DICER

Programmer's Reference

www.novationmusic.com

1 Introduction

This manual describes Dicer's MIDI communication format. This is all the proprietary information you need to be able to control Dicer to its full potential.

It is assumed that you already have a basic knowledge of MIDI, and some appropriate software for writing interactive MIDI applications (for example, Max for Live, Max/MSP, or Pure Data).

Numbers in this manual are given in both hexadecimal and decimal equivalents, as different software favours the use of different conventions. To avoid any ambiguity, hexadecimal numbers are always followed by a lower-case h.

2 Dicer MIDI Overview

Dicer is a class-compliant MIDI device. Depending on your operating system, it will appear automatically to your computer as 'Dicer', 'Novation Dicer', or 'USB Audio Device'.

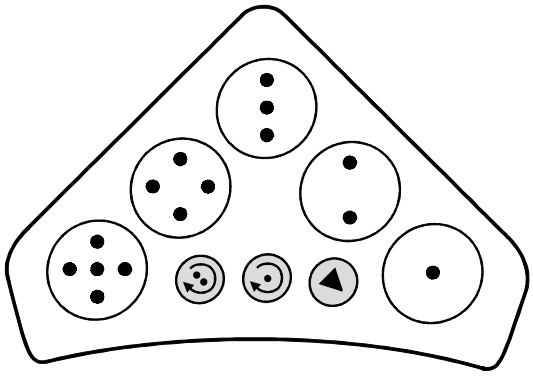

Figure 1. Outline of Dicer, with its eight buttons.

Each unit has eight buttons (Figure 1): three small *mode* buttons, and five large *dice* buttons. Every button is back-lit by a bi-coloured LED that can produce red, a shade of orange or yellow, or green in a number of intensities.

All MIDI communication with Dicer is conducted using note-on, note-off, and controller change messages. Note on and note off messages are used to convey button pushes and releases, and controller change messages are used for configuration data.

Interconnecting Dicers

To conserve USB ports, two Dicers can be interconnected using a normal 3.5mm stereo jack-to-jack cable. They will appear to the computer as a single device and work identically, except that the Dicer connected directly to the USB will communicate on MIDI channels 11 to 13, and the Dicer at the far end of the link cable will communicate on MIDI channels 14 to 16. Throughout this document, this optional second Dicer is referred to as the *slave* Dicer, and the unit connected directly to the USB is called the *master* Dicer.

In order to aid visual identification and detect cable problems, a single Dicer will briefly flash amber when it is plugged into a computer. In a two-Dicer system, the master Dicer will flash red and the slave Dicer will flash green.

Modes and MIDI channels

Although each Dicer has five dice buttons, it keeps a continual record of the status of sixty virtual buttons, mapped to a range of MIDI notes and channels. The backlight LEDs that are shown at any time, and the notes that are sent when buttons are pressed, represent just a small group of these. Every time a mode is changed using one of the mode buttons, or a shift state is entered by holding down the mode button that is currently selected, the page of notes represented by the dice buttons changes.

There are three mode buttons, each of which can work as a shift button, accounting for $3 \times 2 \times 5 = 30$ buttons. The other thirty virtual buttons store the complete status of the slave Dicer. This means that the backlight and configuration status of a slave Dicer can be programmed even if it is not connected, and that a slave Dicer can be hot-plugged or unplugged without any of its status being lost.

Special features

It is possible to make Dicer treat all dice button presses as if they are shifted, by holding down a mode button for longer than a certain time. This mode is called *shift-lock mode*, and persists until it is cancelled by another press of a mode button. The threshold time can be configured using MIDI messages. By default, this feature is turned off.

The mode buttons do not normally send MIDI data to the computer, and Dicer manages its mode backlights automatically. This behaviour can be modified in two ways. If the exact mode and shift status of Dicer is important to a certain program, the mode buttons can be configured to relay MIDI information to keep that program informed. Alternatively, Dicer's mode management can be turned off completely so that it can be used as a dumb eight-button keypad. This is done by sending the appropriate MIDI message to turn Dicer to *one-page mode*.

MIDI summary

A valid Dicer message takes one of the forms below. The status byte (the byte that starts the MIDI message) is summarised in Figure 2. Although Dicer will respond to MIDI note off messages, it does not send them.

Dicer	Message type	Hex version	Decimal version
Master	Note off	8A-8Ch, *Key, Velocity*	138-140, *Key, Velocity*
	Note on	9A-9Ch, *Key, Velocity*	154-156, *Key, Velocity*
	Control change	BAh, *Controller, Data*	186, *Controller, Data*
Slave	Note off	8D-8Fh, *Key, Velocity*	141-143, *Key, Velocity*
	Note on	9D-9Fh, *Key, Velocity*	157-159, *Key, Velocity*
	Control change	BDh, *Controller, Data*	189, *Controller, Data*

Button	Mode / Mode backlight	MIDI channel: Master Dicer	MIDI channel: Slave Dicer
	Hot Cue / red	channel 11 starts: 9Ah (154)	channel 14 starts: 9Dh (157)
	Loop Roll / green	channel 12 starts: 9Bh (155)	channel 15 starts: 9Eh (158)
	Auto Loop / amber	channel 13 starts: 9Ch (156)	channel 16 starts: 9Fh (159)
	One-page mode	channel 11 starts: 9Ah (154)	channel 14 starts: 9Dh (157)
	All controller messages	channel 11 starts: BAh (186)	channel 14 starts: BDh (189)

Figure 2. Summary of modes, MIDI channels, and status bytes.

3 Computer-to-Dicer Messages

Note Off

Hex version	8A-8Fh, *Key*, *Velocity*.
Decimal version	138-143, *Key*, *Velocity*.

This message is interpreted in exactly the same way as a note-on message containing the same key code, and velocity zero. The velocity byte contained within the note-off message is ignored.

Set LEDs

Hex version	9A-9Fh, *Key*, *Velocity*.
Decimal version	154-159, *Key*, *Velocity*.

A note-on message changes the state of a LED. *Key* is the MIDI note number, which determines the LED location. *Velocity* is used to set the LED colour.

The key code is determined according to Figures 3 and 4.

Button	MIDI note	
	normal	**shifted**
	3Ch (60) C-3	41h (65) F-3
	3Dh (61) C#3	42h (66) F#3
	3Eh (62) D-3	43h (67) G-3
	3Fh (63) D#3	44h (68) G#3
	40h (64) E-3	45h (69) A-3

Figure 3. Key codes for each button in conventional modes.

Button								
MIDI note	3Ch (60) C-3	3Dh (61) C#3	3Eh (62) D-3	3Fh (63) D#3	40h (64) E-3	41h (65) F-3	42h (66) F#3	43h (67) G-3

Figure 4. Key codes for each button in one-page mode.

Velocity is determined by choosing a number according to the colour of the button:

Value	Meaning
00h (0)	Red
10h (16)	Red-orange
20h (32)	Orange
30h (48)	Orange-amber
40h (64)	Amber
50h (80)	Yellow
60h (96)	Yellow-green
70h (112)	Green

To this number, a LED intensity from 0 to 15 (Fh) can be added, where 0 is off and 15 is maximum brightness. Dicer will light the button accordingly.

Reset Dicer (master Dicer only)

Hex version	BAh, 00h, 00h.
Decimal version	186, 0, 0.

All LEDs on both Dicers are turned off, and the devices are restored to their power-on defaults. Note that this will not halt a light show if one is in progress: this can be done manually by sending BAh, 00h, 29h (decimal 186, 0, 41) to the master, and BDh, 00h, 29h (decimal 189, 0, 41) to the slave.

Read mode and connection information (master Dicer only)

Hex version	BAh, 00h, 11h.
Decimal version	186, 0, 17.

This command requests status and connection information from both Dicers. The information is returned in a message of the form BAh, 11h, … (decimal 186, 17, ….), which is documented in Section 5.

All LEDs off (master Dicer only)

Hex version	BAh, 00h, 70h.
Decimal version	186, 0, 112.

This command cannot be sent separately to the slave, as it affects both Dicers. This message will turn off all dice LEDs in all modes, but will not affect the Dicers in any other way. It is therefore a useful shortcut for quickly turning off all backlights without affecting the configuration settings.

Light show

Hex version	BA/BDh, 00h, 20-29h.
Decimal version	186/189, 0, 32-41.

Dicer contains two automatic light shows. These are usually used to diagnose problems with the unit, but can also be controlled via MIDI. A continuous chasing effect can be turned on by setting the data byte in this message to 28h (40). A message with the data byte 29h (41) is the only way to turn off the chasing effect. A one-shot effect can be run in any of Dicer's eight chrominance modes, using a data byte in the range 20-27h (32-39) for red to green.

The chasing effect is usually used by Dicer as a diagnostic activity when it is receiving power, but has not yet received configuration data from the host computer or another Dicer. This can happen when it is run from a powered hub with no computer connected, the computer's MIDI class drivers have not been automatically installed, a mono rather than stereo link cable is used to try to connect the Dicers together, or there is a USB cable fault that permits power but not data to reach Dicer.

The one-shot effect is usually seen as a red, amber, or green identifying flash when Dicer is connected to a computer (see 'Interconnecting Dicers' in the previous section).

Change mode

Hex version	BA/BDh, 11h, *Mode*.
Decimal version	186/189, 17, *Mode*.

Force the master or slave Dicer into a different mode, as follows:

mode	**meaning**
0	Hot Cues (red).
1	Hot Cues (red), shift lock.
2	Loop Roll (green).
3	Loop Roll (green), shift lock.
4	Auto Loop (amber).
5	Auto Loop (amber), shift lock.
6	One-page mode.

Sending this message is the only way to enter one-page mode, or a shift lock mode when the shift lock behaviour is disabled.

Unlock advanced features (master Dicer only)

Hex version	BAh, 11h, 55h.
Decimal version	186, 17, 85.

This message must be received by the master Dicer before the more advanced messages will be processed by either the master or the slave. It is an insurance against accidental transmission of MIDI data that might force Dicer into an unusable (or less useful) state.

4 Advanced Computer-to-Dicer Messages

These messages will be ignored unless 'Unlock advanced features' (BAh, 11h, 55h) is transmitted first.

Set shift lock timeout

Hex version	BA/BDh, 12-14h, *Time*.
Decimal version	186/189, 18-20, *Time*.

Set the length of time, in units of 1/20 second, that a shift button must be held down in order to enter shift lock mode. A value of zero, the default, disables shift lock altogether. A sensible enabled value is around Fh (15), or 750ms. The controller numbers 12–14h (18-20) control the threshold times for Hot Cues, Loop Roll, and Auto Loop modes respectively.

Even when advanced features are disabled, a shift lock can temporarily be entered at any time using the appropriate 'change mode' message.

Transmit mode button events (master Dicer only)

Hex version	BAh, 15h, *Data*.
Decimal version	186, 21, *Data*.

This mode is switched off by default. Sending this message with *data* = 1 causes the mode/shift buttons of both Dicers to send MIDI note on messages when they are pressed and released, using their appropriate channels and MIDI note zero (C--2). Sending the message with *data* = 0 switches the mode off again.

5 Dicer-to-Computer Messages

Button pressed

Hex version	9A-9Fh, *Key*, *Velocity*.
Decimal version	154-159, *Key*, *Velocity*.

In these messages, *Key* is the key value as depicted in Figures 3 and 4, or zero when conveying a mode button press when this is configured (see 'Transmit mode button events' in the previous section).

Only three MIDI velocities can be sent by Dicer:

- 7Fh (127) indicates that a button has been pressed;
- 0 indicates that a button has been released;
- 40h (64) indicates a 'partial release', encountered when a mode button has been released, but Dicer has entered shift lock mode.

A partial release is unusual. Transmission of such a message must explicitly be enabled by sending at least three MIDI controller messages. Dicer can be made to send it, for example, by transmitting:

```
BAh, 11h, 55h     enable advanced features
BAh, 15h, 01h     transmit mode button events
BAh, 12h, 0Fh     enable shift lock in Hot Cues mode, timeout 750ms
```

then holding down the red mode button for a second or so, and then releasing it.

Mode and connection information

Hex version	BAh, 11h, *Data*.
Decimal version	186, 17, *Data*.

This message is sent on power-up, whenever a slave Dicer is connected or unplugged, or in response to an explicit request for it. The data byte is encoded in the following way:

bit	meaning	value
6..4	Slave Mode	0–6 (see 'Change mode' message in Section 3.)
3	Paired	1 only if a slave Dicer is connected.
2..0	Master Mode	0–6 (see 'Change mode' message in Section 3.)

The formula for interpreting this value is therefore:

Data = 16 × slave mode number
+ master mode number
+ 8 if paired.