

Creating MIDI Files with binasc

Craig Stuart Sapp

1 February 2018

Download binasc

(install Hombrew from <http://brew.sh> first)

In /Applications/Utilities/Terminal.app, type:

```
git clone http://github.com/craigsapp/binasc  
cd binasc  
make  
make install
```

Binasc means binary  ascii

A form of hex editor: https://en.wikipedia.org/wiki/Hex_editor

Hex byte codes to actual bytes

```
4d 54 68 64 00 00 00 06 00 00 00 01 00 80 4d 54 72 6b 00 00 00 8c 00 ff 58  
04 04 02 30 08 00 ff 59 02 00 00 00 90 3c 28 81 00 90 3c 00 00 90 3c 1e 81  
00 90 3c 00 00 90 43 2d 81 00 90 43 00 00 90 43 32 81 00 90 43 00 00 90 45  
2d 81 00 90 45 00 00 90 45 32 81 00 90 45 00 00 90 43 23 82 00 90 43 00 00  
90 41 32 81 00 90 41 00 00 90 41 2d 81 00 90 41 00 00 90 40 32 40 90 40 00  
40 90 40 28 40 90 40 00 40 90 3e 2d 40 90 3e 00 40 90 3e 32 40 90 3e 00 40  
90 3c 1e 82 00 90 3c 00 00 ff 2f 00
```



`binasc file.txt -c file.mid`



-C → compile the output into file.mid

Lowest level

Hex byte codes are the most direct representation of a MIDI file's contents:

```
4d 54 68 64 00 00 00 06 00 00 00 01 00 80 4d 54 72 6b 00 00 00 8c 00 ff 58
04 04 02 30 08 00 ff 59 02 00 00 00 90 3c 28 81 00 90 3c 00 00 90 3c 1e 81
00 90 3c 00 00 90 43 2d 81 00 90 43 00 00 90 43 32 81 00 90 43 00 00 90 45
2d 81 00 90 45 00 00 90 45 32 81 00 90 45 00 00 90 43 23 82 00 90 43 00 00
90 41 32 81 00 90 41 00 00 90 41 2d 81 00 90 41 00 00 90 40 32 40 90 40 00
40 90 40 28 40 90 40 00 40 90 3e 2d 40 90 3e 00 40 90 3e 32 40 90 3e 00 40
90 3c 1e 82 00 90 3c 00 00 ff 2f 00
```

But this is not very readable for humans...

Next level of abstraction

Using the *binasc* syntax for alternate representation of bytecodes

"MThd" → 4d 54 68 64

Use bytecode of ASCII printable character

4'6 → 00 00 00 06

Four-byte value for decimal number 6

2'1 → 00 01

Two-byte value for decimal number 1

'60 → 3c

One-byte value for decimal number 60

v128 → 81 00

v0 → 00

Variable-length-value for decimal number 128

Linebreaks are not important, just for readability.

```
"MThd"
4'6
2'0
2'1
2'128
"MTTrk"
4'140
v0      ff 58 v4 '4 '2 '48 '8
v0      ff 59 v2 '0 '0
v0      90 '60 '40
v128    90 '60 '0
v0      90 '60 '30
v128    90 '60 '0
v0      90 '67 '45
v128    90 '67 '0
v0      90 '67 '50
v128    90 '67 '0
v0      90 '69 '45
v128    90 '69 '0
v0      90 '69 '50
v128    90 '69 '0
v0      90 '67 '35
v256    90 '67 '0
v0      90 '65 '50
v128    90 '65 '0
v0      90 '65 '45
v128    90 '65 '0
v0      90 '64 '50
v64     90 '64 '0
v64     90 '64 '40
v64     90 '64 '0
v64     90 '62 '45
v64     90 '62 '0
v64     90 '62 '50
v64     90 '62 '0
v64     90 '60 '30
v256    90 '60 '0
v0      ff 2f v0
```

Patch (instrument) Change

<http://www.ccarh.org/courses/253/handout/gminstruments>

Patch change Message:

C_ #

“General MIDI” specifies 128 instruments:

Subtract 1

Piano Timbres:

- 1 Acoustic Grand Piano
- 2 Bright Acoustic Piano
- 3 Electric Grand Piano
- 4 Honky-tonk Piano
- 5 Rhodes Piano
- 6 Chorused Piano
- 7 Harpsichord
- 8 Clavinet

Brass Timbres:

- 57 Trumpet
- 58 Trombone
- 59 Tuba
- 60 Muted Trumpet
- 61 French Horn
- 62 Brass Section
- 63 Synth Brass 1
- 64 Synth Brass 2



```
"MThd"
4 '6
2 '0
2 '1
2 '128
"MTrk"
4 '143
v0      ff 58 v4 '4 '2 '48 '8
v0      ff 59 v2 '0 '0
v0      c0 60
v0      90 '60 '40
v128    90 '60 '0
v0      90 '60 '30
v128    90 '60 '0
v0      90 '67 '45
v128    90 '67 '0
v0      90 '67 '50
v128    90 '67 '0
v0      90 '69 '45
v128    90 '69 '0
v0      90 '69 '50
v128    90 '69 '0
v0      90 '67 '35
v256    90 '67 '0
v0      90 '65 '50
v128    90 '65 '0
v0      90 '65 '45
v128    90 '65 '0
v0      90 '64 '50
v64     90 '64 '0
v64     90 '64 '40
v64     90 '64 '0
v64     90 '62 '45
v64     90 '62 '0
v64     90 '62 '50
v64     90 '62 '0
v64     90 '60 '30
v256    90 '60 '0
v0      ff 2f v0
```

General MIDI drum sounds

- Channel 10 is reserved for percussion sounds in General MIDI
- “Channel 10” → 9 in MIDI command-byte channel nibble
- Note-on command byte: 0x99
- Each key on the percussion channel represents a different instrument:

35	Acoustic Bass Drum	51	Ride Cymbal 1	67	High Agogo
36	Bass Drum 1	52	Chinese Cymbal	68	Low Agogo
37	Side Stick	53	Ride Bell	69	Cabasa
38	Acoustic Snare	54	Tambourine	70	Maracas
39	Hand Clap	55	Splash Cymbal	71	Short Whistle
40	Electric Snare	56	Cowbell	72	Long Whistle
41	Low Floor Tom	57	Crash Cymbal 2	73	Short Guiro
42	Closed High Hat	58	Vibraslap	74	Long Guiro
43	High Floor Tom	59	Ride Cymbal 2	75	Claves
44	Pedal High Hat	60	High Bongo	76	High Wood Block
45	Low Tom	61	Low Bongo	77	Low Wood Block
46	Open High Hat	62	Mute High Conga	78	Mute Cuica
47	Low Mid Tom	63	Open High Conga	79	Open Cuica
48	High Mid Tom	64	Low Conga	80	Mute Triangle
49	Crash Cymbal 1	65	High Timbale	81	Open Triangle
50	High Tom	66	Low Timbale		

- Play the cowbell: 99 ‘56 ‘64
- Drum sounds do not require note-off message

Polyrhythm in Type-1 MIDI files

- Binasc operator **t**: MIDI tempo
- Convert from tempo into 3-byte microsecond value

ff 51 03 **t40** → ff 51 03 **16 e3 60**

0x16e360 = 1,500,000 microseconds
= 1.5 seconds per beat
60/1.5 = 40 beats/minute

- Ticks-per-quarter notes set to 6 (LCM of 2 and 3) **Track 1**



Expression
track

Track 1

Track 2

```
"MThd"  
4 ' 6  
2 ' 1  
2 ' 3  
2 ' 6
```

```
"MTrk"  
4 ' 11  
v0 ff 51 03 t40  
v0 ff 2f 00
```

```
"MTrk"  
4 ' 28  
v0 99 '76 '64  
v2 99 '76 '64  
v2 99 '76 '64  
v2 99 '76 '64  
v2 99 '76 '64  
v2 99 '76 '64  
v0 ff 2f 00
```

```
"MTrk"  
4 ' 24  
v0 99 '77 '64  
v3 99 '77 '64  
v3 99 '77 '64  
v3 99 '77 '64  
v3 99 '77 '64  
v0 ff 2f 00
```

Polyrhythm in Type-0 MIDI file

"MThd"

4'6

2'1

2'3

2'6

"MTrk"

4'11

v0 ff 51 03 t40

v0 ff 2f 00

"MTrk"

4'28

v0 99 '76 '64

v2 99 '76 '64

v2 99 '76 '64

v2 99 '76 '64

v2 99 '76 '64

v2 99 '76 '64

v0 ff 2f 00

"MTrk"

4'24

v0 99 '77 '64

v3 99 '77 '64

v3 99 '77 '64

v3 99 '77 '64

v3 99 '77 '64

v0 ff 2f 00



"MThd"

4'6

2'0

2'1

2'6

"MTrk"

4'59

v0 ff 51 03 t40

v0 ff 2f 00

v0 99 '76 '64

v0 99 '77 '64

v2 99 '76 '64

v1 99 '77 '64

v1 99 '76 '64

v2 99 '76 '64

v0 99 '77 '64

v2 99 '76 '64

v1 99 '77 '64

v1 99 '76 '64

v2 99 '77 '64

v0 ff 2f 00



Pitchbend

Binasc syntax: **p** followed by a floating-point number in the range from -1.0 to +1.0 is converted into the two data bytes for MIDI pitch bend messages.



MIDI Pitch Bend Message:

E_ lsb msb

E_ = hex byte code starting with nibble “E”.

lsb = least significant 7 bits of value

msb = most significant 7 bits of value

p-1	→	00 00	→	000,0000;000,0000	=	0
p-0.5	→	00 20	→	010,0000;000,0000	=	4096
p0	→	00 40	→	100,0000;000,0000	=	8192
p0.5	→	7f 5f	→	101,1111;111,1111	=	12287
p1	→	7f 7f	→	111,1111;111,1111	=	16383

Pitch bend range is programmable (depending on synth), but typical default range is +/- 200 cents (+/- 2 semitones).

```
"MThd"  
4'6  
2'0  
2'1  
2'100
```

```
"MTrk"  
4'72
```

```
v0 90 '60 '100  
v90 80 '60 '100
```

```
v10 e0 p0.5  
v0 90 '59 '100  
v90 80 '59 '100
```

```
v10 e0 p-0.5  
v0 90 '61 '100  
v90 80 '61 '100
```

```
v10 e0 p1  
v0 90 '58 '100  
v90 80 '58 '100
```

```
v10 e0 p-1  
v0 90 '62 '100  
v90 80 '62 '100
```

```
v10 e0 p0  
v0 90 '60 '100  
v90 80 '60 '100
```

```
v0 ff 2f 00
```

Glissandos/Vibrato



```
"MThd"
```

```
4'6
```

```
2'0
```

```
2'1
```

```
2'100
```

```
"MTrk"
```

```
4'66
```

```
v0    ff 51 03 t40
```

```
v0    e0 p0.0
```

```
v0    c0 '71          ; clarinet
```

```
v0    90 '60 64
```

```
v20   e0 p0.05
```

```
v10   e0 p0.10
```

```
v10   e0 p0.15
```

```
v10   e0 p0.20
```

```
v10   e0 p0.25
```

```
v10   e0 p0.30
```

```
v10   e0 p0.35
```

```
v10   e0 p0.40
```

```
v10   e0 p0.45
```

```
v10   e0 p0.50
```

```
v30   80 '60 00
```

```
v0    ff 2f 00
```

Temperament

Quarter-comma meantone

v0	e0	'0	'64
v0	e1	'41	'56
v0	e2	'105	'61
v0	e3	'38	'67
v0	e4	'79	'59
v0	e5	'11	'65
v0	e6	'56	'57
v0	e7	'117	'62
v0	e8	'30	'55
v0	ea	'90	'60
v0	eb	'23	'66
v0	ec	'68	'58

Pitch-class Cent deviation from ET

c	0
c#	-24.0
d	-6.8
e-	+10.3
e	-13.7
f	+3.4
f#	-20.5
g	-3.4
a-	-27.4
a	-10.3
b-	+6.8
b	-17.1



Temperament

Kirnberger III

v0	e0	'0	'64
v0	e1	'112	'60
v0	e2	'104	'61
v0	e3	'16	'62
v0	e4	'79	'59
v0	e5	'48	'63
v0	e6	'112	'60
v0	e7	'116	'62
v0	e8	'64	'61
v0	ea	'92	'60
v0	eb	'96	'62
v0	ec	'32	'60

Pitch-class Cent deviation from ET

c	0
c#	-9.775
d	-6.843
e-	-5.865
e	-13.686
f	-1.955
f#	-9.776
g	-3.422
a-	-7.82
a	-10.265
b-	-3.91
b	-11.73



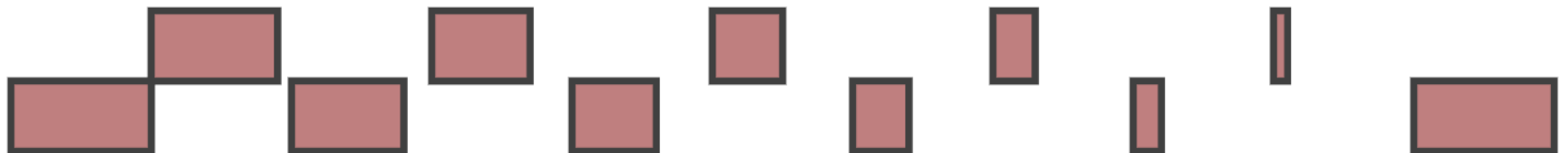
Staccato



```
"MThd"  
4'6  
2'0  
2'1  
2'100
```

```
"MTrk"  
4'95
```

```
v0    c0    '19  
v0     90    '60 64  
v100  80    '60 64  
v0     90    '61 64  
v90    80    '61 64  
v10    90    '60 64  
v80    80    '60 64  
v20    90    '61 64  
v70    80    '61 64  
v30    90    '60 64  
v60    80    '60 64  
v40    90    '61 64  
v50    80    '61 64  
v50    90    '60 64  
v40    80    '60 64  
v60    90    '61 64  
v30    80    '61 64  
v70    90    '60 64  
v20    80    '60 64  
v80    90    '61 64  
v10    80    '61 64  
v90    90    '60 64  
v100  80    '60 64  
v0     ff   2f 00
```



Ticks per Quarter & Round-off Error

"MThd"

4 '6

2 '0

2 '1

2 '128

"MTrk"

4 '76

v0 90 '60 64

v43 80 '60 64

v0 90 '62 64

v43 80 '62 64

v0 90 '64 64

v42 80 '64 64

v0 90 '60 64

v43 80 '60 64

v0 90 '62 64

v43 80 '62 64

v0 90 '64 64

v42 80 '64 64

v0 90 '60 64

v43 80 '60 64

v0 90 '62 64

v43 80 '62 64

v0 90 '64 64

v42 80 '64 64

v128 ff 2f 00



"MThd"

4 '6

2 '0

2 '1

2 '120

"MTrk"

4 '76

v0 90 '60 64

v40 80 '60 64

v0 90 '62 64

v40 80 '62 64

v0 90 '64 64

v40 80 '64 64

v0 90 '60 64

v40 80 '60 64

v0 90 '62 64

v40 80 '62 64

v0 90 '64 64

v40 80 '64 64

v0 90 '60 64

v40 80 '60 64

v0 90 '62 64

v40 80 '62 64

v0 90 '64 64

v40 80 '64 64

v120 ff 2f 00



Continuous controllers

General MIDI controller numbers:

<http://www.ccarh.org/courses/253/handout/controllers>

Controller 10 (zero offset) is stereo pan left/right

0
left



127
right



```
"MThd"  
4'6  
2'1  
2'3  
2'100
```

```
"MTrk"  
4'76
```

```
v0    b9 '10 '0  
v0    99 '60 '90  
v100 b9 '10 '16  
v0    99 '60 '90  
v100 b9 '10 '32  
v0    99 '60 '90  
v100 b9 '10 '48  
v0    99 '60 '90  
v100 b9 '10 '64  
v0    99 '60 '90  
v100 b9 '10 '80  
v0    99 '60 '90  
v100 b9 '10 '96  
v0    99 '60 '90  
v100 b9 '10 '112  
v0    99 '60 '90  
v100 b9 '10 '127  
v0    99 '60 '90
```

```
v0 ff 2f 00
```

Attack velocity versus Main volume

```
"MThd"  
4'6  
2'0  
2'1  
2'200  
"MTrk"  
4'80  
v0 99 '60 '10  
v100 99 '60 '20  
v100 99 '60 '30  
v100 99 '60 '40  
v100 99 '60 '50  
v100 99 '60 '60  
v100 99 '60 '70  
v100 99 '60 '80  
v100 99 '60 '90  
v100 99 '60 '100  
v100 99 '60 '110  
v100 99 '60 '120  
v100 99 '60 '127  
v100 99 '60 '110  
v100 99 '60 '90  
v100 99 '60 '70  
v100 99 '60 '50  
v100 99 '60 '30  
v100 99 '60 '10  
v0 ff 2f 00
```

```
"MThd"  
4'6  
2'0  
2'1  
2'200  
"MTrk"  
4'156  
v0 b9 07 '10 v0 99 '60 '127  
v100 b9 07 '20 v0 99 '60 '127  
v100 b9 07 '30 v0 99 '60 '127  
v100 b9 07 '40 v0 99 '60 '127  
v100 b9 07 '50 v0 99 '60 '127  
v100 b9 07 '60 v0 99 '60 '127  
v100 b9 07 '70 v0 99 '60 '127  
v100 b9 07 '80 v0 99 '60 '127  
v100 b9 07 '90 v0 99 '60 '127  
v100 b9 07 '100 v0 99 '60 '127  
v100 b9 07 '110 v0 99 '60 '127  
v100 b9 07 '120 v0 99 '60 '127  
v100 b9 07 '127 v0 99 '60 '127  
v100 b9 07 '110 v0 99 '60 '127  
v100 b9 07 '90 v0 99 '60 '127  
v100 b9 07 '70 v0 99 '60 '127  
v100 b9 07 '50 v0 99 '60 '127  
v100 b9 07 '30 v0 99 '60 '127  
v100 b9 07 '10 v0 99 '60 '127  
v0 ff 2f 00
```



How to cheat at counting bytes

In Terminal.app, use the “wc” command to count the number of bytes within a track.

Step 1: compile your MIDI file with binasc

```
binasc myfile.txt -c myfile.mid
```

Step 2: decompile your MIDI file into hex bytes

```
binasc myfile.mid > bytecodes.txt
```

Step 3: pull out the bytes of a track (easy if a type-0 MIDI file, otherwise search for next “MTrk” tag and/or ff 2f 00 end-of-track marker) and save to separate text file. Then use wc to count the “words” in the file:

```
wc -w trackbytes.txt
```

Step 4: take that number and put into track size field of original file, `myfile.txt`

Homework 5b

- Create a MIDI file with at least 25 MIDI messages
- Incorporate at least 4 of the following features into the MIDI file:
 - Type-1 MIDI file (multi-track)
 - Patch change (timbre/instrument change)
 - Pitch bend (glissandos, temperament)
 - Articulations (accents, staccato/legato)
 - Continuous controller (master volume, pan)
 - Meta message (tempo, key signature, time signature, track name)
 - Percussion sounds
 - Chords
- Send me a copy of the MIDI file by next Thursday's class so we can play them.