FILE2MIDI & MIDI2FILE

On the Internet, I found the following two programs under "Freeware" at this URL: http://gnmidi.com/

MIDI2TXT.EXE and TXT2MIDI.EXE

If you read the included Text files for each of these two programs, you will see that they were designed for people who create MIDI music and then want to edit notes and other parameters. I decided to use these programs for my purposes only; which was to have the ability to encode files into MIDI music files and then to decode these MIDI music files back into the original files. Please read this document completely before attempting to use these programs.

Originally, I accomplished this with two programs that I wrote: ENCODE.EXE AND DECODE.EXE, which were written to ONLY encode and decode TEXT files, NOT binary files. Later, I figured out a way to encode/decode ANY file, TEXT and/or BINARY files. The results became ENCODE2.EXE & DECODE2.EXE and ENCODE3.EXE & DECODE3.EXE. Each PAIR of programs are designed to work together and each PAIR of programs will be explained later in this document. First, it is necessary to explain MIDI2TXT.EXE and TXT2MIDI.EXE as it was used for my project.

MIDI2TXT.EXE converts an existing MIDI music file into an editable *script text file* It only has a command line interface and MUST use the following parameters:

MIDI2TXT -dec [path]Myname.mid [path]Script.txt
(Note:-dec is case sensitive AND is necessary)

Below is the fixed format of the Script.txt file. First, there is a *header*:

```
    // MidiFile.mid
mthd
 version 0 // single multichanneltrack
 // 1 track
 unit 120 // is 1/4
end mthd

mtrk(1)  // track 1
```

Below this header is the MIDI information as Notes ON and Notes OFF such as:

```
+22 90;      (Explanation: Note 22 ON, Velocity 90)
30;-22 64;   (Explanation: Note 22 OFF, Velocity 64)
+90 2;       (Explanation: Note 90 ON, Velocity 2)
30;-90 64;   (Explanation: Note 90 OFF, Velocity 63)
etc...
```

TXT2MIDI.EXE converts a script text file (as shown above) into a MIDI music file. TXT2MIDI.EXE only has a command line interface which has the following parameters:

TXT2MIDI [path]Myscript.txt [path]Myname.mid

ENCODE.EXE (my program) reads a *TEXT* file (the one to be converted into a MIDI music file) and converts it into the necessary *script text file* that can be read by TXT2MIDI.EXE.

DECODE.EXE (my program) reads the script text file (created by MIDI2TXT.EXE) and converts it back into an exact copy of the original *TEXT* file.

The formulas I used to write the script text file using ENCODE.EXE (my program) are as follows:

ASCII-09 (Tab) = Note 1
ASCII-10 (Line Feed) = Note 2
ASCII-13 (Carriage Return) = Note 3
(ASCII Value - 28) = All other Notes *(Note: Only within the range of ASCII-32 through ASCII-127)*

When using ENCODE.EXE, two parameters can be set by the userto write into the script text file:

TEMPO = Any value from 120 through 240 (Offset 0Dh in MIDI File)
INSTRUMENT = Any instrument from 1 through 128 (Offset 18h in MIDI File)

ENCODE.EXE uses Velocity Ranges assigned randomly from 32 to 96

The formulas I used to read the script text file (created by MIDI2TXT.EXE) using DECODE.EXE (my program) are the exact reverse of those used in ENCODE.EXE:

Note 1 = ASCII-09 (Tab)
Note 2 = ASCII-10 (Line Feed)
Note 3 = ASCII-13 (Carriage Return)
All other Notes = ASCII (Note Value + 28)

The complete list of instruments are as follows: (*Note: Encode.exe subtracts one from the following numbers)*

MIDI Instrument List

Piano:	***Chromatic Percussion:***	***Organ:***
1 Acoustic Grand Piano	9 Celesta	17 Drawbar Organ
2 Bright Acoustic Piano	10 Glockenspiel	18 Percussive Organ
3 Electric Grand Piano	11 Music Box	19 Rock Organ
4 Honky-tonk Piano	12 Vibraphone	20 Church Organ
5 Electric Piano 1	13 Marimba	21 Reed Organ
6 Electric Piano 2	14 Xylophone	22 Accordion
7 Harpsichord	15 Tubular Bells	23 Harmonica
8 Clavinet	16 Dulcimer	24 Tango Accordion

Guitar:	***Bass:***	***Strings:***
25 Acoustic Guitar (nylon)	33 Acoustic Bass	41 Violin
26 Acoustic Guitar (steel)	34 Electric Bass (finger)	42 Viola
27 Electric Guitar (jazz)	35 Electric Bass (pick)	43 Cello
28 Electric Guitar (clean)	36 Fretless Bass	44 Contrabass
29 Electric Guitar (muted)	37 Slap Bass 1	45 Tremolo Strings
30 Overdriven Guitar	38 Slap Bass 2	46 Pizzicato Strings
31 Distortion Guitar	39 Synth Bass 1	47 Orchestral Harp
32 Guitar harmonics	40 Synth Bass 2	48 Timpani

Ensemble:
49 String Ensemble 1
50 String Ensemble 2
51 Synth Strings 1
52 Synth Strings 2
53 Choir Aahs
54 Voice Oohs
55 Synth Voice
56 Orchestra Hit

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

Reed:
65 Soprano Sax
66 Alto Sax
67 Tenor Sax
68 Baritone Sax
69 Oboe
70 English Horn
71 Bassoon
72 Clarinet

Pipe:
73 Piccolo
74 Flute
75 Recorder
76 Pan Flute
77 Blown Bottle
78 Shakuhachi
79 Whistle
80 Ocarina

Synth Lead:
81 Lead 1 (square)
82 Lead 2 (sawtooth)
83 Lead 3 (calliope)
84 Lead 4 (chiff)
85 Lead 5 (charang)
86 Lead 6 (voice)
87 Lead 7 (fifths)
88 Lead 8 (bass + lead)

Synth Pad:
89 Pad 1 (new age)
90 Pad 2 (warm)
91 Pad 3 (polysynth)
92 Pad 4 (choir)
93 Pad 5 (bowed)
94 Pad 6 (metallic)
95 Pad 7 (halo)
96 Pad 8 (sweep)

Synth Effects:
97 FX 1 (rain)
98 FX 2 (soundtrack)
99 FX 3 (crystal)
100 FX 4 (atmosphere)
101 FX 5 (brightness)
102 FX 6 (goblins)
103 FX 7 (echoes)
104 FX 8 (sci-fi)

Ethnic:
105 Sitar
106 Banjo
107 Shamisen
108 Koto
109 Kalimba
110 Bag pipe
111 Fiddle
112 Shanai

Percussive:
113 Tinkle Bell
114 Agogo
115 Steel Drums
116 Woodblock
117 Taiko Drum
118 Melodic Tom
119 Synth Drum
120 Reverse Cymbal

Sound Effects:
121 Guitar Fret Noise
122 Breath Noise
123 Seashore
124 Bird Tweet
125 Telephone Ring
126 Helicopter
127 Applause
128 Gunshot

ENCODE2.EXE AND DECODE2.EXE will work on ANY file, text or binary. This is done by also using the MIDI Velocity: If the Note has a Velocity from 30 through 64, the ASCII Character is the value of the Note. If the Note has a Velocity from 65 through 99, the ASCII Character is the value of the Note + 128.

ENCODE3.EXE AND DECODE3.EXE will also work on ANY file, text or binary. This is done by also using the MIDI Velocity OPPOSITE the way it is used in ENCODE2.EXE and DECODE2.EXE: If the Note has a Velocity from 65 through 99, the value of the Note is the ASCII Character. If the Note has a Velocity from 30 through 64, the ASCII Character is the value of the Note + 128.

To use all of these programs, just make a new folder, such as "MIDI2FILE", and put all the programs within it. Then, include this folder in your PATH settings (usually found in the AUTOEXEC.BAT file). I recommend making a sub-folder below this folder, such as "WORK", for the creation of script text files & MIDI files. Though my programs, ENCODE.EXE and DECODE.EXE will ***not*** overwrite an existing file, MIDI2TXT.EXE and TXT2MIDI.EXE ***will*** overwrite an existing file. So by having a separate folder structure, such as "MIDI2TXT\WORK" to log into, you will isolate your permanent programs from accidentally becoming overwritten. There are no parameters for ENCODE.EXE and DECODE.EXE. Within these program screens there will be data prompts that are self explanatory. If you wish to go-back to a previous prompt to make a correction, just hit <Enter> only at the prompt. If you hit <Enter> only at the first prompt, you will exit back to the DOS prompt.

By including the program folder MIDI2FILE in your PATH settings; and by logging into the sub-folder WORK, you will be able to modify the command line parameters of TXT2MIDI.EXE and MIDI2TXT.EXE as follows:

TXT2MIDI Myscript.txt Myname.mid
and
MIDI2TXT -dec Myname.mid Script.txt

(Important Note: The reason that the option -dec MUST BE USED is because if it is NOT used, the Script.txt created by MIDI2TXT.EXE shows the notes in musical notation and my program DECODE.EXE needs the notes in decimal notation in order to convert the decimal notes into the ASCII characters of the original file.

To see the notes in a sheet-music notation format while playing, there is a cool Windows freeware program, MIDI Notation Player, available here:

http://notation.com/DownloadNotationPlayer.htm

Needless to say, the “music” that you create will have some interesting sounds. Have fun experimenting!

Chuck Mire
artmaker43@gmail.com