science, engineering and health studies

Research Article

Science, Engineering and Health Studies
https://li01.tci-thaijo.org/index.php/sehs
ISSN (Online): 2630-0087

Nanosecond latency drum kit

*Kundan Manohar Jadhav and Abhishek Kumar**

School of Electronics and Electrical Engineering, Lovely Professional University, Punjab, India

***Corresponding author**:
Abhishek Kumar
abkvjti@gmail.com

Received: 9 July 2021
Revised: 11 September 2021
Accepted: 24 November 2021
Published: 18 November 2022

Citation:
Jadhav, K. M., and Kumar, A. (2022). Nanosecond latency drum kit. Science, Engineering and Health Studies, 16, 22040005.

ABSTRACT

Music is incomplete without the beats of the drum. With the advancement in time, music is getting more electronic than acoustic because of its portable feature. Electronics drums are popular musical instruments for producing music and beats with ease. The electronics drum kit is very easy to set up and tune but, and it does not produce heavy noise like acoustic drum kits. The modern music instruments are based on the musical instrument digital interface (MIDI) protocol easier to connect for producing music with another electronic instrument. A musical instrument compatible with MIDI can be connected to the computer system and can produce high-quality sound. In this work, a prototype of a digital drum kit was developed. A piezo sensor at input sensed the pressure, followed by an analog to digital converter generate binary value for the processing based on MIDI protocol. The combination of digital to analog converter and I^2S module created corresponding sound. The static delay due to the component experience latency in milliseconds between action and sound. Action and sound get a millisecond-long static lag due to component latency. In this work we tried to minimize latency to nanosecond.

Keywords: drum; electric drum; latency; piezo sensor; MIDI

1. INTRODUCTION

Music is part of the human being from begin of mankind. Drum is very popular instrument because of its simplicity in playing and learning. Different types of drums have been developed, and the traditional drum was made from a wooden cylinder and animal skin was used as a membrane over it. The musician strikes a stick or hand over it to get sound. Different size of cylinders requires to produce different sound frequencies. Combinations of such different sizes of drums are used while producing music for having different types of sound effects. Two necessary components for producing music via membrane are drum creating low-frequency impulse sound and drum creating high-frequency sound. Even modern music production uses; bass drum with a huge radius cylinder to produce low-frequency sound and a snare drum to low-frequency sound with a smaller radius cylinder. Similar configuration available for Tabla set; bigger drum produces bass sound and smaller one produces flat or bell-like sound. Traditional drums are replaced by electric drums where the sound generates from prerecorded tracks. Digital, musical instruments look to acoustic instruments that foster relationships between gesture and sound. An acoustic instument produces sound in the reaction to action instantly, but some latency is built in the sound generation mechanism of instruments, for example in piano delay for key reaches the bottom and strikes bottom string 35 ms for pp-notes and 5 ms for ff-notes Latency between sound and action must synchronize in a musical context. Generally, 10-40 ms are often present due to the distance between sound and performer (Askenfelt and Jansson, 1987). The musical instrument set its way to control the latency strategies. The impact of latency on the performer depends on the quality of musical instruments. Electric drums are easy to set up and tune and they do not create noise. Digital drum kits or electric drums are readily available in the market at a very high price, encompasses microprocessors or microcontroller. The professional musician avoids the usage of electronic drums since their sound does not match the quality of the acoustic drum.

The threshold value of the latency for musical instruments is often set to <10 ms. It is very difficult for a fast musician to play electronic drums because of the pre-calculated latency of the microcontroller. A sensitive instrument shows a variation of 4 ms. Latency associated with acoustic kick drum 3 ms and snare drum 2 ms (for a drummer) whereas for the person standing away 6 ft experience at least 6 ms delay. Musical instruments with low latency and consistent action to sound delay are acceptable (Jack et al., 2016). Delay between key reaching to bottom and striking about 35 ms and 5 ms for ff notes (Askenfelt and Jansson, 1987). Barriers to electronics drums are latency and jitters with the instrument (Magnusson and Mendieta, 2007). Music instruments have a set of control gestures that performers use strategically to match with latency (Dinulică et al., 2021). Variable latency concept was introduced in Friberg and Sundberg, (1995) and listener were able to detect timing variation of 6 ms. Even highly trained musicians are sensitive to timing variation. They can achieve an error of 2 ms for a metronome and 1000 ms with a standard deviation of 10-16 ms (Fujii et al., 2011). The source of latency and their accumulative delay are shown in Table1.

Table 1. Latency and their sources in musical instrument

Latency reason	Latency (second)
Trigger setting	2-3
MIDI transfer	1
Buffer Size	1.33 at 48 kHz 64 frame
A2D-D2A conversion	1

Alternative solutions are field-programmable gate array (FPGA) based drum machines where latency can minimize with a signal processing algorithm so that fast drummers and musicians can play without lagging. Since selective numbers of lookup table in FPGA needed to enable Lu's method of implementation in Bianch et al. (2020) which saying circuit with lesser area was better than a microcontroller. Low and compatible action to sound latency was highly desirable in digital music instruments. Jack et al. (2016) explored the impact of latency and jitter on instrumental delay and performer action. The presence of jitter significantly degrades the timing synchronization. The present work focuses on developing a prototype of a music instrument digital interface (MIDI) - based drum kit with Spartan 6 FPGA, and resolves the issue of latency, resulting in the sound quality of the electronics drum matching the acoustic drums.

2. MATERIAL AND METHODS

An FPGA-based musical instrument comes up with a very short latency that is undetectable to the humane ear. FPGA based instrument has features low latency, reconfigurability, and portability which make them suitable for professional uses (Bianchi et al., 2020).

2.1 Proposed drum kit

Figure 1 presents the interfaced diagram of a proposed digital drum kit. The input section is composed of a piezo sensor followed by amplifier and analog to digital converter (ADC), the processing unit is Xilinx Spartan 6 FPGA and the output section comprises an intergrated inter-IC sound (I^2S) breakout board and speaker.

The input section requires a sensor followed by ADC for the drum pad. In this work, a group of piezo sensors is used for the drum pad, generating an electric signal whenever pressure is applied over it. The sensor is placed under rubber pads and whenever the pad is hit creates an acoustic property change and this triggers the piezo to generate an electric signal. When the sensor is struck, produces analog signal converts to digital, and is transmitted to FPGA serially. After the drum trigger information is obtained in the FPGA, the correct sound sample should be played with the right sampling frequency. Sometimes multiple drum pads are struck at once so all the sound samples should be first added and played together. The given Figure 2 is the signal processing algorithm for the processing of drum audio samples. Velocity weight is multiplied with each sound sample then resultant samples are again added to provide output. The algorithm gives an 8-bit sample value at each clock. FPGA board process the signal and gives the output to I^2S in pulse code modulation (PCM) format by following some algorithm presented in Figure 2, which outputs an analog signal through the speaker.

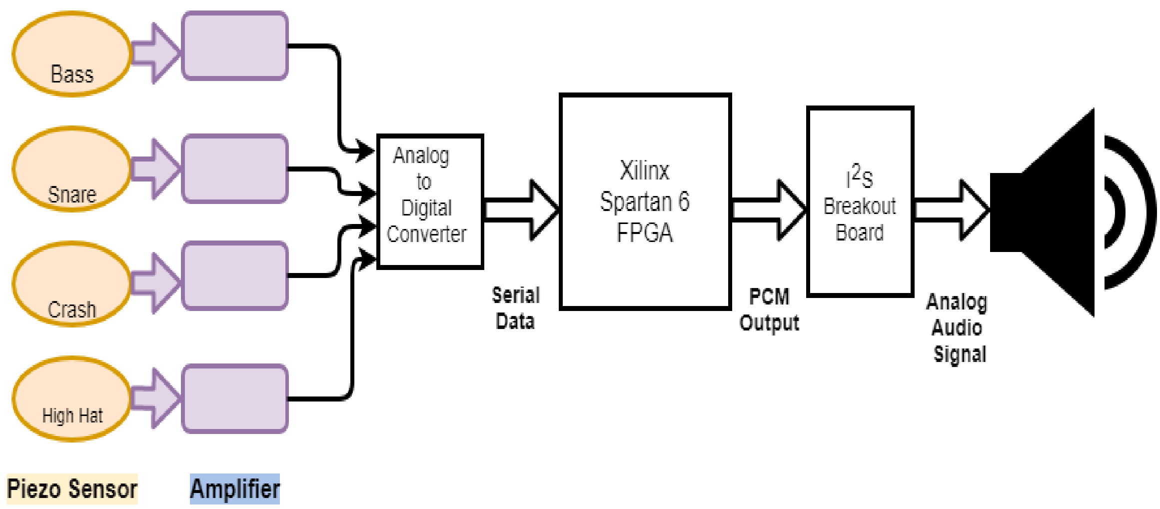

Figure 1. Interfacing diagram of a drum kit

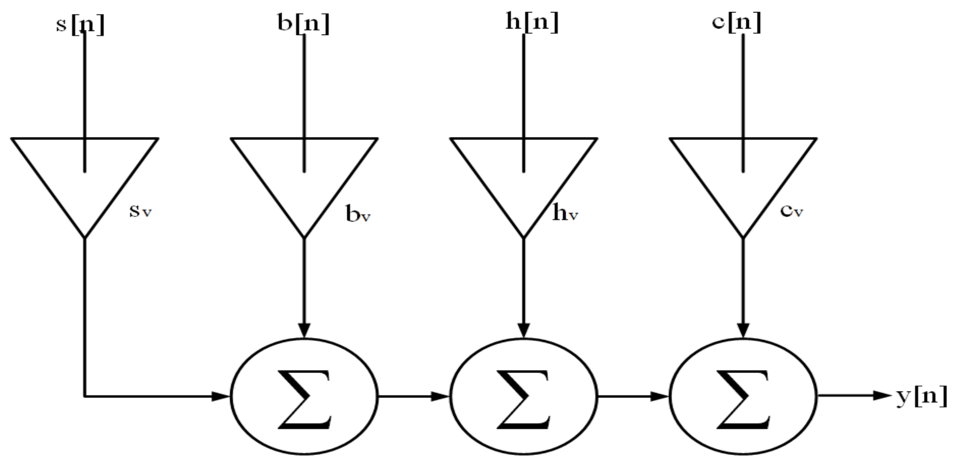

Figure 2. Signal processing algorithm

2.2 Input Section

The sensor is the device that converts one form of energy to electrical energy. The most suitable sensor to create drum pads is a piezo sensor. The piezoelectric effect is characterized as the capacity of specific materials for producing an electric charge concerning applied mechanical weight (Wheeler, 2014). The adjustment in capacitance of a sensor is directly proportional to ice thickness; not exactly the separation between the conductive cathodes (Saleh et al., 2006). The piezoelectric sensor shown in Figure 3 is made up of quartz material that generates an electric charge when pressure is applied to it, used for designing the drum pad. A cushion of an electronic drum has a vibration responsive plate mounted on a pad and its upper surface a moderately slim support cushion (Martin and Nystrom, 2006). A piezoelectric sensor produces the proportional analog value of voltage when vibration is applied to it. The sensor has a lot of receptive to an outer striking power for creating a simple analog signal, each analog pulse represents one beat of the particular instrument (Wolf et al., 2008). Piezo sensor produces analog signal based on pressure and velocity which is very essential to make the digital drum kit. The limitation of piezo the piezoelectric sensor produces a very low voltage (in millivolts) requires an amplifier. To devise for the impact of external parameters on piezoelectric sensors associated with a charge amplifier, a slice is utilized to frame the input capacitor of the intensifier (Wolf et al., 2008). Another limitation is the requirement of calibration which shoots up the cost.

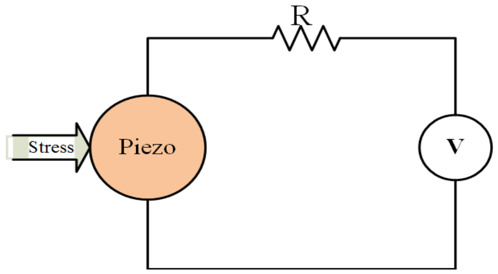

Figure 3. Piezo sensor

2.3 Processing circuit

The processing circuit acts as the brain of the drum machine. It accepts information from the pad via a piezo sensor, in the format of which channel of the drum is hit and velocity of the stick hits and controls the sound data. The processing system is designed with application-specific integrated circuit (ASIC), and microcontroller-based music system is slower due to its predefined latency. Xilinx Spartan 6 FPGA Xc6xlx9 in Figure 4 used for the processing can configure with JTAG programmer.

Figure 4. Spartan 6 FPGA development board

2.4 MIDI

MIDI is the communication protocol used for connecting musical instruments, controllers, consoles, and computers with each other (Liu et al., 2018). MIDI is not a modulation technique; it is the format for the musical data (Jadhao and Patel, 2020). MIDI preferred recording and playing back music from a digital instrument. MIDI carries a series of instructions like the notion, volume, temp, note on/off signal. Instead of real sound, MIDI instructs the machine to playback according to the instruction mapped, since each MIDI note can be edited independently which does not affect the overall performance.

A MIDI format presented in Figure 5 consists of three bytes first byte denotes the channel and note information, the second byte denotes note number and the third byte denotes velocity information. Component of the MIDI interface presented in Table 3 shows; channel information in the MIDI data is the independent path for connecting multiple MIDI devices to the control unit, channel data is 4 bit wide can connect to 2^4 = 16 different devices (Wicaksono and Paradiso, 2020). Firstly 4 bit of the MIDI data is the flag bit for the note on or off. The second byte gives the information about the note number which note has to be played. Note number in the MIDI is 8 bits wide thus 2^8 = 256 notes can be transmitted or received. The last byte is for velocity information, which refers to how hard the note is played. Communication thoroughly MIDI is serial asynchronously. Mostly Universal asynchronous receiver-transmitter (UART) is used for communication of the MIDI data (Gupta et al., 2020; Wheeler, 2014). It uses a five-pin connector in Figure 6 specially made for MIDI signal communication (Huber, 2007) now day's universal serial bus (USB) connectors are popular to communicate through all modern consumed devices like computers and mixers.

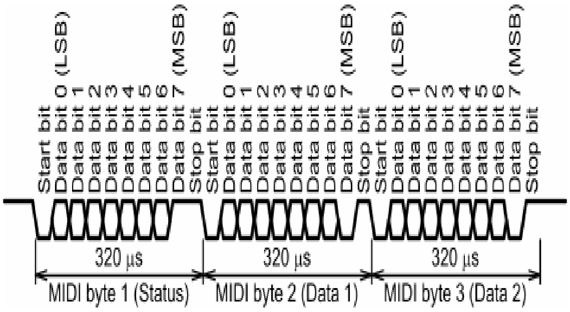

Figure 5. MIDI data format

Table 3. MIDI connector component

Connector	Cable
1	No connection
2	Shield
3	No connection
4	Voltage reference line
5	Data line

Figure 6. MIDI connector

2.5 Output section

The output section of the design consists of a digital to analog converter (DAC) for converting an analog value to play for the speaker. Depending on the speed of the processing, 2 types are tested. First one is the I2S module as an inter IC sound protocol in Figure 7 to convert the digital bit to the analog audio signal. The output of FPGA converted to PCM audio signals or audio track fed into I2S module which converts, digital PCM signal into 16 bit parallel in its data register, amplifies through class D amplifier in Figure 7. I2S module required audio samples which convert into analog signals.

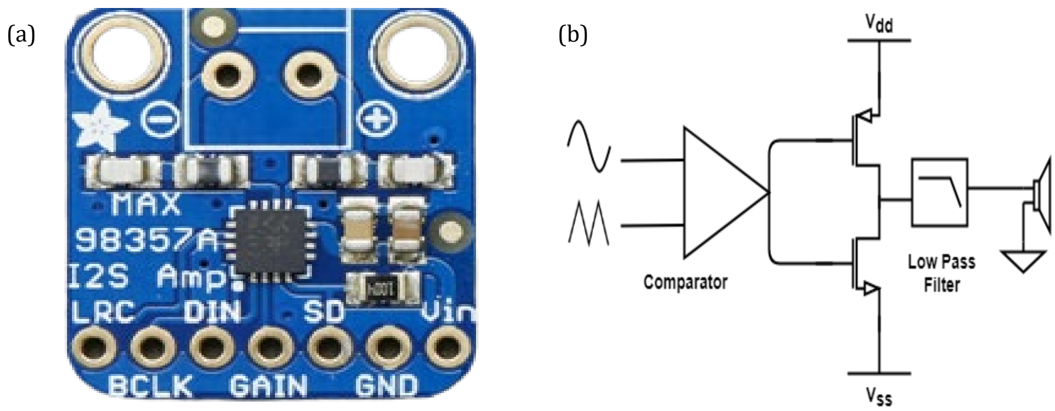

Figure 7. (a) I2S breakout board (b) class D amplifier

3. RESULTS AND DISCUSSION

Serial data received through ADC are formatted according to MIDI protocol and outputted to the MIDI transmitter. MIDI controller produces drum sound according to the hit of the drum pad and piezo sensor.

MIDI controller logic is designed with Very High Speed Integrated Circuit (VHSIC) Hardware Description Language (VHDL) and implemented over the spartan6 FPGA board. When the pad receives strikes creates sound, some sound samples are re-created. Before the physical testing 4 hardware generates sounds undergo through MATLAB simulation, the process justifies the initial design and provides a reference output with which to compare the output of the physical testing stage. The drum kit requires PCM data of 8 bits with 44100 sampling frequency with only a mono channel. Figure 8 shows audio data plots of a snare, bass, h-hat, and crash cymbal respectively.

The communication between ADC and FPGA has been established with a UART. A Moore state machine diagram shown in Figure 9 implements the serial receiver in 5 states to detect serial bit. On detect of start bit 'detect=1' control move from 'Idle' to 'Start Bit' state and confirming the start bit by counting the half time of the baud cycle ticks. When the start bit is survived or confirmed the third state, 'Rx Data' activates the shift register on the clock cycle of baud ticks. When all the bits are shifted and the state machine goes to the 'Stop Bit' state, they confirms the stop bit detected. As the stop bit is detected, 'Clean Up' state eliminates the values in the register and again machine is set to 'Idle' state for the next sequence.

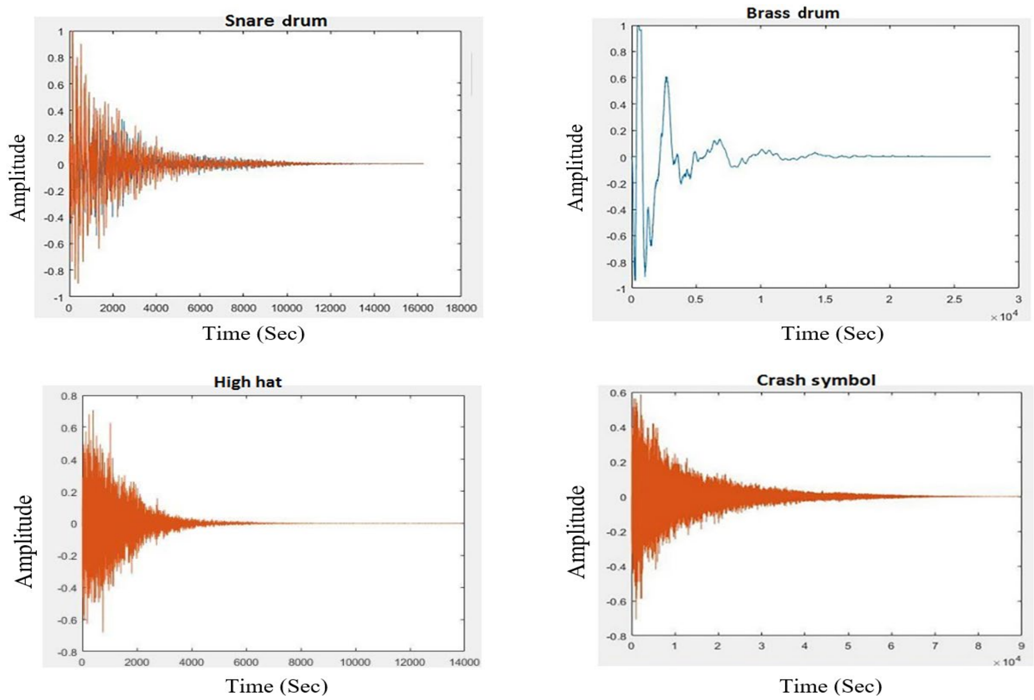

Figure 8. The plot of audio data in MATLAB

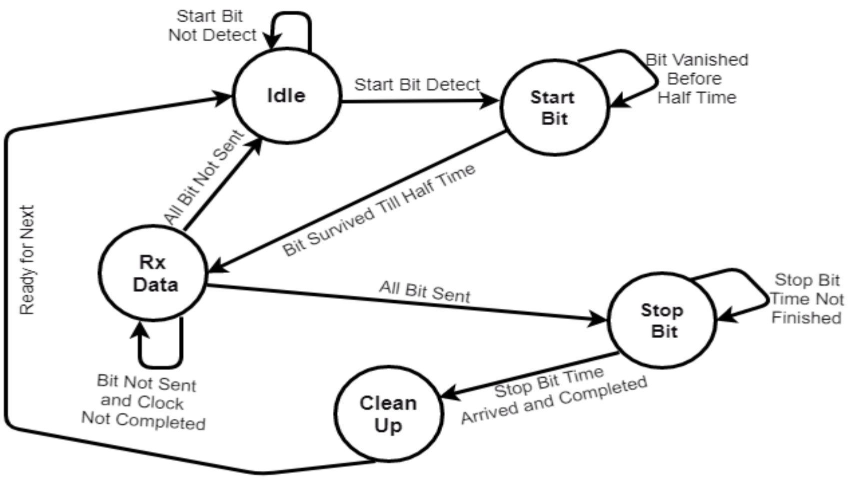

Figure 9. State diagram for universal asynchronous transmitter and receiver

The state machine is shown in Figure 10 to implement the transmission of MIDI data to the UART transmitter. The state machine consists of 4 states i.e. Idle state, start state, transmit data is equal to state, and stop state. Default state in 'idle' busy flag =0 (reset position). When Tx bit is equal to 1 indicates data, which is ready to be transmitted, and the state changes from 'idle' to 'start' state. A busy flag is set to 1, which denotes that the machine is busy transmitting the data. As soon as the baud tick=1, the state changes from 'start' to 'transmit data' state. The transmit data state is a data sending state which triggers the shift register to shift the data on the baud tick=1. The transmit data state iterates until the stop bit is encountered, in this iteration the whole data shifting process completes. As the stop bit=1 is detected, the state changes from 'transmit data' to 'stop' state, indicating that the full data word is transmitted and ready for sending new data. As the baud ticks, the 'stop' state again changes 'idle', and the set busy flag is set to 0; denoting that the machine is again ready for transmitting the new data.

MIDI data format is plotted on digital storage oscilloscope (DSO) and presented in Figure 11. The probes of the scope are connected on PIN 86 of the FPGA development board where it is directly connected to the Rx of the USB Joint Test Action Group (JTAG) on board, the Figure shows some noise because the oscillator in the FPGA board does not produce a smooth square wave.

I²S breakout module composed of the class-D amplifier and inbuilt low pass filter generates an equivalent analog signal to the output wave of MIDI transmitter. A speaker at the output generates noise-like data output from the I²S board as shown in Figure 12.

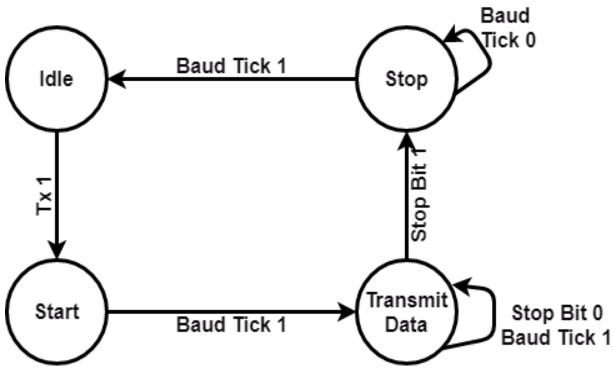

Figure 10. State diagram for MIDI transmitter

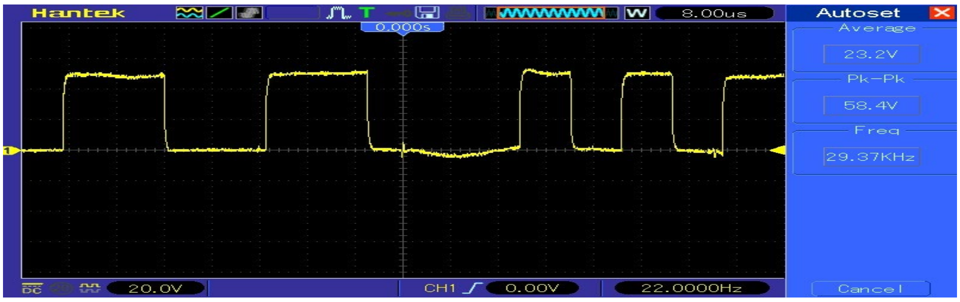

Figure 11. MIDI data on digital storage oscilloscope

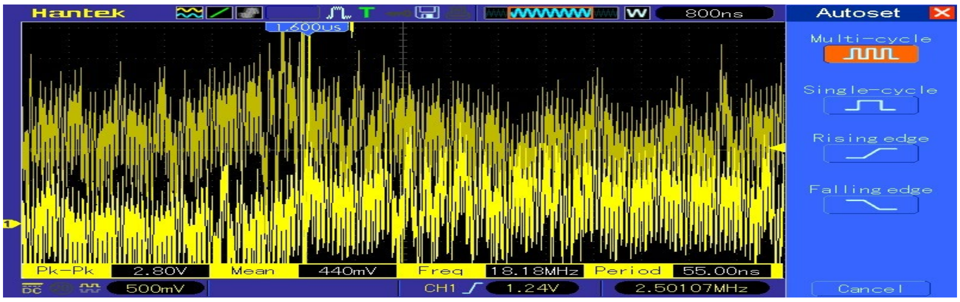

Figure 12. I²S output on digital storage oscilloscope

Table 4 presents the delay of the existing electronic/digital drum kit (Jack et al., 2018). A snare mechanism synthesis is 11.15 ms due to its static delay. Digital music synthesized implemented by (Zhou et al., 2021) with FPGA accepts real-time data from MIDI achieve latency of 12-24 ms which is higher than the threshold limit of 10 ms. A low-cost Arduino-based musical instrument: Fliperama (Vieira and Schiavoni, 2020) achieves a delay of 0.3 ms. The reconfiguration feature of FPGA with signal processing algorithm achieves minimum latency compared to existing instruments. MIDI transmitter module implemented with Spartan 6 FPGA requires 43 units logic block and 12 units IO block shows the delay associated with critical path 11.10 ns, consume power 14 mw. The processing module adds a delay of 9.987 ns and additional delay due to the I²S board and speaker and wires being 18 ns, total delay 11.10+9.987+18=39.87 ns. FPGA is preferred for faster processing. In this work, the achieved latency of 39.087 ns is in the range of transmitting audio over the internet, acceptable for playing prerecorded sound of the drum kit. There can be limitation in case of live concert where latency is variable for varies 200-1000 ms depending on distance from performer.

Table 4. Comparision of latency of a musical drum kit

Delay	References
11.15 ms	Williams et al. (2021)
300 µs	Jack et al. (2018)
12-24 ms	Zhou et al. (2021)
900 µs	Takemura et al. (2015)
41-52 ms @ 40Hz	Wang et al. (2021)
21-27 ms @ >80Hz	Wang et al. (2021)
Proposed	**39.087 ns**

4. CONCLUSION

In modern music electronics drum can help the musician to solve their problem of huge gear carrying, setting the drum on every gig tuning, and many more. Microcontroller-based drum kits are popular and cheap but they have high latency, whereas ASIC-based drums are very expensive. An FPGA-based drum kit can be a choice. In this work, the processing of the signal received from the sensor serially implemented with UART and transmission of PCM signal bias MIDI transmitter has been implemented with FPGA. The total processing delay from input to end is 39.087 ns, which is suited architecture for Internet of Musical Things (IoMusT) devices.

REFERENCES

Askenfelt, A., and Jansson, E. (1987). From touch to string vibrations—the initial course of the piano tone. *The Journal of the Acoustical Society of America*, 81(S1), S61.

Bianchi, V, Bassoli, M., and De Munari, I. (2020). Comparison of FPGA and Microcontroller Implementations of an Innovative Method for Error Magnitude Evaluation in Reed-Solomon Codes. *Electronics*, 9(1), 89.

Dinulică, F., Bucur, V., Albu, C. T., Vasilescu, M. M., Curtu, A. L., and Nicolescu, N. V. (2021). Relevant phenotypic descriptors of the resonance Norway spruce standing trees for the acoustical quality of wood for musical instruments. *European Journal of Forest Research*, 140, 105-125.

Friberg, A., and Sundberg, J. (1995). Time discrimination in a monotonic, isochronous sequence. *The Journal of the Acoustical Society of America*, 98(5), 2524-2531.

Fujii, S., Hirashima, M., Kudo, K., Ohtsuki, T., Nakamura, Y., and Oda, S. (2011). Synchronization error of drum kit playing with a metronome at different tempi by professional drummers. *Music Perception*, 28(5), 491-503.

Gupta, A. K., Raman, A., Kumar, N., and Ranjan, R. (2020). Design and implementation of high-speed universal asynchronous receiver and transmitter (UART). In *7th International Conference on Signal Processing and Integrated Networks*, pp. 295-300. Noida, India.

Huber, D. M. (2007). *The MIDI Manual: A Practical Guide to MIDI in the Project Studio*, 3rd, Waltham, MA: Focal Press, pp. 200-250.

Jack, R. H., Mehrabi, A., Stockman, T., and McPherson, A. (2018). Action-sound latency and the perceived quality of digital musical instruments: Comparing professional percussionists and amateur musicians. *Music Perception*, 36(1), 109-128.

Jack, R. H., Stockman, T., and McPherson, A. (2016). Effect of latency on performer interaction and subjective quality assessment of a digital musical instrument. In *Proceedings of the Audio Mostly*, pp. 116-123. Norrköping, Sweden.

Jadhao, K. M., and Patel, G. S. (2020). Hardware architecture of digital drum kit using FPGA. In *IEEE International Conference on Advent Trends in Multidisciplinary Research and Innovation*, pp. 1-4. Buldhana, India.

Liu, Q., Ba, S., Wu, L., Huang, J., and Li, H. (2018). Virtual dulcimer auxiliary teaching system based on musical instrument digital interface. In *Proceeding of International Symposium on Educational Technology,* pp. 82-86. Osaka, Japan.

Magnusson, T., and Mendieta, E. H. (2007). The acoustic, the digital and the body: A survey on musical instruments. In *Proceedings of the 7th International Conference on New Interfaces for Musical Expression*, pp. 94-99. New York, USA.

Martin, A. J., and Nystrom, M. (2006). Asynchronous techniques for system-on-chip design. *Proceedings of the IEEE*, 94(6), 1089-1120.

Saleh, R., Wilton, S., Mirabbasi, S., Hu, A., Greenstreet, M., Lemieux, G., and Ivanov, A. (2006). System-on-chip: Reuse and integration. *Proceedings of the IEEE*, 94(6), 1050-1069.

Takemura, K., Yoshimi, A., Nishikawa, H., Tanaka, A., and Douseki, T. (2015). Batteryless 900-µs-latency FM transmitter powered by piezoelectric generator for wireless electronic drums. In *Proceeding of the IEEE Sensors*, pp. 1-4. Busan, South Korea.

Vieira, R., and Schiavoni, F. (2020). Fliperama: An affordable Arduino based MIDI controller. In *Proceedings of the International Conference on New Interfaces for Musical Expression,* pp. 375-379. Birmingham, UK.

Wang, L., Noordanus, E., and van Opstal, A. J. (2021). on a Estimating multiple latencies in the auditory system from auditory steady-state responses single EEG channel. *Scientific Reports*, 11(1), 2150.

Wheeler, P. (2014). The Musical instrument digital interface (MIDI): The digital organ for organists and non-organists. *The Journal of the Acoustical Society of America*, 135(4), 2245.

Wicaksono, I., and Paradiso, J. A. (2020, July). KnittedKeyboard: Digital knitting of electronic textile musical controllers. In *Proceedings of the International Conference on New Interfaces for Musical Expression,* pp. 323-326. Birmingham, UK.

Williams, P., and Overholt, D. (2021). Design and evaluation of a digitally active drum. *Personal and Ubiquitous Computing*, 25(4), 783-795.

Wolf, W., Jerraya, A. A., and Martin, G. (2008). Multiprocessor system-on-chip (MPSoC) technology. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 27(10), 1701-1713.

Zhou, H., Zhou, J., and Chang, M. (2021). *conFFTi: an FPGA music synthesizer* (18-500 Final Report). Electrical and Computer Engineering, Carnegie Mellon University.