MIDI Channel Voice Messages

All MIDI status byte and data byte values are in hexadecimal

MIDI Channel Voice Messages

Status Byte	Data Bytes		Message		Description	
8*n*	*kk*	*vv*	 Note off		Normally sent when a key (on a synthesizer) is released (note 1)	
			Value	**Range**	**Description**	
			n	0-F	MIDI Channel	0 = Ch 1
			kk	00-7F	Key which was released This must correspond to a previous note-on message for correct operation	3C = Middle-C
			vv	00-7F	Velocity with which key was released Devices which are not velocity sensitive should send *vv*=40 The interpretation of this message is up to the receiving MIDI device	00=min 40=default 7F=max
Status Byte	**Data Bytes**		**Message**		**Description**	
9*n*	*kk*	*vv*	 Note on		Normally sent when a key (on a synthesizer) is pressed A corresponding note-off message *must* be sent for each and every note-on message	
			Value	**Range**	**Description**	
			n	0-F	MIDI Channel	0 = Ch 1
			kk	00-7F	Key which was pressed Each value is a 'half-step' above or below the adjacent values	3C = Middle-C
			vv	00-7F	Velocity with which key was pressed Devices which are not velocity sensitive should send *vv*=40 By definition, a note-on message with *vv*=0 is equivalent to the message: "note-off *vv*=40" (note 2)	00=note-off (40) 01=*ppp* 40=*mf* 7F=*fff*
Status Byte	**Data Bytes**		**Message**		**Description**	
A*n*	*kk*	*ww*	 Polyphonic Key Pressure		Also known as Aftertouch. This message is sent when there is a change in the pressure being applied to a key (ie on a per-key basis).	
			Value	**Range**	**Description**	

		n	0-F	MIDI Channel	0 = Ch 1
		kk	00-7F	Key which was pressed Each value is a 'half-step' above or below the adjacent values	3C = Middle-C
		ww	00-7F	Pressure with which key is being pressed	00=min, 7F=max
Status Byte	**Data Bytes**		**Message**	**Description**	
B*n*	*cc*	*nn*	 Controller Change	Sent when a change is made in a footswitch, expression pedal, slider, or other controller.	
		Value	**Range**	**Description**	
		n	0-F	MIDI Channel	0 = Ch 1
		cc	00-77	Controller Number	see detailed descriptions
		nn	00-7F	Controller Value	00=min, 40=center, 7F=max
Status Byte	**Data Bytes**		**Message**	**Description**	
C*n*	*pp*		Program Change 	Used to change the instrument (or sound) to be played when a note-on message is received. This is usually not retro-active, and only applies to subsequent note-on messages This message may have a completely different interpretation depending on the type of device. For example, it could change the current rhythm on a drum-machine.	
		Value	**Range**	**Description**	
		n	0-F	MIDI Channel	0 = Ch 1
		pp	00-7F	New Program number	00=1st program
Status Byte	**Data Bytes**		**Message**	**Description**	
D*n*	*ww*		 Channel Key Pressure	Also known as Aftertouch. This message is sent when there is a change in the overall pressure being applied to the keyboard (ie for the channel overall, and *not* on a per-key basis).	
		Value	**Range**	**Description**	
		n	0-F	MIDI Channel	0 = Ch 1
		ww	00-7F	Channel Pressure Value	00=min, 7F=max
Status Byte	**Data Bytes**		**Message**	**Description**	
E*n*	*lsb*	*msb*	 Pitch Bend	Sent when a change is made in a pitch-bender lever.	
		Value	**Range**	**Description**	

	n	0-F	MIDI Channel	0 = Ch 1
	lsb	00-7F	Least significant byte	00=min, 00=center, 7F=max
	msb	00-7F	Most Significant Byte	00=min, 40=center, 7F=max

Footnotes

Note 1

Just because a device has received a note-off message does not automatically imply that the note should cease abruptly. Some sounds, such as organ and trumpet sounds will do so. Others, such as piano and guitar sounds, will decay (fade-out) instead, albeit more quickly after the note-off message is received.

Note 2

Sending note-on with *vv*=0 improves the effectiveness of Running Status. Hence this message is preferred over the regular note-off message for devices which do not detect release velocity.

MIDI Channel Voice Messages

The table below presents a summary of the MIDI Channel Voice Message codes in binary form.
A MIDI channel voice message consists of a **Status Byte** followed by one or two **Data Bytes.**

Status Byte	Data Byte 1	Data Byte 2	Message	Legend
1000nnnn	0kkkkkkk	0vvvvvvv	Note Off	n=channel* k=key # 0-127(60=middle C) v=velocity (0-127)
1001nnnn	0kkkkkkk	0vvvvvvv	Note On	n=channel k=key # 0-127(60=middle C) v=velocity (0-127)
1010nnnn	0kkkkkkk	0ppppppp	Poly Key Pressure	n=channel k=key # 0-127(60=middle C) p=pressure (0-127)
1011nnnn	0ccccccc	0vvvvvvv	Controller Change	n=channel c=controller v=controller value(0-127)
1100nnnn	0ppppppp	[none]	Program Change	n=channel p=preset number (0-127)
1101nnnn	0ppppppp	[none]	Channel Pressure	n=channel p=pressure (0-127)
1101nnnn	0ccccccc	0fffffff	Pitch Bend	n=channel c=coarse f=fine (c+f = 14-bit resolution)

*The last four bits of a status byte signify the MIDI channel (1-16) the message is intended for offset by 1 [0000=channel #1, 1111=channel #16]

An example of a MIDI channel voice message would be:

10010001 00111100 01000000

The first four bits of the status byte specify the Note On command, while the last four specify MIDI channel #2. The first data byte specifies the key number for middle C (=60 decimal), while the second data byte specifies a median velocity (= 64 decimal).

Running Status

MIDI's Running Status allows a single status byte's action to remain in effect for an unlimited number of data byte pairs which follow. For example, to play three notes on the same MIDI channel, a Note On status byte can be sent, followed by six data bytes (key # + velocity of note 1, key # + velocity of note 2, key # + velocity of note 3). To help minimize excessive data by using running status, the Note On command can also function to turn notes off by sending a velocity value of zero for the key # to be turned off.

midi_modes.html Contents midi_control_change.html midi_channel_mode.html