BLAGSHEET MIDI #07

By courtesy of The Crow Hill Company

AES Convention: 70 (October 1981)
Paper Number: 1845
Authors: Smith, Dave; Wood, Chet
Affiliation: Sequential Circuits, Inc., San Jose, CA

Publication Date: October 1, 1981
Subject: Electronic Music and Musical Instruments

THE UNIVERSAL SYNTHESIZER INTERFACE

The Universal Synthesizer Interface is a specification designed to enable interconnecting synthesizers, sequencers and home computers with an industry-wide standard interface. This is a preliminary specification; comments, criticism, and alternative proposals are welcome. This interface specification has not been tested and would need to be retrofitted to any equipment presently in the field. The interface is basically specified as one-to-one between two units; ie, a synthesizer and a sequencer. Under certain circumstances, however, more units may be placed on a single line.

HISTORY

MIDI is a rare occasion where technologists come together to better legitimise their operations, increase the success of their businesses. By burying rivalries, or their desire to jump on competitive advantages create an opportunity for global culture to experience a paradigm shift. No invention since music notation has had a greater impact on the history of music. Something that each and every one of us has experienced without thinking about on a day to day basis now that we live predominantly in the box. But it's still there and we would sorely miss it if it went away.

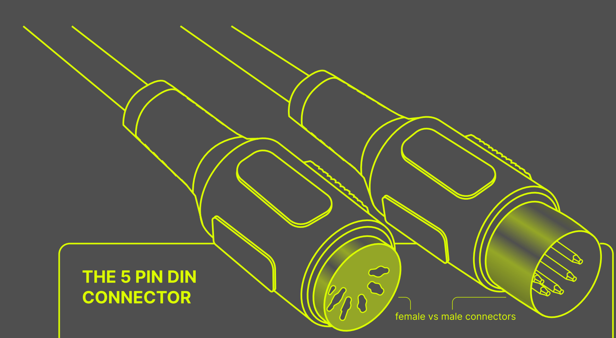
The excerpt above is a response to Roland founder Ikutaro Kakehashi's belief shared with fellow manufacturers Oberheim, Sequential Circuits, Yamaha, Korg and Kawai. That an inability to synchronise electronic musical devices made by different manufacturers was hindering the growth of the music-tech industry. The paper outlines a protocol defined by Dave Smith and Chet Wood based on different historical attempts to get kit to communicate. From the days of CV-gate, through a fledgling but (as Kakehashi lamented) massively complex system of Smith's devising settling on Roland's own DCB innovation as a basis.

The agreed protocol was dubbed Universal Musical Interface, or UMI (pronounced you-me) until Dave Smith aired his displeasure with a name so "corny" for something so serious. Whereby MIDI, musical instrument, digital interface was settled upon. Smith demonstrated MIDI for the first time at the 1983 NAMM convention. This ushered in the formation of the MMA (Midi, Manufacturers Association) from which the protocol was agreed upon and MIDI 1.0 was born.

WHAT IS IT?

MIDI is a digital protocol that allows two or more musical devices to communicate between each other using a very basic 8bit binary code. Whilst the transmitter and receiver inevitably are devices used for the creation of sound and music, it is truly a protocol for a microprocessor to encode and another to decode. It contains no sonic information and has often been compared to a "piano roll" found in pianola automatic pianos. Some greats of the art of piano performances and compositions have created piano rolls so their performances can be recreated. Not through some form of audio recording, or notation but an input that a pianola responds to mechanically. Whilst this explains the principal well, it doesn't truly respect the thinking behind the protocol which relates more to Morse as a coded form of music-communication that had no physical kinetic connection with its creation.

This blagsheet has two purposes. One as a reference document but also we believe that information is better retained when the concept and indeed context behind it is better understood. We hope it is a help to you in both respects.



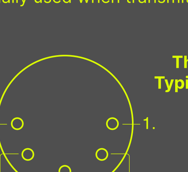
THE 5 PIN DIN CONNECTOR

Enabling consumers to connect turntables, tape machines, tuners with amplifiers as part of "separates systems". The Din plug was the preferred choice of European electrical manufacturers in the mid 20th century. Initially starting as a 3 pin connector for mono sources. The 5 pin was adopted to accommodate stereophonic signals and was standardised by the ***D**eutsches **I**nstitut für **N**ormung* (which gives the cable it's name) in the 1950s.

It's use outside Europe had not been as common and it was largely discarded in the 1980s in favour of the more universally adopted RCA cable.

When negotiations were taking place between Japanese and American corporations regarding what connectors to adopt for use in transmission of USI (or MIDI as it became known). It was felt that the connections afforded by TR and TRS cables wouldn't offer sufficient grounding of the electrical current and would result in interference.

With warehouses full of these now defunct cables it was suggested the 5 pin din could be adopted as it's shielding, grounding and availability made it an excellent candidate. It is because of this make-do solution that only 3 of the 5 pins is actually used when transmitting MIDI data.



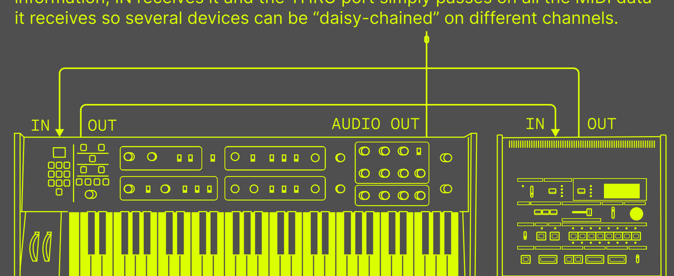
The 5 Pin DIN... Typical Use In MIDI.

- **Pin 2** (Shield): This pin serves as the ground for the cable, providing a return path for the signal and reducing noise.
- **Pin 4** (Current Source): This pin typically carries the positive MIDI signal.
- **Pin 5** (Current Sink): This pin typically carries the negative MIDI signal.
- **Pin 1 & Pin 3** (Unused): These pins are not used for standard MIDI communication, although they might be utilized by specific devices or applications.

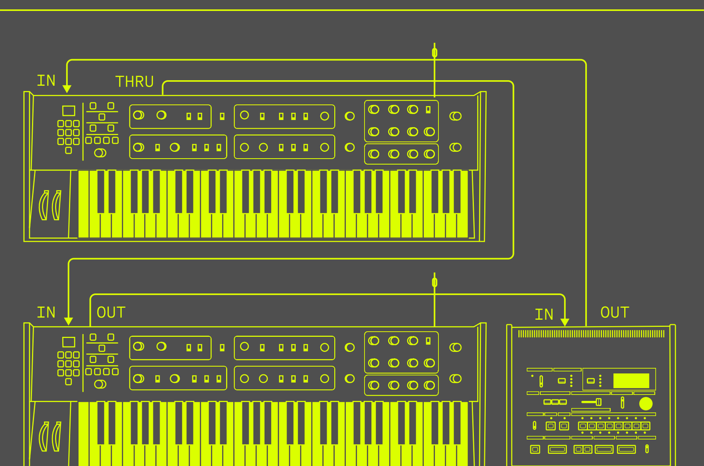
The MIDI cable is a unidirectional one, so outputs (OUT & THRU) always need to be connected to inputs (IN).

IN, OUT & THRU

Midi allows a number of devices to be connected together, performed and sequenced by one and other. It uses dedicated "channels" to send and receive on. However with only one cable, and a uni-directional one at that. Different ports are required to communicate correctly. The OUT port sends MIDI information, IN receives it and the THRU port simply passes on all the MIDI data it receives so several devices can be "daisy-chained" on different channels.



In this example the performer plays MIDI into the sequencer from a MIDI synthesizer to record. The sequencer then transmits the MIDI information back to the synthesizer to play back the performance from the sound being generated in the keyboard. The keyboard outputs that sound to a mixer, amplifier or pair of cans independently of the sequencer.



In this example the performer wants to play MIDI into the sequencer via the same keyboard but layer two different sound sources on top of each other on separate sequencer tracks. This achieved by setting each synthesizer to its own unique channel (1-16). In this instance the sequencer then plays out both channels via it's MIDI out. The second (upper pictured) keyboard follows the transmission on it's own channel ignoring anything that isn't. However it also passes on all the MIDI data being transmitted via it's thru port. This then plays keyboard 1. according to the channel it is set to.

WHY IS MIDI SO "QUIRKY".

8 different types of MIDI messages, 16 Channels, 0-127 values counted in 128 steps. Why???

Midi is a stream of data traditionally sent through an analogue audio-grade electric cable. In order to understand what could be percieved as "quirks" of MIDI one must first understand how data is transmitted via copper wire.

Binary data is sent over a continuous flowing current by differences in voltage; 0v for 0 for example 5v for 1. Each fluctuation is referred to as a bit. And the rate that a cable can transmit these bits is called the "baud rate" (a baud rate of 31250 bps (bits per second) ws agreed amongst the MMA).

It does this by grouping a series of voltage fluctuations into bytes. MIDI is an 8bit system referring to the number of bits in a byte. In order to deliver all the neccessary information without cramming the cable with too much data and therefore electrical lag that could impede a musical performance it was decided that each MIDI "message" was to be made up of 3 bytes. With byte 1 describing the type of message and bytes 2 and 3 giving two independent values. All deliverable in 320 microseconds.

An actual depiction of a MIDI message as seen through an oscilloscope.

MIDI speification determines two basic types of message bytes: the **status byte** and the **data byte**. The **M**ost **S**ignificant **B**it (MSB - or, first bit of a byte!) of a **status** byte is always 1. The MSB of a **data** byte is always 0. Lets first establish how a **status** byte works...

The status byte is split into 'nibbles' (these tech-bros and their sense of humour).

Take this byte:

```
MSB  TYPE     VALUE
10000010
1st NIBBLE  2nd NIBBLE
```

As the byte's first bit (its MSB) starts with a 1, this declares it's a status byte. This is then combined with the next 3 bits to give 8 possible outcomes numbered 1000-1111 in binary = 8-15 in decimal. The next nibble has 16 possible outcomes 0000-1111 = 0-15. Giving us a total of 16.

With the 8 possible status types, the agreed MIDI protocol is that the first 7 are channel dependent

BINARY	DECIMAL	TYPE	2nd NIBBLE DATA VALUE
1000	8	Note Off	xxxx = channel number
1001	9	Note On	xxxx = channel number
1010	10	Polyphonic Aftertouch	xxxx = channel number
1011	11	Control Change	xxxx = channel number
1100	12	Program Change	xxxx = channel number
1101	13	Channel Aftertouch	xxxx = channel number
1110	14	Pitch Wheel	xxxx = channel number
1111	15	System	xxxx = system type value

With the 8th type (status type 1111 or 15) being non-channel-dependent and the next 4 bits determining what type of system status message this is (sequence start/stop, MTC etc).

SO, we can finally answer the three questions posited above. MIDI has 8 types of message because it has 4 bits to work with where the first is always 1. As channel dependent messages have another 4 bits to state their channel No° there are 16 possible outcomes. 0000-1111 = 0-15 = 16 MIDI Channels.

SO, with the second two BYTES in any given MIDI message being purely DATA each of which with 8 bits to work with, but where the first bit is always 0 (to determine that it is a DATA byte *not* STATUS) You have outcomes from 00000000-011111111 = 0-127 = 128 possible values - agreed as these:

STATUS TYPE	DATA 1	DATA 2
Note Off	Note Number	Velocity
Note On	Note Number	Velocity
Polyphonic Aftertouch	Note Number	Pressure
Control Change (CC)	Controller Number	Data / Value
Program Change	Program Number	Unused
Channel Aftertouch	Pressure	Unused
Pitch Wheel	LSB (from)	MSB (to)

In this next example we demonstrate what a command to play a note, transmit it on channel 1, make that note middle C (note number 60) and play it at a velocity of 120 looks like:

```
     MSB  TYPE   CHANNEL   LSB   DATA 1    LSB   DATA 2
  __1tttnnnn_0xxxxxxx_0xxxxxxx__
MIDI      BYTE 1 = STATUS    BYTE 2 = DATA 1    BYTE 3 = DATA 2
MESSAGE
  __10010001_00111100_01111000__
       9     01        60          120
 = NOTE ON_CHANNEL 1_MIDDLE C_VELOCITY 120
```

We hope now MIDI seems less arbitrary and totally pragmatic based on the 8bit technology available at the time. But also the desire to make something that was as musical as it was technical. Whilst few of us break out a 5 pin DIN these days, electing instead for the far superior USB connections. Don't be fooled by it no-longer being MIDI. It is still the exact same set of packages or messages being delivered in binary form from one microprocessor to another.

If there is one quirk of MIDI it is an historical one. So much emphasis in so few possibilities given to aftertouch! Particularly polyphonic aftertouch. Was it because YAMAHA had invested so much faith in their CS80 perhaps? But also, looking at the General MIDI list above. Of 128 possible instruments 2 are surrendered to slap bass sounds, and one to a gun shot? Poor bass player.

GENERAL MIDI (GM)

Developed by the MMA and launched in 1991 this was the second MIDI revolution that never really materialised. Simply put it is an agreed list of 128 instruments and sounds that could offer a universal collaborative opportunity by standardising program change events when sharing MIDI files. It never really caught on largely down to technology at the time not having the capabilities to create affordable sound-devices that could cater to the entire list without sonic compromise.**

Keyboards 0=Acoustic Grand Piano 1=Bright Acoustic Piano 2=Electric Grand Piano 3=Honky-tonk Piano 4=Rhodes Piano 5=Chorused Piano 6=Harpsichord 7=Clavinet	Reeds 64=Soprano Sax 65=Alto Sax 66=Tenor Sax 67=Baritone Sax 68=Oboe 69=English Horn 70=Bassoon 71=Clarinet 72=Piccolo
Tuned Percussion 8=Celesta 9=Glockenspiel 10=Music Box 11=Vibraphone 12=Marimba 13=Xylophone 14=Tubular Bells 15=Dulcimer	Winds 73=Flute 74=Recorder 75=Pan Flute 76=Bottle Blow 77=Shakuhachi 78=Whistle 79=Ocarina
Organs 16=Hammond Organ 17=Percussive Organ 18=Rock Organ 19=Church Organ 20=Reed Organ 21=Accordion 22=Harmonica 23=Tango Accordion	Synth Leads 80=Lead 1 (square) 81=Lead 2 (sawtooth) 82=Lead 3 (calliope lead) 83=Lead 4 (chiff lead) 84=Lead 5 (charang) 85=Lead 6 (voice) 86=Lead 7 (fifths) 87=Lead 8 (bass + lead)
Guitars 24=Acoustic Guitar (nylon) 25=Acoustic Guitar (steel) 26=Electric Guitar (jazz) 27=Electric Guitar (clean) 28=Electric Guitar (muted) 29=Overdriven Guitar 30=Distortion Guitar 31=Guitar Harmonics	Synth Pads 88=Pad 1 (new age) 89=Pad 2 (warm) 90=Pad 3 (polysynth) 91=Pad 4 (choir) 92=Pad 5 (bowed) 93=Pad 6 (metallic) 94=Pad 7 (halo) 95=Pad 8 (sweep)
Basses 32=Acoustic Bass 33=Electric Bass (finger) 34=Electric Bass (pick) 35=Fretless Bass 36=Slap Bass 1 37=Slap Bass 2 38=Synth Bass 1 39=Synth Bass 2	Synth Effects 96=FX 1 (rain) 97=FX 2 (soundtrack) 98=FX 3 (crystal) 99=FX 4 (atmosphere) 100=FX 5 (brightness) 101=FX 6 (goblins) 102=FX 7 (echoes) 103=FX 8 (sci-fi)
Strings 1 40=Violin 41=Viola 42=Cello 43=Contrabass 44=tremolo Strings 45=Pizzicato Strings 46=Orchestral Harp 47=Timpani	Folk Instruments 104=Sitar 105=Banjo 106=Shamisen 107=Koto 108=Kalimba 109=Bagpipe 110=Fiddle 111=Shanai
Strings 2 48=String Ensemble 1 49=String Ensemble 2 50=SynthStrings 1 51=SynthStrings 2 52=Choir Aahs 53=Voice Oohs 54=Synth Voice 55=Orchestra Hit	Percussion 112=Tinkle Bell 113=Agogo 114=Steel Drums 115=Woodblock 116=Taiko Drum 117=Melodic Tom 118=Synth Drum 119=Reverse Cymbal 120=Guitar Fret Noise
Brass 56=trumpet 57=trombone 58=Tuba 59=Muted trumpet 60=French Horn 61=Brass Section 62=Synth Brass 1 63=Synth Brass 2	Sound Effects 121=Breath Noise 122=Seashore 123=Bird Tweet 124=Telephone Ring 125=Helicopter 126=Applause 127=Gunshot

** There is one place where GM is still very much live and kicking. Download a MIDI file on a mac and play it back in the finder. Turns out Apple has a bank of GM sounds as part of its operating system.

CONTROLLER CHANGE KEY

Control Change (CC) No°	Use Protocol	Value(s)	Notes
MIDI CC 0	Bank Select (MSB)	0-127	Allows user to switch bank for patch selection. Program change used with Bank Select. MIDI can access 16,384 patches per MIDI channel.
MIDI CC 1	Modulation Wheel (MSB)	0-127	Generally this CC controls a vibrato effect (pitch, loudness, brighness). What is modulated is based on the patch.
MIDI CC 2	Breath Controller (MSB)	0-127	Oftentimes associated with aftertouch messages. It was originally intended for use with a breath MIDI controller in which blowing harder produced higher MIDI control values. It can be used for modulation as well.
MIDI CC 3	Undefined (MSB)	0-127	
MIDI CC 4	Foot Pedal (MSB)	0-127	Often used with aftertouch messages. It can send a continuous stream of values based on how the pedal is used.
MIDI CC 5	Portamento Time (MSB)	0-127	Controls portamento rate to slide between 2 notes played subsequently.
MIDI CC 6	Data Entry (MSB)	0-127	Controls Value for NRPN or RPN parameters.
MIDI CC 7	Volume (MSB)	0-127	Controls the volume of the channel.
MIDI CC 8	Balance (MSB)	0-127	Controls the left and right balance, generally for stereo patches. A value of 64 equals the center.
MIDI CC 9	Undefined (MSB)	0-127	
MIDI CC 10	Pan (MSB)	0-127	Controls the left and right balance, generally for mono patches. A value of 64 equals the center.
MIDI CC 11	Expression (MSB)	0-127	Expression is a percentage of volume (CC7).
MIDI CC 12	Effect Controller 1 (MSB)	0-127	Usually used to control a parameter of an effect within the synth or workstation.
MIDI CC 13	Effect Controller 2 (MSB)	0-127	Usually used to control a parameter of an effect within the synth or workstation.
MIDI CC 14	Undefined (MSB)	0-127	
MIDI CC 15	Undefined (MSB)	0-127	
MIDI CC 16 – 19	General Purpose (MSB)	0-127	
MIDI CC 20 – 31	Undefined (MSB)	0-127	
MIDI CC 32 – 63	LSB Controller for 0-31	0-127	
MIDI CC 64	Damper Pedal on/off	≤63 off, ≥64 on	On/off switch that controls sustain pedal. Nearly every synth will react to CC 64. (See also Sostenuto CC 66)
MIDI CC 65	Portamento on/off	≤63 off, ≥64 on	On/off switch
MIDI CC 66	Sostenuto Pedal on/off	≤63 off, ≥64 on	On/off switch – Like the Sustain controller (CC 64), However, it only holds notes that were "On" when the pedal was pressed. People use it to "hold" chords" and play melodies over the held chord.
MIDI CC 67	Soft Pedal on/off	≤63 off, ≥64 on	On/off switch – Lowers the volume of notes played.
MIDI CC 68	Legato FootSwitch	≤63 off, ≥64 on	On/off switch – Turns Legato effect between 2 subsequent notes on or off.
MIDI CC 69	Hold 2	≤63 off, ≥64 on	Another way to "hold notes" (see MIDI CC 64 and MIDI CC 66). However notes fade out according to their release parameter rather than when the pedal is released.
MIDI CC 70	Sound Controller 1	0-127	Usually controls the way a sound is produced. Default = Sound Variation.
MIDI CC 71	Sound Controller 2	0-127	Allows shaping the Voltage Controlled Filter (VCF). Default = Resonance also (Timbre or Harmonics)
MIDI CC 72	Sound Controller 3	0-127	Controls release time of the Voltage controlled Amplifier (VCA). Default = Release Time.
MIDI CC 73	Sound Controller 4	0-127	Controls the "Attack" of a sound. The attack is the amount of time it takes for the sound to reach maximum amplitude.
MIDI CC 74	Sound Controller 5	0-127	Controls VCFs cutoff frequency of the filter.
MIDI CC 75	Sound Controller 6	0-127	Generic – Some manufacturers may use to further shave their sounds.
MIDI CC 76	Sound Controller 7	0-127	Generic – Some manufacturers may use to further shave their sounds.
MIDI CC 77	Sound Controller 8	0-127	Generic – Some manufacturers may use to further shave their sounds.
MIDI CC 78	Sound Controller 9	0-127	Generic – Some manufacturers may use to further shave their sounds.
MIDI CC 79	Sound Controller 10	0-127	Generic – Some manufacturers may use to further shave their sounds.
MIDI CC 80	General Purpose MIDI CC Controller	0-127	Decay Generic or on/off switch ≤63 off, ≥64 on
MIDI CC 81	General Purpose MIDI CC Controller	0-127	Hi-Pass Filter Frequency or Generic on/off switch ≤63 off, ≥64 on
MIDI CC 82	General Purpose MIDI CC Controller	0-127	Generic on/off switch ≤63 off, ≥64 on
MIDI CC 83	General Purpose MIDI CC Controller	0-127	Generic on/off switch ≤63 off, ≥64 on
MIDI CC 84	Portamento CC Control	0-127	Controls the amount of Portamento.
MIDI CC 85 – 87	Undefined	—	
MIDI CC 88	High Resolution Velocity Prefix	0-127	Extends the range of possible velocity values
MIDI CC 89 & 90	Undefined	—	
MIDI CC 91	Effect 1 Depth	0-127	Usually controls reverb send amount
MIDI CC 92	Effect 2 Depth	0-127	Usually controls tremolo amount
MIDI CC 93	Effect 3 Depth	0-127	Usually controls chorus amount
MIDI CC 94	Effect 4 Depth	0-127	Usually controls detune amount
MIDI CC 95	Effect 5 Depth	0-127	Usually controls phaser amount
MIDI CC 96	(+1) Data Increment	N/A	Usually used to increment data for RPN and NRPN messages.
MIDI CC 97	(-1) Data Decrement	N/A	Usually used to decrement data for RPN and NRPN messages.
MIDI CC 98	Non-Registered Parameter Number LSB (NRPN)	0-127	For controllers 6, 38, 96, and 97, it selects the NRPN parameter.
MIDI CC 99	Non-Registered Parameter Number MSB (NRPN)	0-127	For controllers 6, 38, 96, and 97, it selects the NRPN parameter.
MIDI CC 100	Registered Parameter Number LSB (RPN)	0-127	For controllers 6, 38, 96, and 97, it selects the RPN parameter.
MIDI CC 101	Registered Parameter Number MSB (RPN)	0-127	For controllers 6, 38, 96, and 97, it selects the RPN parameter.
MIDI CC 102 – 119	Undefined	—	
MIDI CC 120 to 127 are "Channel Mode Messages."			
MIDI CC 120	All Sound Off	0	Mutes all sound. It does so regardless of release time or sustain. (See MIDI CC 123)
MIDI CC 121	Reset All Controllers	0	It will reset all controllers to their default.
MIDI CC 122	Local on/off Switch	0 off, 127 on	Turns internal connection of a MIDI keyboard or workstation, etc. on or off. If you use a computer, you will most likely want local control off to avoid notes being played twice. Once locally and twice when the note is sent back from the computer to your keyboard.
MIDI CC 123	All Notes Off	0	Mutes all sounding notes. Release time will still be maintained, and notes held by sustain will not turn off until sustain pedal is depressed.
MIDI CC 124	Omni Mode Off	0	Sets to "Omni Off" mode.
MIDI CC 125	Omni Mode On	0	Sets to "Omni On" mode.
MIDI CC 126	Mono Mode	→	Sets device mode to Monophonic. The value equals the number of channels, or 0 if the number of channels equals the number of voices in the receiver.
MIDI CC 127	Poly Mode	0	Sets device mode to Polyphonic.

MIDI NOTE NAMES vs NUMBERS

MIDI No°	Piano Key No°	Note Name (Roland)	Note Name (Yamaha)
108	88	C8	C7
107	87	B7	B6
106	86	A#7/Bb7	A#6/Bb6
105	85	A7	A6
104	84	G#7/Ab7	G#6/Ab6
103	83	G7	G6
102	82	F#7/Gb7	F#6/Gb6
101	81	F7	F6
100	80	E7	E6
99	79	D#7/Eb7	D#6/Eb6
98	78	D7	D6
97	77	C#7/Db7	C#6/Db6
96	76	C7	C6
95	75	B6	B5
94	74	A#6/Bb6	A#5/Bb5
93	73	A6	A5
92	72	G#6/Ab6	G#5/Ab5
91	71	G6	G5
90	70	F#6/Gb6	F#5/Gb5
89	69	F6	F5
88	68	E6	E5
87	67	D#6/Eb6	D#5/Eb5
86	66	D6	D5
85	65	C#6/Db6	C#5/Db5
84	64	C6	C5
83	63	B5	B4
82	62	A#5/Bb5	A#4/Bb4
81	61	A5	A4
80	60	G#5/Ab5	G#4/Ab4
79	59	G5	G4
78	58	F#5/Gb5	F#4/Gb4
77	57	F5	F4
76	56	E5	E4
75	55	D#5/Eb5	D#4/Eb4
74	54	D5	D4
73	53	C#5/Db5	C#4/Db4
72	52	C5	C4
71	51	B4	B3
70	50	A#4/Bb4	A#3/Bb3
69	49	A4 concert	A3
68	48	G#4/Ab4	G#3/Ab3
67	47	G4	G3
66	46	F#4/Gb4	F#3/Gb3
65	45	F4	F3
64	44	E4	E3
63	43	D#4/Eb4	D#3/Eb3
62	42	D4	D3
61	41	C#4/Db4	C#3/Db3
60	40	C4 (middle C)	C3
59	39	B3	B2
58	38	A#3/Bb3	A#2/Bb2
57	37	A3	A2
56	36	G#3/Ab3	G#2/Ab2
55	35	G3	G2
54	34	F#3/Gb3	F#2/Gb2
53	33	F3	F2
52	32	E3	E2
51	31	D#3/Eb3	D#2/Eb2
50	30	D3	D2
49	29	C#3/Db3	C#2/Db2
48	28	C3	C2
47	27	B2	B1
46	26	A#2/Bb2	A#1/Bb1
45	25	A2	A1
44	24	G#2/Ab2	G#1/Ab1
43	23	G2	G1
42	22	F#2/Gb2	F#1/Gb1
41	21	F2	F1
40	20	E2	E1
39	19	D#2/Eb2	D#1/Eb1
38	18	D2	D1
37	17	C#2/Db2	C#1/Db1
36	16	C2	C1
35	15	B1	B0
34	14	A#1/Bb1	A#0/Bb0
33	13	A1	A0
32	12	G#1/Ab1	G#0/Ab0
31	11	G1	G0
30	10	F#1/Gb1	F#0/Gb0
29	9	F1	F0
28	8	E1	E0
27	7	D#1/Eb1	D#0/Eb0
26	6	D1	D0
25	5	C#1/Db1	C#-1/Db0
24	4	C1	C-1
23	3	B0	B-1
22	2	A#0/Bb0	A#-1/Bb-1
21	1	A0	A-1

* Two Japanese giants of music technology get behind a new universal communication protocol for electronic musical instruments. Did it all go smoothly? Well yes, but with one major catch. Yamaha refused to adopt a lower first note (it's an octave beneath that of a standard piano) as it's keyboard didn't go down that far. So much confusion has arisen from these two protocols for Middle C. Roland's preferred C4 (more commonly adopted in scientific and music-theory study) and the all-new Yamaha tuning where the lowest note is A-1 and middle C is C3. Check your DAW's preferences to see which C they're pinning their flag to. For avoidance of doubt it's always MIDI Note No° 60.

www.thecrowhillcompany.com