Implementation of Microcontroller Based MIDI Message Output for MIDI Drum

Tin Sang Hnin, Khaing Khaing Wai
Computer University, Maubin
joseph1.tete@gmail.com, khaingkhaing.73@gmail.com

Abstract

Musical Instrument Digital Interface (MIDI) message can command synthesizers to produce sound. This paper presents the method to generate MIDI message from microprocessor. This system is based on MIDI protocol and wavetable synthesis technique of sound synthesis theory. The main goal is to create a low cost drum sequencer. The application outlined in this work is the MIDI drum sequencer that command synthesizer to generate drum sounds .These sounds can be recorded so that they can be played back using a synthesizer or PC. No microphone is required for recording MIDI drum. The technique for recording is called sampling.

Keywords: MIDI message, MIDI protocol, wavetable synthesis technique, sound synthesis theory, sampling

1. Introduction

This paper presents a device built to generate sound when connects to synthesizer or PC as a synthesizer source. The functions of MIDI protocol and wavetable synthesis technique are constructed in microcontroller to generate MIDI message. Assembly language is used to implement the system. Piezo electric transducers, convert vibrations to electric signal, are used as drum pads to sense the force apply them when striking the piezo pads [6].

Microprocessor generates MIDI notes that command synthesizer to produce sound. PC or synthesizer is used to generate sounds. Synthesizer software is needed in PC to generate sound. When connecting to a computer, there are three possible options:

a) Use the games port, which connects directly the computer's sound card; or

b) Use the serial outlet and connect this directly to a serial port on the computer; or

c) Use the computer's USB port.

Generated electric signals are send to microprocessor. The microprocessor then converts the analogue signals to digital signals by using ADC. Sampling, quantization and encoding processes are performed in microprocessor [12]. Output from ADC is taken as velocity value as part of MIDI massage.

2. System block

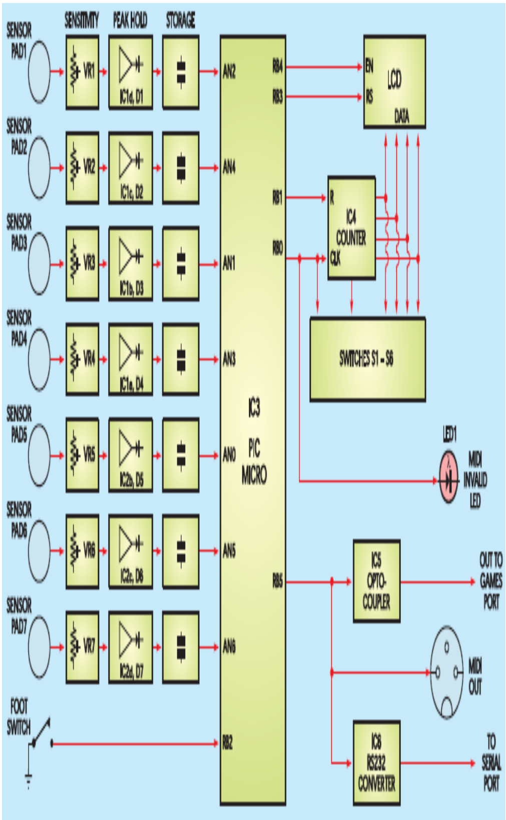

Figure 1. Block diagram of MIDI drum

Figure.1 shows the block diagram for the MIDI drum kit. Sensor pads from PAD 1 to 7 are connected to identical circuitry, including a sensitivity trimpot, peak hold buffer storage. The sensitivity adjustment allows any sensors to be adjusted to match the sensitivity of other sensor pads. Following the sensitivity sensor adjustment, the signal form each sensor is rectified and the peak value from the sensor is stored.

A microcontroller is used to monitor the stored signals from the sensors at the analogue input AN0 to AN6 and the footswitch at RB2. If a signal at any of the PAD input reaches the predetermined value, then the microcontroller decides that the sensor pad has been struck and a MIDI signal is produced at output RB5. Then this is applied to an optocoupler, the MIDI output socket and the RS-232 converter [11]. The microcontroller also drives the LCD and monitors switches S1 to S6.

3. Theory background

In this section, the theories that are used to generate MIDI message for MIDI drum are explained.

3.1. MIDI (Musical Instrument Digital Interface)

MIDI is an 8 bit language. It is a computer language. Like all computer languages MIDI is constructed from 1s and 0s (so called bits).

MIDI is an 8 bit language. It is a computer language. Like all computer languages MIDI is constructed from 1s and 0s (so called bits).

In the MIDI language, bits are grouped into units called bytes. The bytes in MIDI are made from 8 bits. Here is an example MIDI byte **10010011.**

Because the bytes in MIDI are 8 bits long it is said to be an 8-bit language. Other computer languages are 16-bit (16 bits in a byte) or even 24-bit, 32-bit, 64-bit, or 128-bit. MIDI is 8-bit because it is transmitted very slowly (for cost reasons) compared to many languages and therefore needs to be as small as possible to ensure that timing errors do not become noticeable.

In the MIDI language there are 256 different bytes. Each byte has a specific purpose. This example MIDI byte "says" Note On / Channel 2 **10010011**

This example MIDI byte "says"127 **01111111**

MIDI bytes fall into one of 2 categories

(a.) **Status bytes** always start with a 1 and define the type of message being sent. This is an example status byte **10010011** which means Note On / Channel 2

(b.) **Data bytes** start with a 0 and simply give a value between 0 and 127[5].

3.2. MIDI message

By combining bytes together, MIDI messages are constructed. There are 2 categories of MIDI message.

(a.) **Channel messages** are mostly to do with performance information sent on MIDI channels.

(b.) **System messages** handle system wide jobs like MIDI Time code and are not sent on channels.

All MIDI messages are constructed from at least 1 status byte and 1 data byte, often many many more.

Here is an example of a simple 3 byte MIDI message comprising a Status byte and 2 Data bytes. This message is telling a sound module set to respond on MIDI channel 1 to start playing a note (C3) at a velocity of 101 [5].

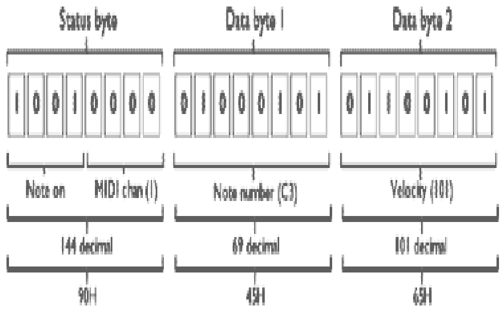

Figure2. MIDI note for C3

3.3. Wavetable Synthesis Technique

The audio samples or waveforms are stored in an array is known as a wavetable. To save memory, most wavetables do not store a recording for every note of every instrument.
Instead of only a handful of notes are stored for each instrument, and in some cases sample data may be shared across multiple instruments [10].

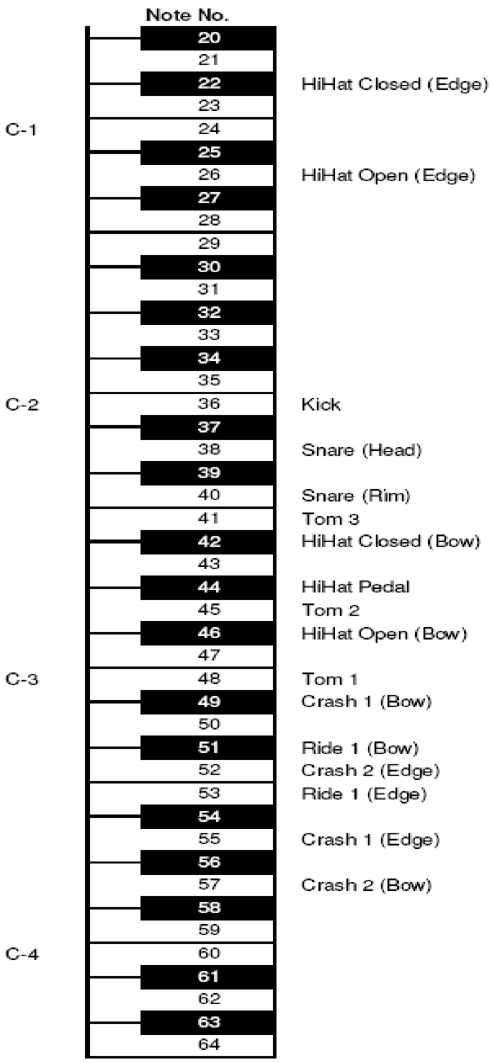

Figure 3. MIDI note number for drum pads

3.4. MIDI note number to frequency conversion

The standard tuning pitch is A4 (An above middle C) 440 Hertz (Hz). When the octave is divided into 12 equally spaced pitches, then each interval between these pitches when they are arranged within the same octave in an ascending sequence is the equal-tempered semitone:

21/12 = 1.059463094 . . .

The frequencies of equal-tempered pitches are easily calculated from any reference frequency according to:

$f = 2\ n/12\ f_{ref}$

where n is the number of semitones between the reference frequency (f_{ref}), and the desired frequency, f [8].

For example, the frequency of the equal-tempered pitch located a major third up (+4 semitones) from A440 is 440(24/12) = 554.365... Hz, while middle C, a major sixth down (-9 semitones), has a frequency of 440(2-9/12) = 261.6255...Hz.

Table 1. MIDI note number and frequency

51	B4	h'	493.883
50	A♯4/B♭4	ais'/b'	466.164
49	A4 concert pitch	a' Kammerton	440.000
48	G♯4/A♭4	gis'/as'	415.305
47	G4	g'	391.995
46	F♯4/G♭4	fis'/ges'	369.994
45	F4	f'	349.228
44	E4	e'	329.628
43	D♯4/E♭4	dis'/es'	311.127
42	D4	d'	293.665
41	C♯4/D♭4	cis'/des'	277.183
40	C4 (middle C)	c' (Schloss-C)	261.626
39	B3	h	246.942
38	A♯3/B♭3	ais/b	233.082
37	A3	a	220.000
36	G♯3/A♭3	gis/as	207.652
35	G3	g	195.998
34	F♯3/G♭3	fis/ges	184.997
33	F3	f	174.614
32	E3	e	164.814
31	D♯3/E♭3	dis/es	155.563
30	D3	d	146.832
29	C♯3/D♭3	cis/des	138.591
28	C3	c	130.813
27	B2	H	123.471
26	A♯2/B♭2	Ais/B	116.541
25	A2	A	110.000
24	G♯2/A♭2	Gis/As	103.826
23	G2	G	97.9989
22	F♯2/G♭2	Fis/Ges	92.4986
21	F2	F	87.3071
20	E2	E	82.4069
19	D♯2/E♭2	Dis/Es	77.7817
18	D2	D	73.4162
17	C♯2/D♭2	Cis/Des	69.2957
16	C2	C	65.4064
15	B1	H1	61.7354
14	A♯1/B♭1	Ais1/b1	58.2705
13	A1	A1	55.0000
12	G♯1/A♭1	Gis1/As1	51.9130
11	G1	G1	48.9995
10	F♯1/G♭1	Fis1/Ges1	46.2493
9	F1	F1	43.6536
8	E1	E1	41.2035
7	D♯1/E♭1	Dis1/Es1	38.8909

The calculation results of MIDI note number 7 to MIDI note number 51 are listed in Table 1.The frequencies for drum pads can also be calculated in the same way.

4. Proposed system design

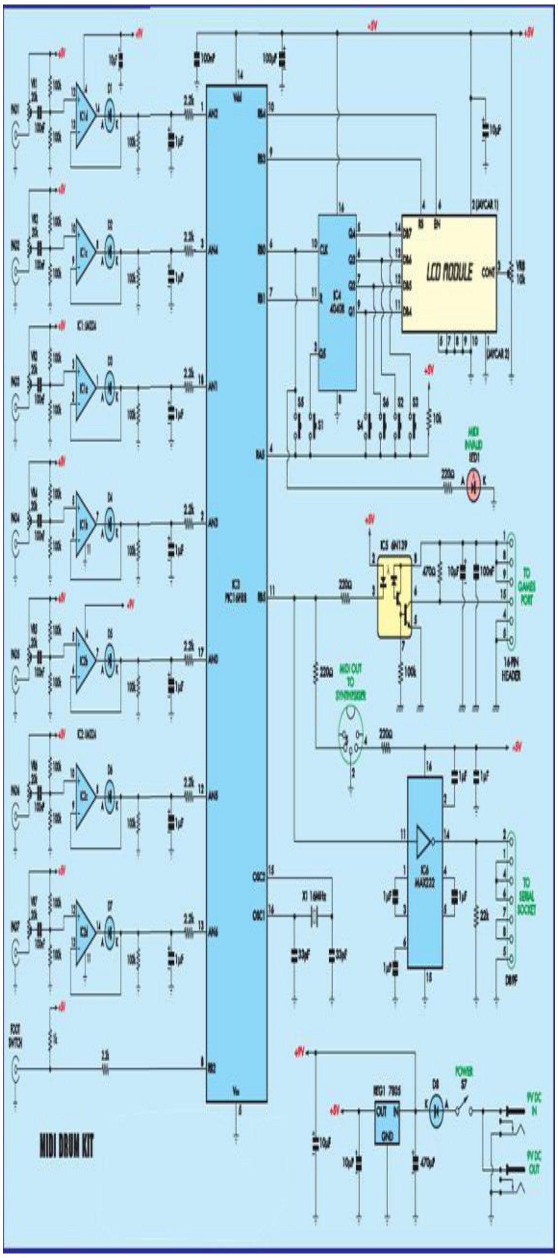

Figure 4. Complete circuit diagram

7 piezoelectric transducers and foot switch are used as drum pads that generate voltage directly proportional to the force apply them. A piezoelectric sensor is a device that uses the piezoelectric effect to measure pressure, acceleration, strain or force by converting them to an electrical signal. Piezoelectric elements generate a voltage proportional to the force applied to them [3, 6].

Signals from drum pads are sent as a series of codes that command the synthesizer to produce sounds. The codes are sent in MIDI format, standard signal used by the music industry to control synthesizers. MIDI codes include information such as the required information instrument to be played, it's position in the left and right sound stage. Sampling, quantization and encoding processes are performed in PIC as ADC (analogue-to-digital converter). Output from ADC is taken as velocity value as part of MIDI code.

There are four output sessions:

1. LCD
2. Game port
3. MIDI out
4. Serial port.

LCD displays the selected drum for each sensor pad input and the various settings. When connecting to a computer, there are three possible options:

(a.) Use the games port, which connects directly the computer's sound card; or

(b.) Use the serial outlet and connect this directly to a serial port on the computer; or

(c.) Use the computer's USB port.

MIDI signal provided on the MIDI output can't be use to drive a MIDI instrument when the unit is configured for serial output. Use optocoupler and the RS-232 converter to solve this problem. The optocoupler provides isolation between the MIDI drum and the computer connection via the games port. The RS-232 converter converts the 0-5V signal from PIC to a normal +_10V signal for the serial port. Software is required when using the computer as the synthesizer source.

Figure5 shows MIDI message for bass drum is playing in PC.

Figure 5. Photo of playing MIDI message for MIDI drum in PC

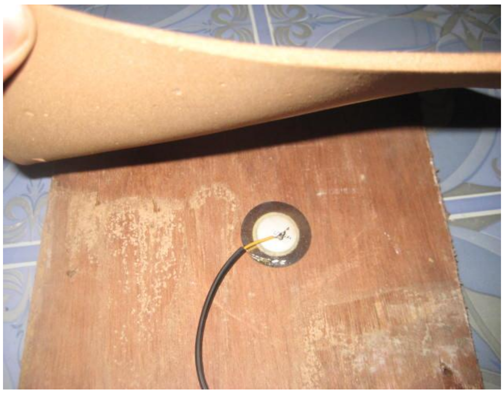

Figure 6. Photo of piezo sensor in drum pad

Figure 7. Photo of MIDI drum pads

Figure 8. Photo of complete circuit

5. Conclusion

The main advantage of ADC in this system is to produce variable velocity value for MIDI message. So the output of MIDI message can command the synthesizer to produce sound more likely to the real drum sound. For recording studios, at least one set of branded drum kit and several number of microphones are required to capture the drum sound. It is very high cost to buy one set of drum kit and microphones that need to capture each pad of drum kit. By this paper, very low cost drum machine can be created.

References

[1] http://en.wikipedia.org/wiki/MIDI

[2] http://en.wikipedia.org/wiki/Music_sequencer

[3] http://en.wikipedia.org/wiki/Piezoelectric_sensor

[4]http://en.wikipedia.org/wiki/Synthesizer

[5]http://tomscarff.110mb.com/midi_analyser/midi_note_frequency.html

[6]http://www.americanpiezo.com/piezo_theory/index.html

[7]http://www.edrum.info/

[8]http://www.planetoftunes.com/sequence/messages.html

[9]http://www.sengpielaudio.com/calculator-notenames.html

[10] Jim Heckroth. "Tutorial on MIDI and Music Synthesis", The MIDI Manufacturers Association

[11] Meg Walraed-Sullivan. "SHEET MUSIC GENERATOR: MIDI-PC INTERFACE", Cornell University

[12] Monson H. Hayes. "Schaum's Outline of Theory and Problems of Digital Signal Processing", SCHAUM'S OUTLINE SERIES