FOR GM LITE CONTENT

This section provides implementation guidelines for a GM Lite Player (playback device). It describes the content file format and how it is used by the Player. The implementers should reference the full MIDI specification and Standard MIDI File specification.

5.1 Target File

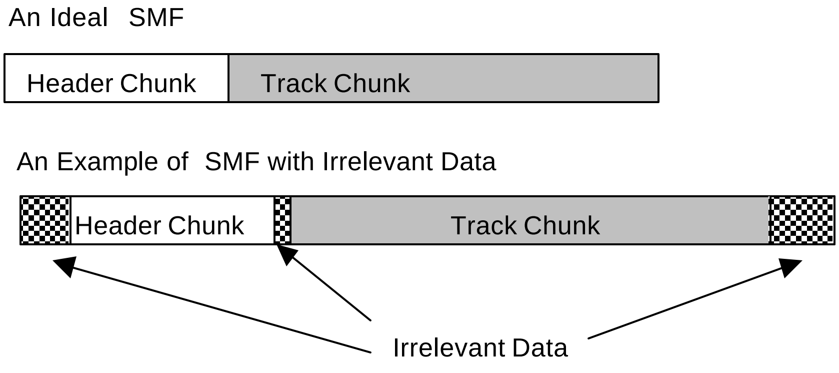
All GM Lite Players must support a SMF format 0 (Single Track) File that was created using the AUTHORING GUIDELINES FOR GM LITE CONTENT (section 4).

5.1.1 Chunk Extraction

The SMF data format contains 2 top-level element types: Header chunks and Track chunks. Each chunk contains a 4-byte Chunk Type field, a 4 byte Chunk Size field, and 0 or more data bytes. The Header chunk always precedes one or more Track chunks. A Type 0 SMF file contains exactly one Track chunk. (See the Complete MIDI 1.0 Detailed Specification for the SMF specification.)

In some cases, additional data may appear before the Header chunk, after the Track chunk, or even between the two chunks. This can occur when a SMF data image is embedded within another file format, or because of platform-specific differences in file storage. Such additional data is irrelevant to a GM Lite Player.

A GM Lite Player should take proper steps to handle data images that include such irrelevant data. Specifically, the Player should locate the Header and Track chunks, determine the length of each chunk from the corresponding ckSize field, and extract the corresponding valid data.



5.1.2 Header Chunk

The Header Chunk consists of the following fields:

	Size	Definition	
Chunk Type	:4 byte	‘M’’T’’h’’d’	in ASCII
Length	:4 byte	00 00 00 06	
Format	:2 byte	00 00	
ntrks	:2 byte	00 01	
Division	:2 byte	dM, dL	

Fields other than “Division” have fixed values; if other values are found, the Header Chunk is invalid.

(c) 2001 MIDI Manufacturers Association Incorporated All Rights Reserved

5.1.2.1 Division

The Division parameter defines how to interpret Delta Time values within the subsequent Track Chunks. The state of Bit 15 determines whether to use Time-code-based Time or Metrical (Bar-Beat-Tick) Time, as shown in the table below.

Byte line;	MSB								LSB								
Bit line;	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	0	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	: Metrical Time
	1	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	: Time-code-based Time

In files generated for GM Lite, Metrical Time is required and Time-code-based Time is not supported.

5.1.2.2 Metrical Time

Bits 0…14 define how the number of delta-time “ticks” which make up a single quarter note. For example, if `<Division>` is 96, then a time interval of an eighth-note between two events in the data would be 48.

The actual duration of a quarter note is determined by Set Tempo meta-events, which are contained within a Track Chunk. Tempo is measured in units of microseconds per quarter note. For example, the default tempo of 120 beats per minute would be indicated by a value of 500,000.

The following Divisions are normally used.
12, 24, 48, 60, 96, 120, 192, 240, 384, 480

It is highly recommended that GM Lite players use the highest feasible precision (e.g. 480).

5.1.3 Track Chunk

The Track Chunk consist of the following fields:

	Size	Definition
Chunk Type	: 4byte	‘M’’T’’r’’k’ in ASCII
Length	: 4byte	
Track Data	: Variable	Stream of <Delta Time, MIDI event> elements

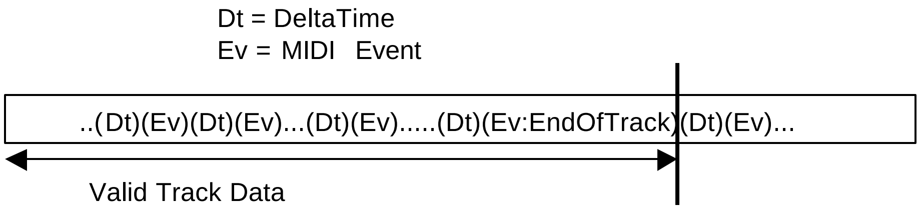
The Length field defines the size of the data in the Track Chunk (excluding the chunk type and length fields). It does not necessarily indicate the actual end of the stream of <Delta Times, MIDI event> elements.

The Track Data field contains the Stream of <Delta Times, MIDI event> elements. Occasionally, when the End of Track meta-event occurs before the last byte contained in the Track Data field, it is possible that the final portion of the Track Data field may contain invalid data.

(c) 2001 MIDI Manufacturers Association Incorporated All Rights Reserved

5.1.3.1 “End Of Track” on Player’s Recognition

The Player shall parse each <Delta Time, MIDI event> starting from the beginning of the Track Data.
If the End of Track (EOT) meta-event is recognized during parsing, any subsequent data within the Track Data field should be considered invalid. (In fact, there should not be any subsequent data!)


Dt = DeltaTime
Ev = MIDI Event
..(Dt)(Ev)(Dt)(Ev)...(Dt)(Ev).....(Dt)(Ev:EndOfTrack)(Dt)(Ev)...
Valid Track Data

Delta times within the valid Track Data should be included as part of performance time.

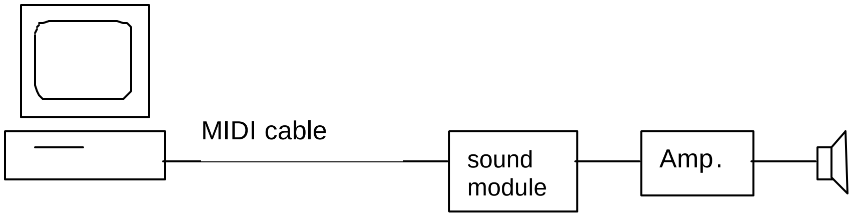
When player recognizes the End of Track during the performance, All Notes Off and All Sound Off commands must be sent to every channel that is in use by that track.
- See Section 4. AUTHORING GUIDELINES FOR GM LITE CONTENT for further information.

5.2 Player Configuration

The Authoring Guidelines state that song data content production may be carried out by ordinary consumers using commercially available MIDI sequencers. This means there is the possibility of non-supported messages being contained in content that were generated by such applications. The Player implementation must take this into account and should be designed with a full understanding of SMF structure and basic MIDI concepts, including the ability to ignore unrecognized events.

5.2.1 General Authoring Configuration

In a typical content authoring configuration, the player is some type of PC application software (e.g. MIDI sequencer). The actual music is generated by a sound module linked to the PC by a MIDI cable.
The following shows a typical authoring setup:



The MIDI 1.0 serial data transmission protocol operates at a rate of 31.25 kbps. That means that a 3 byte message takes about 1 millisecond to be transmitted over a MIDI-DIN cable. However, within the logical SMF song content format, it is possible to specify two or more messages that all have the same start time. As a result, even though it appears that such messages will occur simultaneously, there will actually be an interval of at least 0.7 to 1 millisecond between each successive message (most messages are 2 or 3 bytes in length).

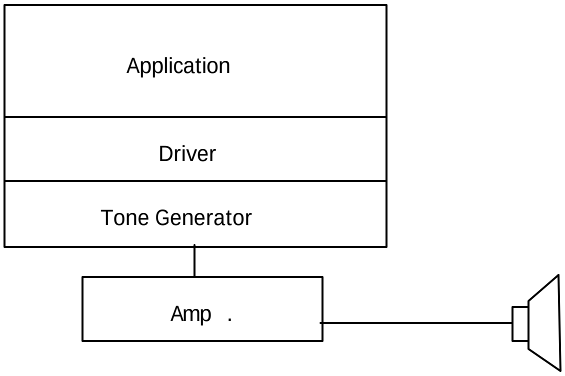
It is essential to give the above facts full consideration when designing and implementing a GM Lite Player.

(c) 2001 MIDI Manufacturers Association Incorporated All Rights Reserved

5.2.2 Player Configuration for Portable Devices

A portable device containing a GM Lite Player typically includes the playback application, sound module (including a driver and a tone generator), amplifier and speakers within a single device. It is quite possible that such a device may not be affected by the physical data transmission speed constraints discussed in 5.2.1 above. In such portable devices, certain events might take tens of milliseconds to process, while other events might be processed very quickly (e.g. processing 3 events in less than a millisecond, when usually it would take 3 milliseconds to process 3 events).

It is highly recommended that this issue be given full attention when implementing a Player, in order to ensure compatibility and consistent playback performance. The recommended player configuration for portable devices is shown below:



The application is responsible for parsing the SMF into track chunks and individual message events, scheduling each message event, and sending each message event to the driver at the appropriate time. The driver is responsible for transmitting each message to the tone generator and managing the hardware- or register-level interface to the tone generator. The tone generator may be hardware-based, software-based or some combination of both.

5.2.2.1 Driver Considerations

The driver is usually required to transmit messages to the tone generator at maximum speed. However, optimum reproduction will occur if MIDI messages are processed at a maximum rate of 3125 bytes/sec (equivalent to 31.25 kbps). This avoids certain problems which can occur if the song data content is authored improperly. (Note: this restriction is appropriate for GM Lite players, but does not apply to high-performance music products.)

In order to comply with this recommendation, it may be necessary to introduce some additional delay in the high-speed interface between the driver and the tone generator.

Problems can occur when the effective driver-to-tone-generator message transmission rate is significantly faster or slower than 3125 bytes per second.

Example 1: High-Speed Driver

A guitar note has a fast (abrupt) attack. However, a guitarist can use a volume pedal to fade in the note, creating a “swelling” effect somewhat like a violin. A single MIDI Note On message and a series of Channel Volume messages can be used to simulate this, effectively “erasing” the normal attack of the synthesized guitar sound. The corresponding SMF data might look something like this:

(c) 2001 MIDI Manufacturers Association Incorporated All Rights Reserved

Time Index	MIDI Message	
2 1 00	CC#	Channel Volume = 127
2 2 00	CC#	Channel Volume = 0
2 2 00	NoteOn	Middle C
2 2 00	CC#	Channel Volume = 5
2 2 05	CC#	Channel Volume = 20
2 2 07	CC#	Channel Volume = 25
2 2 09	CC#	Channel Volume = 30
2 2 11	CC#	Channel Volume = 35
2 2 13	CC#	Channel Volume = 45
….	….	….

Note that at time index 2.2.00, the Note On message is preceded by one Channel Volume message and followed immediately by another, but there is no delta time between these three events. Because it will take about 1 millisecond to transmit the Channel Volume over a MIDI-DIN cable, and another millisecond to transmit the Note On, the receiving sound module has plenty of time to set Channel Volume to 0 before it starts to play the note.

However, if the Channel Volume and Note On messages are sent over a high-speed connection, with no time delay between them, it's possible that the sound module will briefly play the normal guitar attack before setting the volume to zero. (Some sound modules may limit the maximum rate of change for volume, to avoid introducing audible clicks; which may allow the normal guitar attack to sound even longer.) In this case, the Player will produce results different from those heard when the content was authored.

[Note] Such problems can and should be prevented during Content Authoring, by making sure that the first such Channel Volume message precedes the Note On by a few ticks. In addition, the Player sound module should be designed so that Volume Changes preceding a given Note On affect that note immediately, without applying the rate-limiting mechanism used for de-clicking volume changes to a note that is already in progress.

Example 2: Low-Speed Driver

A low-speed driver (with bandwidth significantly less than 3125 bytes/sec) can degrade the rhythmic accuracy of the music content, making it sound clumsy or irregular. Music data typically contains clusters of tightly-spaced events separated by wider intervals. A low-speed driver can "smear" or distort the fine timing detail of a grace note or arpeggiated chord, or change the rhythmic feel of a drum part. Furthermore, the ability to identify musical timbres is strongly linked with their attack transients. If a low-speed driver changes the relative timing between the attacks of two or more nearly-simultaneous sounds, this can change the perceived musical timbre.

Consider a single chord, containing four Note On messages along with Channel Volume and Modulation messages for expression. At content authoring time, all events in the chord were spaced at 1 millisecond intervals, requiring about 6 milliseconds to transmit and start playing the chord.

A low-speed driver might take 10 to 20 milliseconds (or more) to process the same chord. Instead of a relatively tight, percussive attack, the chord attack is smeared, and may even sound arpeggiated.

(c) 2001 MIDI Manufacturers Association Incorporated All Rights Reserved

5.3 Special Considerations

This section discusses several aspects that must be handled carefully under different operating conditions, such as the playback of simple data or the playback of repeated data such as alert tones or ringing tones with melodies.

5.3.1 The Beginning of Performance

Normally, there is a period of silence (from several tens of milliseconds to a few seconds) at the beginning of a GM Lite song data file. This silent portion includes the GM1 System On, Set-Up Bar and optional control changes. This initial silence can reasonably be interpreted as “a part of the performance”. However, in applications which need repeated playback, such as ringing tones with melodies in mobile phones, this initial silence is inappropriate and should be avoided. It is essential to understand these facts in order to handle the set-up bar correctly. (See Section 5.3.3 Dealing with the Setup Bar.)

[*Recommended*]
The Player should check the song data in order to determine whether or not there is a Set-Up Bar. To do this, the Player should check the initial portion of the data for the following events:

Time Index	Data (Hex)	
1 1 000	FF 58 04 01 02 24 08	Beat = 1/4 meta event
1 1 000	FF 51 03 03 D0 90	Tempo=240 meta event
1 1 000	F0 7E 7F 09 01 F7	GM1 System On sysex message
(The above simultaneous events may occur in any order)		
……….		
2 1 000	FF 51 03 ** ** **	Tempo=n (tempo of the actual performance)

If the Player does not find the events shown above at time index {1 1 0} , the Player should assume that there is no Set-Up Bar. If there is no Beat or Tempo message of any kind at the beginning of the performance, the Player should assume that there is no Set-Up Bar, and apply the regular SMF default values:

Beat: 4/4, Tempo: 120

5.3.2 Dealing with Reset of Sound Module (GM1 System On)

Normally, the output should be muted when the sound module receives GM1 System On.
However, if the GM Lite song data content is played repeatedly (looped), resending the GM1 System On message could cause the release portion of one or more notes to be cut off prematurely when the Player begins to repeat a portion of the song data content.

A GM1 System On message must be sent shortly before a particular performance begins, to specify the active mode of the target sound module. After the sound module has been properly set up for a performance, it is not necessary to resend the GM1 System On message each time that part or all of the song data content is played repeatedly as part of a single continuous performance.

[*Recommended*]
The designer should consider when (and when not) to send the GM1 System On messages, taking the above information into account.

5.3.3 Dealing with the Set-Up Bar

Section 4 AUTHORING GUIDELINES FOR GM LITE CONTENT specifies that the first bar of song data content is a Set-Up Bar. This Set-Up Bar is silent (contains no playable notes).

If the song data content is played repeatedly (looped), and the Set-Up Bar is played at normal speed each time it is encountered, there will be an audible gap or silence at this point in the performance starting from the 2nd playback.

[*Recommended*]
If a valid Set-Up Bar is found, the Player should transmit all data in the Set-Up Bar the first time that song data content is played. The first message sent should be the GM1 System On system exclusive message. There should be an interval of at least 125 milliseconds between the GM1 System On message and any subsequent data in the Set-Up Bar:

GM1 System On .... 125 milliseconds .... (Program Change), (Control Change, etc......)
..... start of Bar 2
.... first actual note data

After the GM1 System On is sent, and the 125 millisecond delay has elapsed, any remaining data in the Set-Up Bar should be chased as rapidly as possible (see note below), so that the audible song data can be played immediately (at normal speed).

If the song data content is played repeatedly (looped), the Set-Up Bar should be handled differently.
For the second and subsequent playback(s) of the song data, the Player should NOT send the GM1 System On message (and insert a 125 millisecond delay). This would reset the sound module and cause an audible gap between the end of one repeat and the start of the following repeat. Instead, the Player should “chase” all other data in the Set-Up Bar as rapidly as possible (excluding the GM1 System On and related delay). Normal song data playback should resume starting from Bar 2, with no audible silence or rhythmic irregularity between the end of the previous repeat and the start of the following repeat.

[Note] “Chase” means to process song data by transmitting non-note sequence data as quickly as possible until the target time position is reached. No tone generation should occur during the chase, and Note Messages should not be transmitted. When multiple messages of the same type (e.g. volume and expression) occur during the data segment processed in this way, only the last such message found within the data segment should be transmitted.

5.3.4 Tempo and Calculating Delta Time (Elapsed Time)

The location of each event within a Track Chunk is measured in ticks. (A tick is the smallest unit of musical time, and the actual duration of a tick varies according to tempo.)

In order to play an event, the tick position of that event must be converted to the corresponding time value:

tick position ::= musical position of an event, relative to the beginning of the Track (not affected by tempo)
time value ::= actual time when an event occurs, in microseconds (affected by tempo)

During performance, the Set Tempo meta-events and the Delta-time of each MIDI event are used to calculate the actual time value of each event. In making these calculations, however, utmost consideration must be given to minimizing computing errors.

(c) 2001 MIDI Manufacturers Association Incorporated All Rights Reserved

As described in Metrical Time (5.1.2.2), the Set Tempo meta-event specifies the actual duration of a quarter note in microseconds. The number of ticks per quarter note is specified by the Division field in the Header Chunk. If a Track Chunk does not contain a Set Tempo meta-event, the tempo is set to the default value of 120 beats per minute (500,000 microseconds per quarter note). Thus, using the Division and Set Tempo event values, the actual duration of a tick is calculated as follows:

Actual duration of 1 tick [in microseconds] = Current Set Tempo meta-event value / Division

However, simple application of the above formula can produce significant rounding errors. Assume that (1) the tick position-to-time value calculation for one event contains some rounding error, and (2) the time value for each event is calculated based on the accumulated time values for all preceding events. In this case, the final playback time value will include the sum total of all rounding errors for all preceding events. In other words, there is a high probability that playback will not sound as intended by the composer or creator, because the timing will not be correct.

[Example]
Let Division = 480 ticks and Set Tempo value = 500,000 microseconds (120 beats per minute).
The duration of 1 tick is calculated by dividing 500,000 by 480:

Tick Duration = 500,000 / 480 = 1041.66666.... microseconds
(or 0.00104166666... seconds)

At a tempo of 120 BPM, a 3.5 minute song will have a duration of about 200,000 ticks
(120 BPM = 2 quarter notes per second or 960 ticks/sec; 960 * 210 seconds == 201,600).

Assume the Player implementation calculates time values to five decimal places (to a precision of 10 microseconds, or .00001 seconds). The time value corresponding to tick position 200,000 should be calculated as 208.33333 seconds. However, note that this implementation will calculate the duration of a single tick as 0.00104 seconds (1040 microseconds).

Assume that at least one event (such as an “Expression” Control Change) occurs in every tick, and that the entire Track contains 200,000 MIDI events. Assume that the Player calculates the time valueof the final event by summing the 0.00104 second interval between events. The final event time value will be calculated as 208.00000 seconds, rather than 208.33333 seconds – resulting in an error of 0.33333 seconds.

Thus, an error of about 0.3 seconds could occur in an average song with a length of about 3.5 minutes. This amounts to a performance timing error of more than one eighth note (at 120 BPM, 1/8th note == 0.25 seconds).

When only a single device is playing, this amount of timing error does not create a noticeable problem. However, this is not true when the same data (or different parts of the same song) are played back using several different devices, or when such data is played back in synchronization with other content such as MPEG data. In such cases, the timing differences resulting from computing error could be noticeable enough to merit serious consideration.

Thus, it is strongly recommended that player implementations should take steps to decrease timing errors as much as possible.

(c) 2001 MIDI Manufacturers Association Incorporated All Rights Reserved

5.3.5 Player Management of the Event List (Message List)

As mentioned above, computing errors will accumulate to a significant extent if the actual durations of the delta-times for a series of events are used to calculate the performance time for following events.

Therefore, the recommended approach for managing the Event List is as follows:

a) The absolute tick position of each event, relative to the beginning of the Track, should be determined by summing the Delta-time values (in ticks) of all preceding events.

b) The absolute tick position of each event should then be converted to an absolute time value, rather than to a time duration relative to the previous event time.

c) Set Tempo events must be taken into consideration. If there are multiple Set Tempo events in a Track, a simple conversion from absolute tick position to absolute time value using only the most recent Set Tempo value will produce errors. One method is to calculate and save the absolute time value and tick position for each Set Tempo event, and then calculate the absolute time value for each successive event using the <time value,tick position> saved for the most recent Set Tempo event.

[Note] Higher-precision integer or fixed-point calculations can optionally be used to further improve timing accuracy.

(c) 2001 MIDI Manufacturers Association Incorporated All Rights Reserved

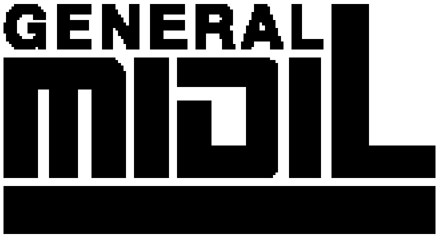
6 GENERAL MIDI LITE LOGO

The sound module that meets the definition of this RP is hereby qualified as a Sound Module for “General MIDI Lite”. The content produced in compliance with the Authoring Guidelines for GM Lite Content indicated in this RP is approved as Song Data Content for “General MIDI Lite”.

“General MIDI Lite” shall be abbreviated to ”GML”. A license to use the logo shown below is available to equipment with GML sound modules and software as well as Content for GML. The license for this logo implies obligation to assure utmost compatibility between sounds of GML sound module when the GML content which is compliant with the “Authoring Guidelines for GM Lite Content” is played by the “Player Guidelines for GM Lite Content” compliant player. To apply the logo onto devices where a GML sound module and a player are combined together, it is required that the player meets at least what this specification and associated guidelines define.

If it is difficult to display the logo exactly on an LCD or a small-sized device, it is required to display the logo as accurately as possible.

This logo is the property of AMEI and MMA and the trademark and copyright are reserved.
The GML logo is used under the application at AMEI (Japan) or MMA (all other countries) and must be used in accordance with guidelines provided by AMEI or MMA. Please contact AMEI or MMA for further details.



6.1 Contacts

AMEI (Association of Musical Electronics Industry)
Ito-Bldg. 2-16-9 Misaki-cho, Chiyoda-ku, Tokyo, 101-0061, Japan
Phone : +81-3-5226-8550 Fax : +81-3-5226-8549
http://www.amei.or.jp/

MMA (MIDI Manufactures Association)
P.O. Box 3173
La Habra, CA 90632-3173,USA
Email: mma@midi.org
http://www.midi.org/

7 REFERENCES

- **The Complete MIDI 1.0 Detailed Specification**
  (Includes SMF and General MIDI 1 Specifications)

  (Japanese Version 98.1)
  Written and published by AMEI (Association of Musical Electronics Industry)
  Distributed by RittorMusic ISBN 4-8456-0348-9
  http://www.amei.or.jp/news/news5.html

  (English Version)
  Written and published by MMA (MIDI Manufacturers Association Incorporated)
  http://www.midi.org/

(c) 2001 MIDI Manufacturers Association Incorporated All Rights Reserved