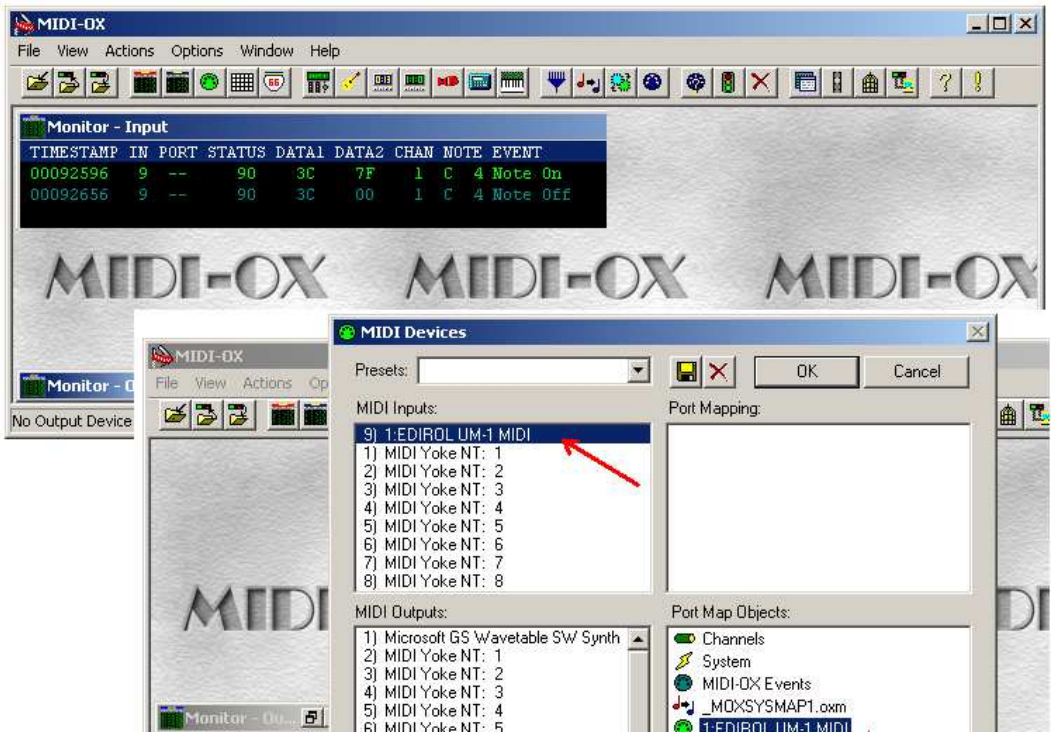
Simple MIDI for Virtual Organ (using Midi-Ox)



Index

Introduction

The aim of this document is to provide an understanding, and tools required to demystify the 'invisible' MIDI signals that drive a Virtual Organ, specifically the MIDI signals from your physical console to the input of Virtual Organ software. This information is generic and will be of assistance as a basic knowledge base for all Virtual Organ applications. I have chosen to use MidiOx as a freely available MIDI viewing program. There are other programs that will provide a similar function.

The technical level will be kept simple and only cover MIDI aspects specific to the main common functions. The information gained from this will be a big help for those with no MIDI experience at all, to set up a Virtual Organ, and provide a useful base for those wanting to configure more advanced console functions.

It may initially look complicated, but it is broken down into each small step, with pictures of what you need to do, and what you will see happening.

We will cover:

The basic setup of Midi-Ox as a means of viewing (otherwise invisible) MIDI signals
The difference between Hexadecimal and Decimal, how to tell which is which and convert them
The three basic MIDI functions used by Virtual Organs, how to recognise them and read them

Requirements

To use this as a tutorial you will need:-

- A keyboard or console producing MIDI out
- A MIDI interface to USB or soundcard game port input
- Software Drivers for the MIDI interface
- Midi-Ox software http://www.midiox.com/ (free download)

(MidiOx is a 32 bit program, and will function on 64 bit platforms. These notes are based on a Windows XP 32 bit installation)

To start we will assume:-

- You have at least a MIDI keyboard
- You have a MIDI interface and cable connecting the keyboard to your computer
- You have software drivers loaded for the MIDI interface
- You have the MIDI-Ox program installed on your computer

Note: Always make sure you have your MIDI interface connected and operating before starting a program that will be looking for MIDI input.

Don't start MidiOx or your Virtual Organ program till after your MIDI interface is connected.
Eg. If a USB device, its power LED is on, and you have heard the USB start-up "DING>ding" sound.

If connecting the interface for the first time, (or the first time to a different USB port) make sure the drivers are loaded properly and Windows gives the message "the device is loaded and ready to use".

Check: Do you have an indicator showing power to your interface?
Do you have an indicator showing MIDI activity when you press and release keys?
With audio on, do you hear a DING>ding when you plug in the USB plug? This is good
Do you hear a DING<dong when you plug in the USB plug? This indicates a problem
(you will need to delete the device using Windows Device Manager and install it again)

Three Basic MIDI Functions - Overview

Virtual Organ applications usually only use two or three basic Midi functions to drive all the features of the organ. They use addressing variations in these to ensure the correct function is performed when requested. First we will look at a quick overview of the functions which could also be useful for quick reference. Then we will look at each in detail.

If you look up generic MIDI information you will find reams of information covering hundreds of functions. It is good to know we only need to learn two or three.

MIDI Note ON / Note OFF

Virtual Organs use the standard Midi "Note ON / Note OFF" functions as used by all the major keyboard manufactures.

We will usually only need to verify the function and determine the Channel Number.

Example	HEX	90 3C 7F	Note On, Channel 1, Middle C, Full Velocity
	DEC	144 60 127	as above

(DEC 144 to 159, Note On – Ch 1 to 16)

MIDI Program Change

Some Virtual Organs will use the Midi "Program Change" function to trigger the console pistons driving the programmable combination action. Most commercial organ consoles with Midi Out or MIDI controller keyboards will drive this feature.

We will need to verify the function, determine the Channel Number and the Program Change Number.

Example	HEX	CF 00 - -	Program Change (Piston), Channel 16, Number 0
	DEC	207 0	as above

(DEC 192 to 207, Program Change – Ch 1 to 16)

MIDI Controller

Virtual Organs use the standard Midi "Controller" function for its volume control input (SWELL). Most Midi keyboards and organs will drive this feature. With some unique addressing changes, a second SWELL and/or a CRESCENDO can be driven using this function. It is unlikely other equipment will drive these functions without special configuration.

We will need to verify the function, determine the Channel Number, the Controller Number, and the Range of the Value (min & max).

Example	HEX	B0 07 02	Controller, Channel 1, Cont 7, Value 2 (Swell low)
	DEC	176 7 2	as above

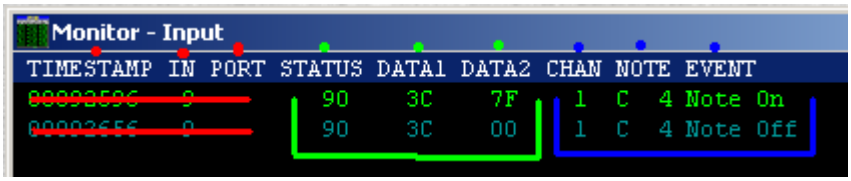
(DEC 176 to 191, Controller – Ch 1 to 16)

Three Basic MIDI Functions - Detail

MIDI Note ON / Note OFF

We will start with the MIDI Note On / Note Off messages we have seen used as the example data in Midi-Ox "Monitor – Input". It is really a lot simpler than it looks, because we can ignore a lot of what we see, and also Midi-Ox is trying to help us decipher what we are seeing.

In most cases you will only need to know the Channel number.



Ignore the TIMESTAMP, IN and PORT columns.

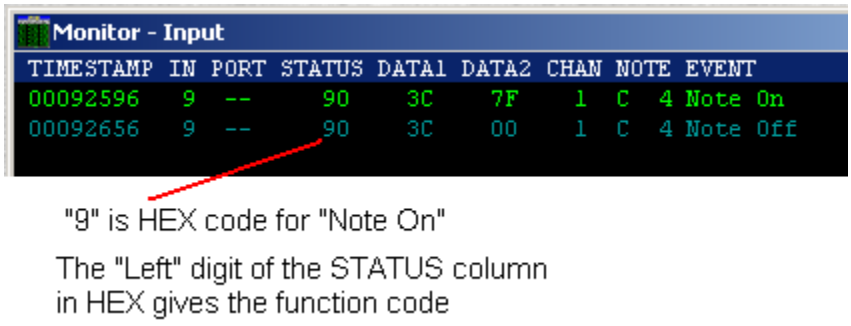
STATUS, DATA1 and DATA2 are the ones we will look at in more detail.

CHAN, NOTE and EVENT is Midi-Ox trying to help us.

As you can see in the three columns on the right, Midi-Ox is telling us what it sees in General MIDI Code. In this case, because this aspect of Virtual Organs uses the standard MIDI coding, this tells us everything we need to know.

CHAN / 1	NOTE / C 4	EVENT / Note On
CHAN / 1	NOTE / C 4	EVENT / Note Off

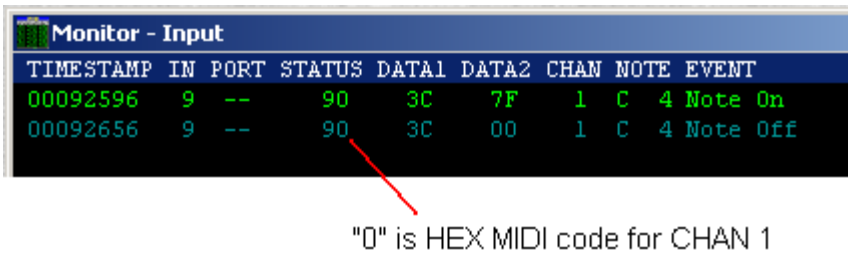
Next we need to learn how we can get this same information by learning what the STATUS, DATA1 and DATA2 columns mean. *This will also help us learn the other two MIDI function types.*



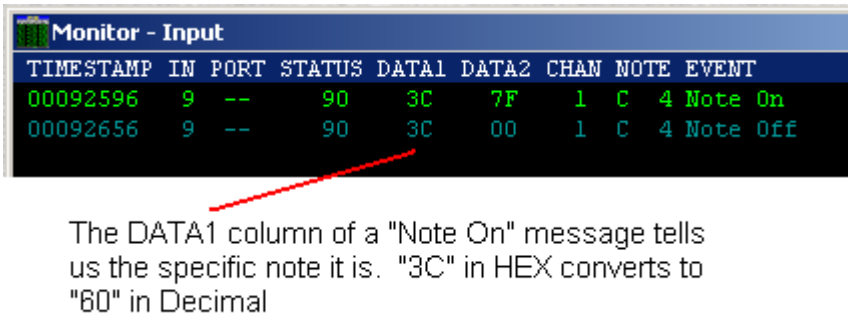
The left Status digit (in HEX) in this example is "9" this gives us the function "Note On / Note Off"

(There is actually also a HEX MIDI code "8" which is a "Note Off" function, but is rarely used. The most common application of MIDI is to use MIDI function HEX "9" with "00" velocity for "Note Off")

If this digit was a "C" it would give us the function of a Piston (MIDI Program Change)
If this digit was a "B" it would give us the function of Swell (MIDI Controller)



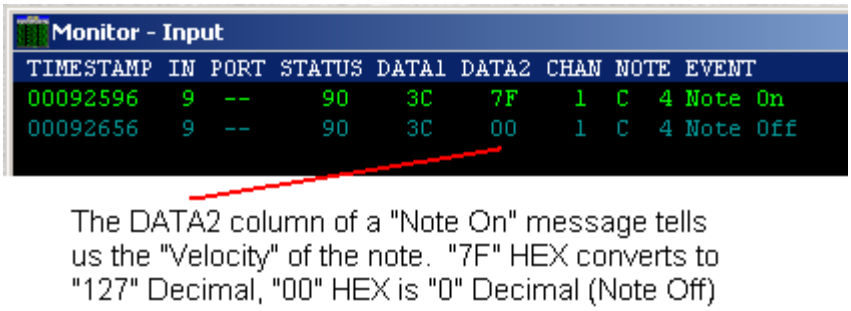
The right Status digit (in HEX) in this example is “0” this tells us the MIDI channel 1 (HEX) 0 to F - equal (DEC) Chanel numbers - 1 to 16



When using commercial keyboards or consoles with MIDI the specific notes will follow the MIDI standard. If you are adding MIDI to an old console you will need to check you are setting your MIDI system up to send the correct note numbers. See Appendix Note number lookup chart on page 15

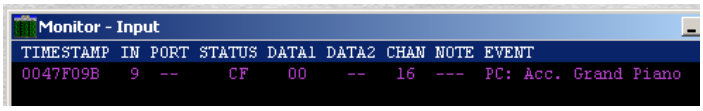
If you are using Note On/ Note Off messages for other functions like Piston control or Stops (Tabs) On and Off, you will need to make lists of which note number relates to your specific stop.

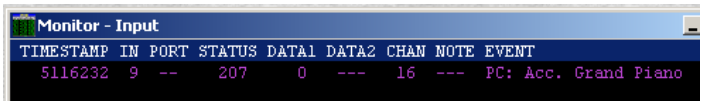
(Whilst there is a “C” in this HEX code it is just a coincidence. It is not the left character in the Status column so has no relationship to the “Program Change Function” we will look at in the next section.)



Many electronic keyboards are “Velocity” sensitive as a form of volume control or tonal change, much like a piano. There are two forms of this function. One is the initial velocity (most common) the other is after-touch velocity, or pressure. Most MIDI encoders often fitted to old electronic organs as after-market MIDI, only accept switch contacts, so have no means of measuring velocity or after-touch. These are often hard coded with “7F” as their default velocity.

MIDI Program Change – Piston Signal





The MIDI Program Change command is used by some Virtual Organ software as the trigger for "Pistons" from the combination memory.

Some Virtual Organ software will also accept MIDI Note On / Note Off to trigger this function, or in other cases any command you configure it to look for.

To get your console Pistons working with your Virtual Organ software you will need determine two things.
1/ What messages are coming from your console
2/ What messages is your Virtual Organ software looking for (or can you configure this?)

These instructions will help you identify the messages coming from your console. You will need to know

What type of MIDI message it is (HEX Status left)
What MIDI channel the messages are coming on (HEX status right [plus 1])
Which number the message is (DEC Data1).

To read the message:

Ignore the TIMESTAMP, IN and PORT columns.

STATUS and DATA1 are the ones we will look at in more detail. (DATA2 is not used with this function)

CHAN, NOTE (*not used*) and EVENT is Midi-Ox trying to help us.

Again in the three columns on the right, Midi-Ox is telling us what it sees in General MIDI Code. In this case, only the CHANnel information is useful. The rest relates to General MIDI and is not used in this form for Virtual Organ applications.

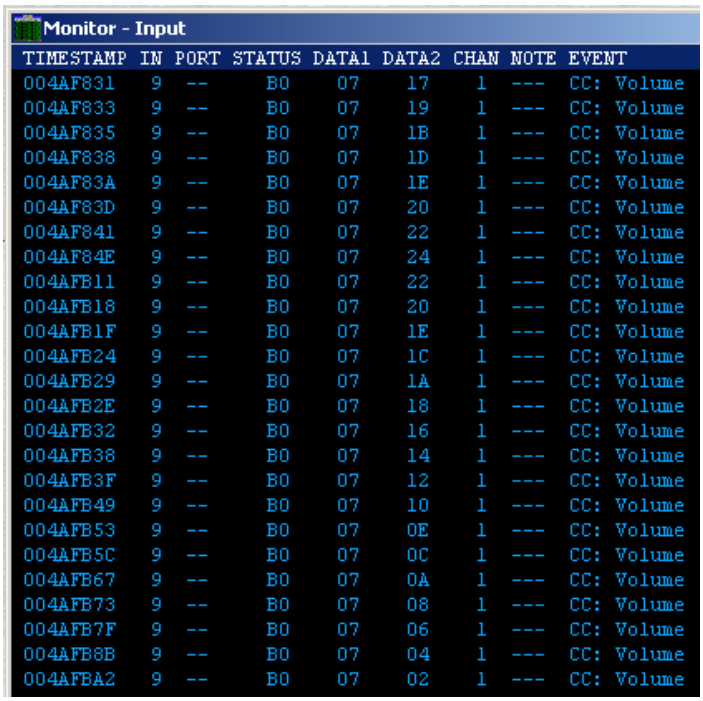
The left Status digit (in HEX) in this example is "C" this gives us the function "Program Change"

The right Status digit (in HEX) in this example is "F" giving us MIDI Channel 16

(To convert to DECimal, the HEX "CF" = DEC "207", Program Change commands can be: 192 - 207)

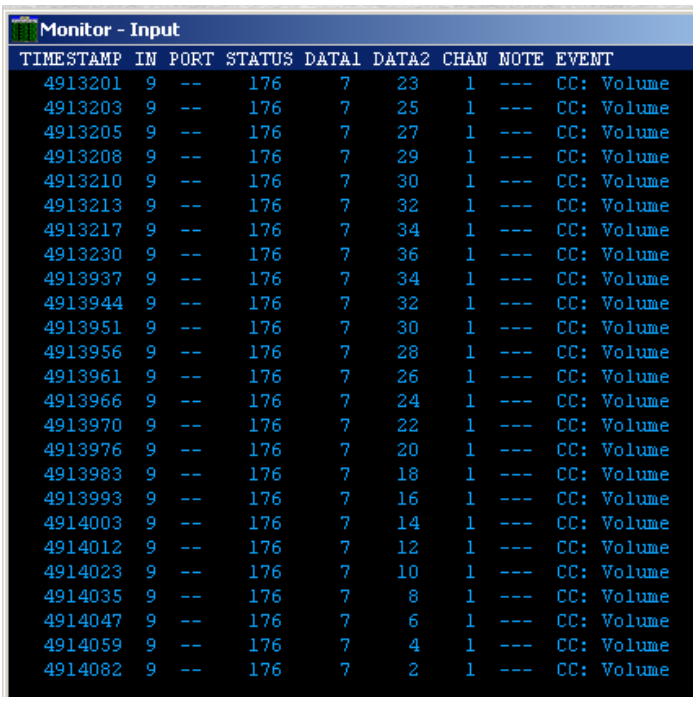
DATA1 in this example is DEC "0" so this the first of the possible 128 Program Change commands available on this MIDI channel.

MIDI Controller – Swell / Crescendo signal



Monitor - Input

TIMESTAMP	IN	PORT	STATUS	DATA1	DATA2	CHAN	NOTE	EVENT
004AF831	9	--	B0	07	17	1	---	CC: Volume
004AF833	9	--	B0	07	19	1	---	CC: Volume
004AF835	9	--	B0	07	1B	1	---	CC: Volume
004AF838	9	--	B0	07	1D	1	---	CC: Volume
004AF83A	9	--	B0	07	1E	1	---	CC: Volume
004AF83D	9	--	B0	07	20	1	---	CC: Volume
004AF841	9	--	B0	07	22	1	---	CC: Volume
004AF84E	9	--	B0	07	24	1	---	CC: Volume
004AFB11	9	--	B0	07	22	1	---	CC: Volume
004AFB18	9	--	B0	07	20	1	---	CC: Volume
004AFB1F	9	--	B0	07	1E	1	---	CC: Volume
004AFB24	9	--	B0	07	1C	1	---	CC: Volume
004AFB29	9	--	B0	07	1A	1	---	CC: Volume
004AFB2E	9	--	B0	07	18	1	---	CC: Volume
004AFB32	9	--	B0	07	16	1	---	CC: Volume
004AFB38	9	--	B0	07	14	1	---	CC: Volume
004AFB3F	9	--	B0	07	12	1	---	CC: Volume
004AFB49	9	--	B0	07	10	1	---	CC: Volume
004AFB53	9	--	B0	07	0E	1	---	CC: Volume
004AFB5C	9	--	B0	07	0C	1	---	CC: Volume
004AFB67	9	--	B0	07	0A	1	---	CC: Volume
004AFB73	9	--	B0	07	08	1	---	CC: Volume
004AFB7F	9	--	B0	07	06	1	---	CC: Volume
004AFB8B	9	--	B0	07	04	1	---	CC: Volume
004AFBA2	9	--	B0	07	02	1	---	CC: Volume



Monitor - Input

TIMESTAMP	IN	PORT	STATUS	DATA1	DATA2	CHAN	NOTE	EVENT
4913201	9	--	176	7	23	1	---	CC: Volume
4913203	9	--	176	7	25	1	---	CC: Volume
4913205	9	--	176	7	27	1	---	CC: Volume
4913208	9	--	176	7	29	1	---	CC: Volume
4913210	9	--	176	7	30	1	---	CC: Volume
4913213	9	--	176	7	32	1	---	CC: Volume
4913217	9	--	176	7	34	1	---	CC: Volume
4913230	9	--	176	7	36	1	---	CC: Volume
4913937	9	--	176	7	34	1	---	CC: Volume
4913944	9	--	176	7	32	1	---	CC: Volume
4913951	9	--	176	7	30	1	---	CC: Volume
4913956	9	--	176	7	28	1	---	CC: Volume
4913961	9	--	176	7	26	1	---	CC: Volume
4913966	9	--	176	7	24	1	---	CC: Volume
4913970	9	--	176	7	22	1	---	CC: Volume
4913976	9	--	176	7	20	1	---	CC: Volume
4913983	9	--	176	7	18	1	---	CC: Volume
4913993	9	--	176	7	16	1	---	CC: Volume
4914003	9	--	176	7	14	1	---	CC: Volume
4914012	9	--	176	7	12	1	---	CC: Volume
4914023	9	--	176	7	10	1	---	CC: Volume
4914035	9	--	176	7	8	1	---	CC: Volume
4914047	9	--	176	7	6	1	---	CC: Volume
4914059	9	--	176	7	4	1	---	CC: Volume
4914082	9	--	176	7	2	1	---	CC: Volume

The MIDI Controller command is used by Virtual Organ software as the console signals for Swell and Crescendo control. This command allows the continuous variation between 128 levels, so is ideally suited to the control of Swell (or volume). This type of control is sometimes referred to as "analogue' control. *(The other functions we have looked are either on or off, therefore referred to as "digital" control, or only having two states.)*

To get your console Swell and/or Crescendo functions working with your Virtual Organ software, you will need to follow a very similar strategy to what we discussed for the MIDI Program Change to setup Pistons. Following are the specifics relating to the MIDI Controller command.

What type of MIDI message is it (HEX Status left)
What MIDI channel the messages are coming on (HEX status right [plus 1])
Which number is the message (DEC Data1)
What is the range of the messages (DEC Data2)

To read the message:

Ignore the TIMESTAMP, IN and PORT columns

STATUS, DATA1 and DATA2 are the ones we will look at in more detail.

CHAN, NOTE (*not used*) and EVENT is Midi-Ox trying to help us.

Again in the three columns on the right, Midi-Ox is telling us what it sees in General MIDI Code. In this case, only the CHANnel information is useful. The rest relates to General MIDI and is not used in this form for Virtual Organ applications. (you can see though that CC-7 is common in MIDI as Volume)

The left Status digit (in HEX) in this example is “B” this gives us the function “Controller”

The right Status digit (in HEX) in this example is “0” giving us MIDI Channel 1

(To convert to DECimal, the HEX “B0” = DEC “176”, Controller commands can be: 176 – 191)

DATA1 in this example is DEC “7” so this is referred to Controller 7 or CC7.
(In Virtual Organ software you will also see CC11 [often a second swell] and CC1 [often crescendo])

DATA2 is the range of the output. As you move your swell pedal you will see these numbers go up and down. *It is important to note the Minimum and Maximum values sent by your pedal(s) as you will often need to enter these values into the software so you get the full travel effect in the software.*

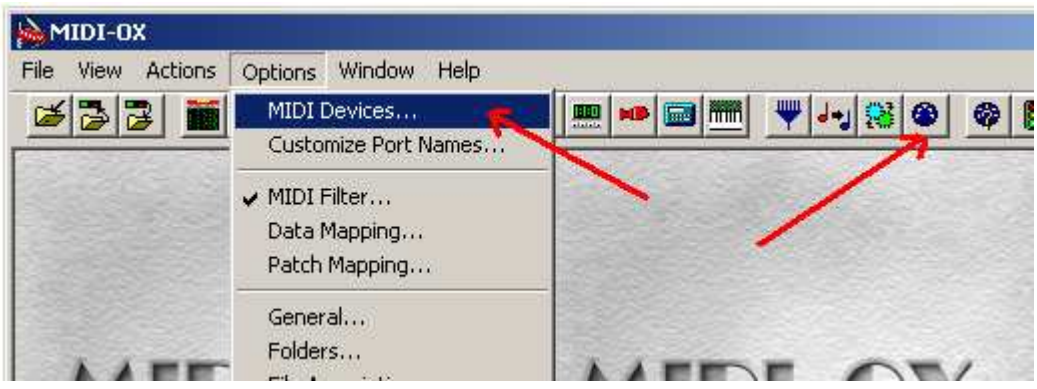
MidiOx Setup Steps

MIDI device, interface and cables connected

Start MIDI-OX software

Click the Blue DIN plug symbol (or Options, MIDI Devices…)

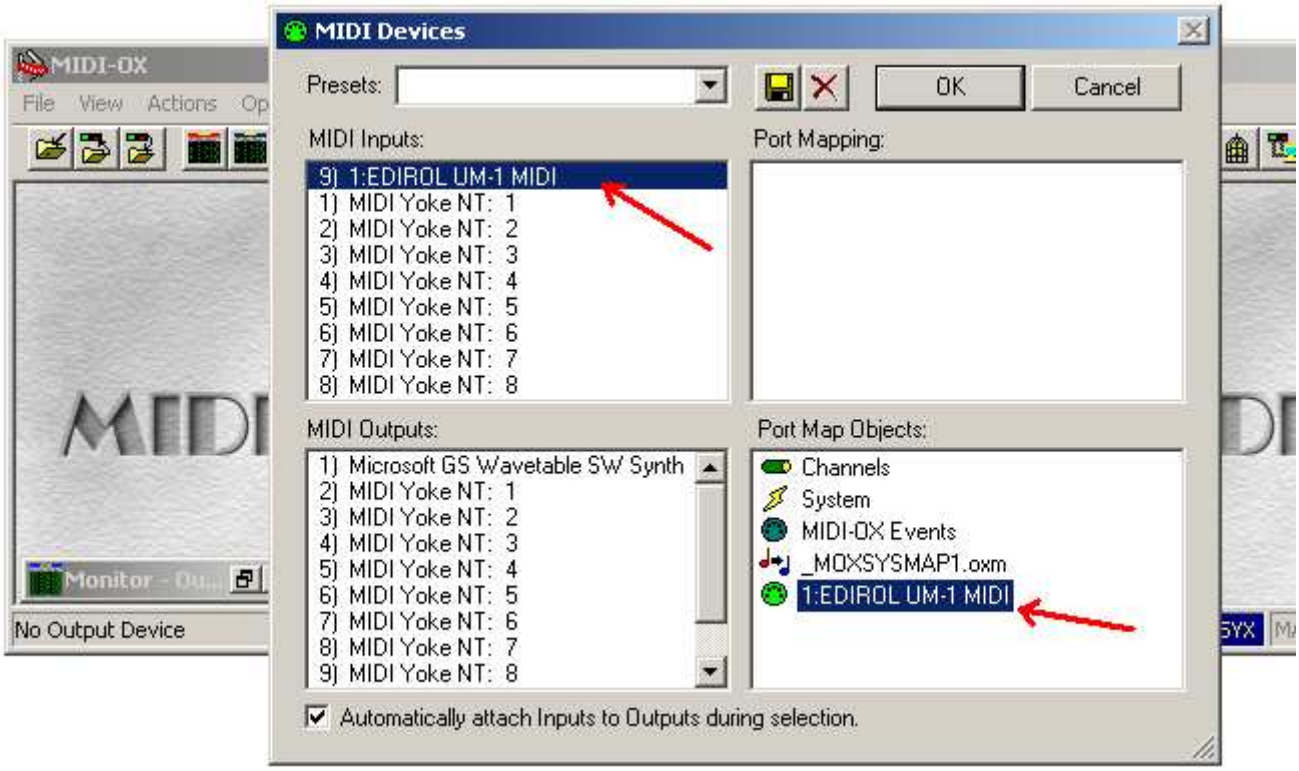
There is also a Green DIN plug symbol which will display a "Port Activity Monitor". Don't get them confused. (You can run the port activity monitor as well if you like.)



In the "MIDI inputs:" window, select the MIDI interface device that you have your keyboard connected to.

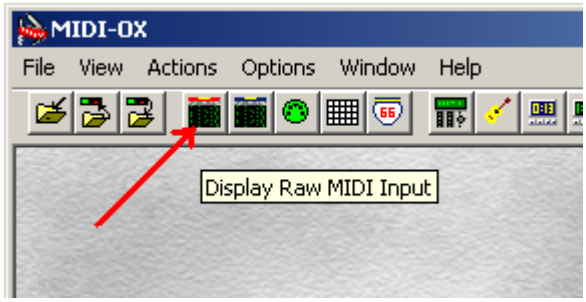
Click on your MIDI interface item. It will now show up in the "Port Map Objects:" window.

{In my case it is item 9) 1:EDIROL UM-1 MIDI}

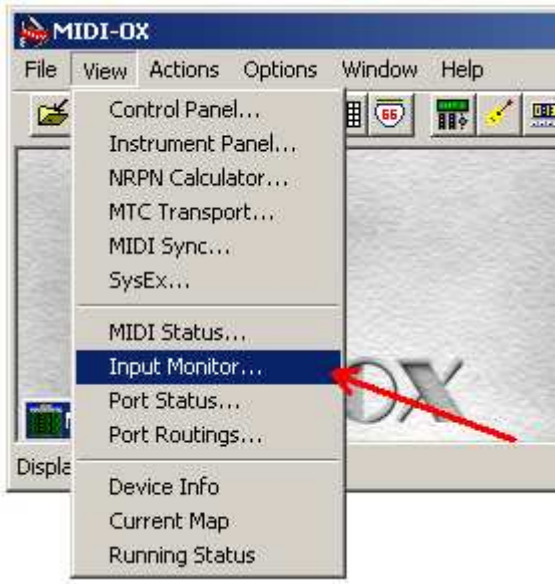


Click OK to close the "MIDI devices" window.

"1 Input Device" will now be displayed at the bottom of your Midi-Ox display.

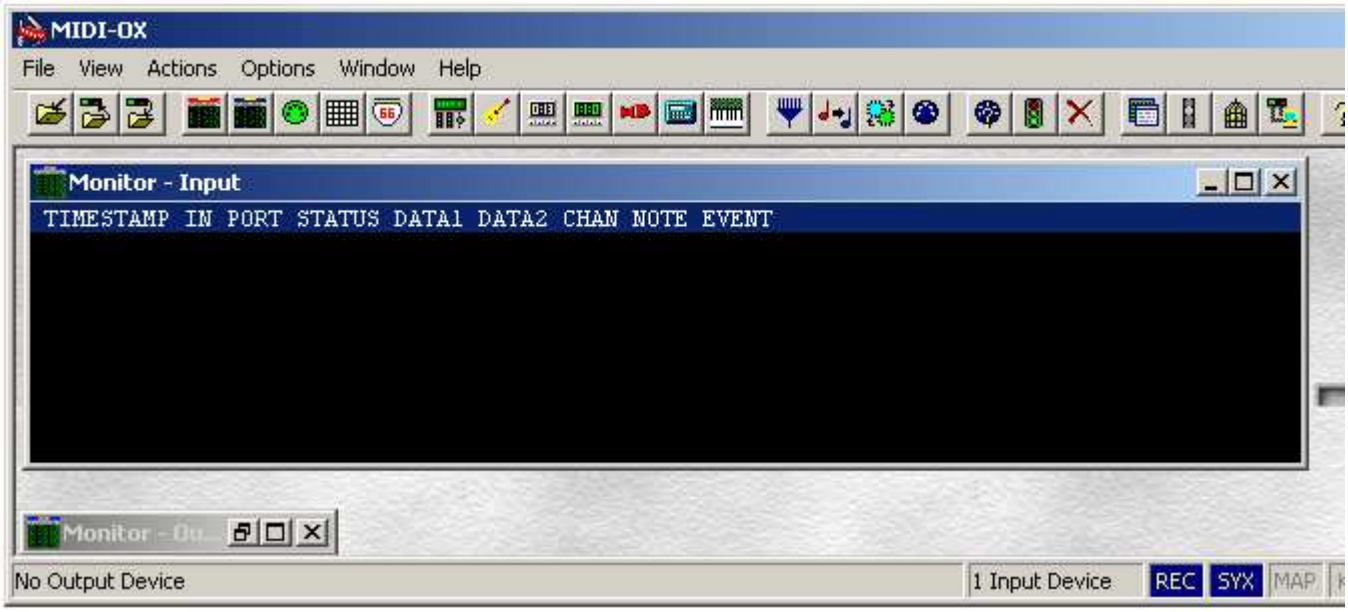


Click the red monitor icon, “Display Raw MIDI Input”.



Or, View, “Input Monitor…”

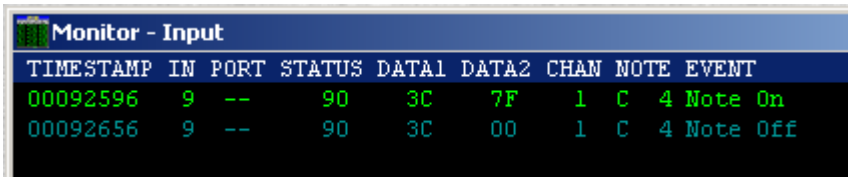
This brings up the “Monitor – Input” window.



Now, press and release a note on your keyboard.

(In this case I have pressed middle C on a keyboard sending MIDI on channel 1)

This is what you should see:-

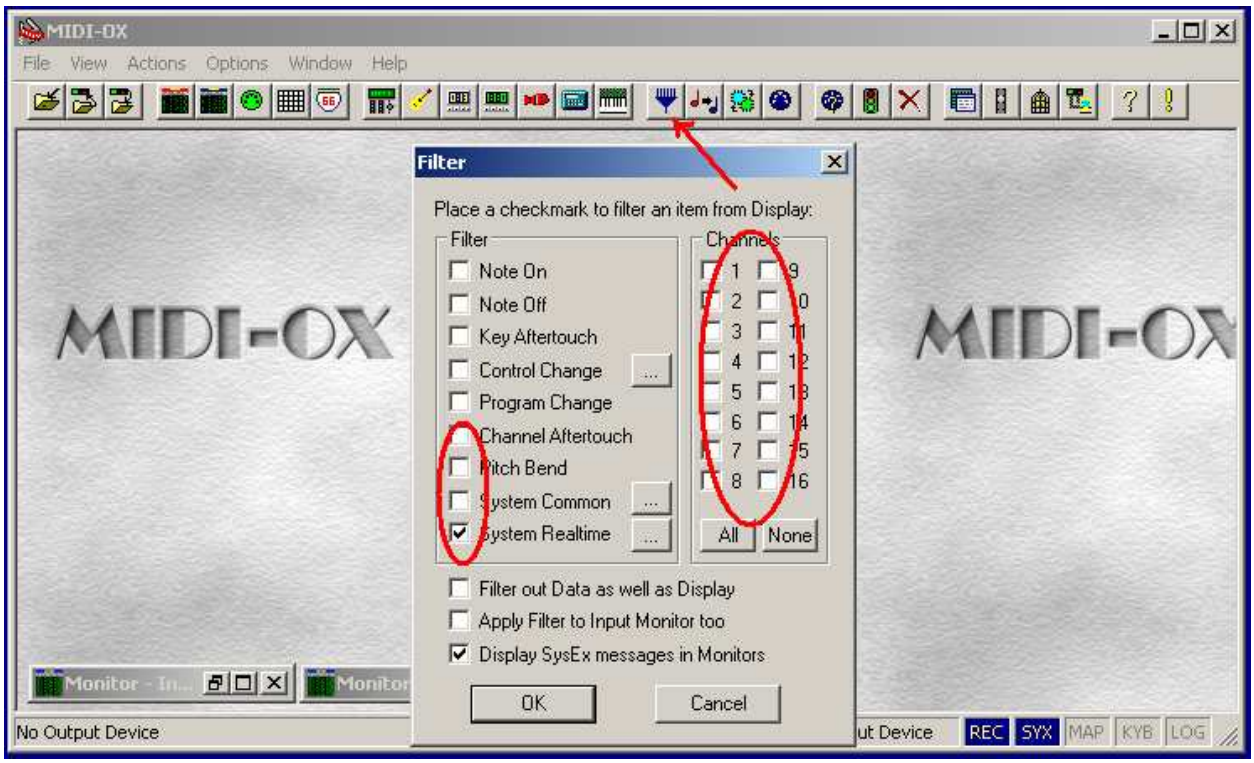


If you are with me to this point, you now have the capability to view MIDI signals coming from your keyboard or console.

Filter – Continuously streaming messages

In some cases you may not be able to see the simple messages as shown above. You may be seeing streams of MIDI data coming, even if you are not pressing a note or piston or swell control.
If this is the case, your MIDI device is sending extra messages that are not required for our purpose of seeing and understanding the simple common MIDI messages.

MidiOx has a mechanism to stop this. It is a "Filter"



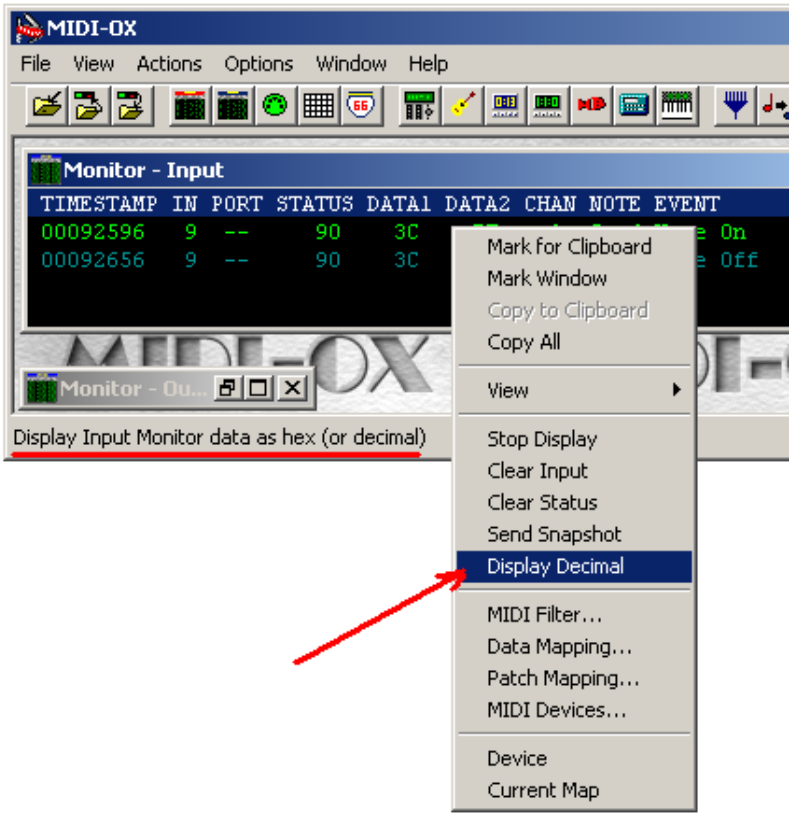
System Realtime and System Common are most likely to be the messages we don't want to see.
Experiment with settings for your system to get the minimal view required. Also see extra settings under the "…" buttons.
(Be careful with what you select NOT to see as it could hide messages you DO need to see at some stages)

HEXadecimal and DECimal

At this point we need to understand that MIDI messages can be written in HEX (Hexadecimal) or in DEC (Decimal). You need to be able to recognise which is which, and keep a conversion chart handy.
See Appendix page 15.

Midi-Ox is able to display either one or the other, and again you need to be able to recognise which is which and know how to change from one to the other.

With your mouse pointer in the black area of the "Monitor – Input", Right Click your mouse.
You will see the following. Scroll down with your mouse to highlight "Display Decimal".

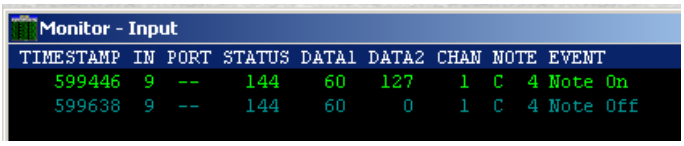


You will also notice the help text in the bottom left of the Midi-Ox window.

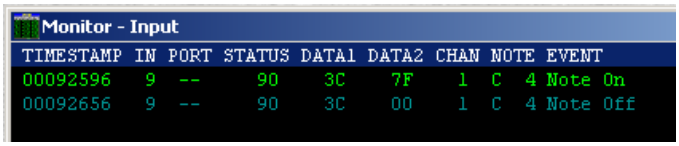
Click "Display Decimal", you will now see the display on the left.

(For reference I have put the same note data as we sore before in HEX beside it to the right.)

DECimal



HEXadecimal



Notice now that the STATUS is a three digit number.
Now "144" where before it was a two digit number "90".

In HEX the STATUS, DATA1 and DATA2 will only ever be two digit numbers and will sometimes have letters with them A to F (like the "C" in "3C" and the "F" in "7F" in our first Monitor – Input display).

In DEC these values may be three digits (like the "144" in our second Monitor- Input display) but could still be two digits, however will never have letters with them.

I believe it is easier to learn and recognise the MIDI STATUS codes in HEX, but easier to read the DATA1 and DATA2 values in DEC.

In HEX, the MIDI status has a left and a right digit. The left digit tells us the "TYPE" of MIDI message and the right digit tells us the MIDI "CHANNEL". We will go into more detail below.

In DEC the number displayed for STATUS is the DECimal conversion of the two HEX digits. The number displayed now gives no clue as to either the type of function or the channel without looking up a reference chart.

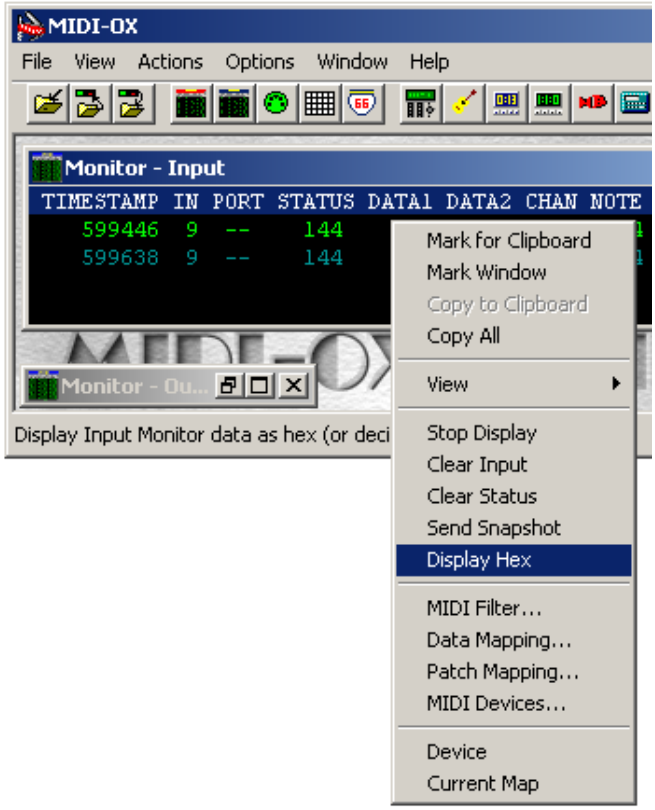
Most Virtual Organ applications will show their Midi references in DEC as they mainly relate to DATA1 and DATA2 values. You will just need to recognise the difference, keep the conversion chart handy, and either check or change Midi-Ox to display the information in the format you need.

To check or change Midi-Ox back and forth, right click in the black area of the "Monitor – Input".

Scroll down to "Display Xxx".

This item will toggle. If it says "Display Hex", then you are currently viewing "Decimal" and it will change to HEX if you click it.

If it says "Display Decimal", you would be currently viewing "HEX" and it would change to "Decimal" if you clicked it.



Advanced Functions

MidiOx is a powerful program in its own right. There are many more advanced functions that can be used if you have a need for them. In most cases you will then need to connect the output of MidiOx to your other MIDI software ie. Virtual Organ. This is done using a "Virtual MIDI cable". You will need to download a companion program of MidiOx called MidiYoke. This becomes a MIDI patch cable that links the MIDI signals between programs running on your computer. (MidiOx is a 32 bit program only. LoopMIDI or LoopBe1 are options for 64 bit operating systems.)

PC Keyboard to play virtual instrument

With MidiOx you are able to set up so you can play music notes using your computer keyboard. This can be useful for some virtual instrument testing if you don't have a keyboard with you.

Data Mapping – or Translation

MidiOx has a powerful function called "Data Mapping" or "Translation". This allows the recognition of a specific input message and have it "Mapped" or "Translated" to a different specific message you nominate.

This feature could be useful if you have messages coming from your console that are not recognised by your Virtual Organ software (or it is not possible to reconfigure the Virtual Organ software to recognise your console signals).

It is beyond the scope of this document to go into any detail on this, however the MidiOx Help files should be useful based on your newfound understanding and ability to read MIDI signals.

MIDIBar – MIDI Recorder / Player

MIDIBar is a useful MIDI recorder function. This can capture in real time all the signals coming from your console on their way to your virtual organ program, then play them back whenever you want. You can "record" yourself playing a song, or maybe some mundane console test functions.

Appendix

HEXadecimal to DECimal conversion chart for MIDI

HEX	DEC	HEX	DEC	HEX	DEC	HEX	DEC	HEX	DEC	HEX	DEC	HEX	DEC	HEX	DEC
00	0	10	16	20	32	30	48	40	64	50	80	60	96	70	112
01	1	11	17	21	33	31	49	41	65	51	81	61	97	71	113
02	2	12	18	22	34	32	50	42	66	52	82	62	98	72	114
03	3	13	19	23	35	33	51	43	67	53	83	63	99	73	115
04	4	14	20	24	36	34	52	44	68	54	84	64	100	74	116
05	5	15	21	25	37	35	53	45	69	55	85	65	101	75	117
06	6	16	22	26	38	36	54	46	70	56	86	66	102	76	118
07	7	17	23	27	39	37	55	47	71	57	87	67	103	77	119
08	8	18	24	28	40	38	56	48	72	58	88	68	104	78	120
09	9	19	25	29	41	39	57	49	73	59	89	69	105	79	121
0A	10	1A	26	2A	42	3A	58	4A	74	5A	90	6A	106	7A	122
0B	11	1B	27	2B	43	3B	59	4B	75	5B	91	6B	107	7B	123
0C	12	1C	28	2C	44	3C	60	4C	76	5C	92	6C	108	7C	124
0D	13	1D	29	2D	45	3D	61	4D	77	5D	93	6D	109	7D	125
0E	14	1E	30	2E	46	3E	62	4E	78	5E	94	6E	110	7E	126
0F	15	1F	31	2F	47	3F	63	4F	79	5F	95	6F	111	7F	127

HEX	DEC	HEX	DEC	HEX	DEC	HEX	DEC	HEX	DEC	HEX	DEC	HEX	DEC	HEX	DEC
80	128	90	144	A0	160	B0	176	C0	192	D0	208	E0	224	F0	240
81	129	91	145	A1	161	B1	177	C1	193	D1	209	E1	225	F1	241
82	130	92	146	A2	162	B2	178	C2	194	D2	210	E2	226	F2	242
83	131	93	147	A3	163	B3	179	C3	195	D3	211	E3	227	F3	243
84	132	94	148	A4	164	B4	180	C4	196	D4	212	E4	228	F4	244
85	133	95	149	A5	165	B5	181	C5	197	D5	213	E5	229	F5	245
86	134	96	150	A6	166	B6	182	C6	198	D6	214	E6	230	F6	246
87	135	97	151	A7	167	B7	183	C7	199	D7	215	E7	231	F7	247
88	136	98	152	A8	168	B8	184	C8	200	D8	216	E8	232	F8	248
89	137	99	153	A9	169	B9	185	C9	201	D9	217	E9	233	F9	249
8A	138	9A	154	AA	170	BA	186	CA	202	DA	218	EA	234	FA	250
8B	139	9B	155	AB	171	BB	187	CB	203	DB	219	EB	235	FB	251
8C	140	9C	156	AC	172	BC	188	CC	204	DC	220	EC	236	FC	252
8D	141	9D	157	AD	173	BD	189	CD	205	DD	221	ED	237	FD	253
8E	142	9E	158	AE	174	BE	190	CE	206	DE	222	EE	238	FE	254
8F	143	9F	159	AF	175	BF	191	CF	207	DF	223	EF	239	FF	255

HEX MIDI note number lookup chart.

Octave#	C	C#	D	D#	E	F	F#	G	G#	A	A#	B
2	24	25	26	27	28	29	2A	2B	2C	2D	2E	2F
3	30	31	32	33	34	35	36	37	38	39	3A	3B
4	3C	3D	3E	3F	40	41	42	43	44	45	46	47
5	48	49	4A	4B	4C	4D	4E	4F	50	51	52	53
6	54	55	56	57	58	59	5A	5B	5C	5D	5E	5F
7	60											