International Journal of Signal Processing, Image Processing and Pattern Recognition
Vol.9, No.10, (2016), pp.161-170
http://dx.doi.org/10.14257/ijsip.2016.9.10.15

The Impact of FPGAs on Speech Processing in The 21st Century

Aaron Roopnarine[1] and Marcus Lloyde George[2]

[1]Department of Electrical and Computer Engineering
University of the West Indies
St. Augustine, Trinidad and Tobago
[2]Department of Electrical and Computer Engineering
University of the West Indies
St. Augustine, Trinidad and Tobago
[1]aaron.roopnarine1@my.uwi.edu, [2]Marcus.George@sta.uwi.edu

Abstract

Speech processing involves making changes to speech signals for a required application. Therefore it is very important in the area of communications. In the 21st century the use of FPGAs has been more prevalent in many areas. In this paper, the impact of FPGAs on speech processing is evaluated. Speech processing algorithms are classified and the current standards are overviewed. The role of micro-processors in speech processing is evaluated. Current implementations of FPGAs in speech processing are outlined. Finally, aspeech processing application that does not use FPGAs is evaluated on the basis that itis implemented using FPGAs.

Keywords-Speech Processing*: FPGAs; Speech processing with FPGAs; Speech processing applications; Speech processing implemenations; Impact of FPGAs*

1. Introduction

Speech processing is the study of making changes to speech signals, which are usually converted to electrical signals, for the required application [20] and [1]. According to [1] these applications include

- Understanding speech as a means of communication
- Representing speech for transmission and reproduction
- Speech Synthesis
- Analyzing speech for extraction of information e.g. discovering some physical characteristics of the person talking or recognizing spoken words of a person

Off-the shelf micro-processors and Discrete Signal Processors (DSPs) are usually used for speech processing applications. However, Field Programmable Gate Arrays (FPGAs) have advanced significantly of the past years. These devices outperform DSPs in terms of throughput and cost and so they are being used more frequently for speech processing applications [2].

This research paper seeks to investigate how far FPGAs have penetrated the speech processing field and its potential to further advance the speech processing field. Therefore this paper first seeks to identify the current classifications of speech processing algorithms. After, the standards used in speech processing are outlined. A literature survey is used to investigate microprocessor-based speech processing applications and the adequacy of microprocessors in those applications is evaluated by a look into the extent to which FPGAs have been used in the 21st century. This is followed by a look into the possibility of

ISSN: 2005-4254 IJSIP

using FPGAs to achieve a speech processing application that is not currently used by FPGAs.

2. Classification of Speech Processing Algorithms

Many different speech processing algorithms exist. There are also many different applications of speech processing. These algorithms are classified within a specific general application of speech processing. For example, the way algorithms are classified for the application of speech recognition may be different from the way algorithms for the application of speech coding are classified.

Speech codingis the process of transforming the digitized speech signal into a representation for efficient storage and transmission of speech [1]. Therefore speech coders essentially compress the inputted bit-stream of data to produce an encoded bit stream whose bit rate is less than the inputted bit rate [5].The following is diagram shows the general process of speech coding.

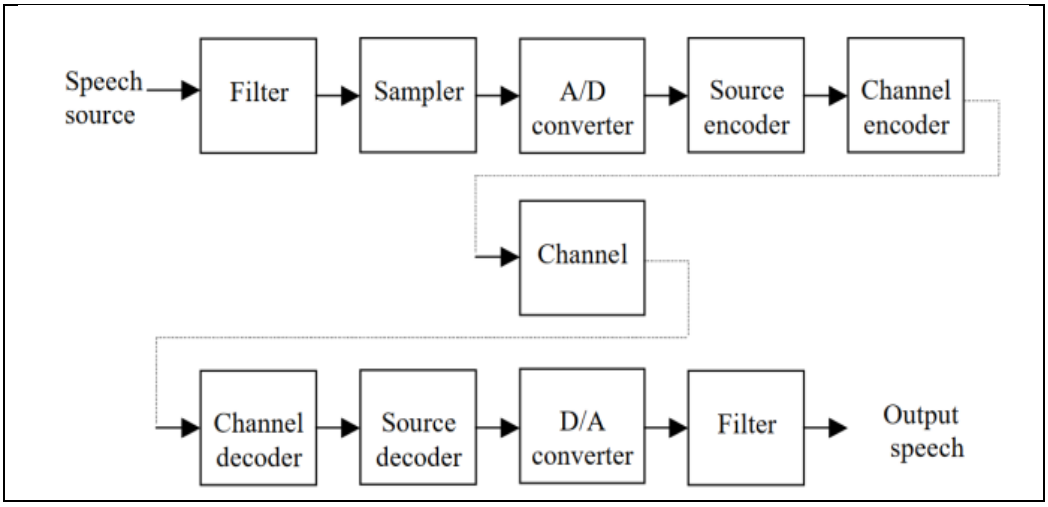

Figure 1. A Summary of the Speech Coding Process [5]

The speech signal is passed through a filter to eliminate the aliasing phenomena. The sampler is used for discrete time conversion and the A/D converter quantizes the amplitudes resulting in digital speech being achieved. The outputted signal is assumed to be uniformly quantized. The source encoder reduces the inputted bit rate according to the algorithm specified. The channel encoder provides error detection before it passes through the channel for transmission. At this point the coded signal is achieved. The decoding process involves passing the signal through the channel decoder which corrects the error in accordance with the error protection mechanism employed by the channel coder and the source decoder which has at its output the original digital speech. The D/A converter and the filter converts the digital signal to a continuous time signal [5].

Speech recognition is the process of extracting linguistic information from speech signals in the area of human to machine communication [1]. Bear in mind that since each general application of speech processing is advancing, the means with which each area is being classified will change with time also. This section of this paper will address the classification of speech coding algorithms as much attention is placed on this aspect of speech coding nowadays. The following sub-sections will explain the classifications used for speech coding algorithms that currently exist as suggested by and explained by [5].

A. Waveform Coders

This class of speech coders aims to preserve the original shape of the inputted signal's waveform. As a result, coders of this class could be applied to generally any signal source. The performance of the waveform coder drops as the encoded bit rate decreases.
Recall that all speech coders are designed to reduce the bit rate of the inputted bit stream of data to lower values. The inputted bit stream of data is always referenced to 128 kbps. Therefore each speech coding algorithm will have a different encoded bit rate for the same bit rate of inputted data. The following table shows general classifications of speech coding techniques based on their encoded bit rate [5].

Table 1. The Classification of a Speech Coder for the Respective rRange of Bit Rates of the Encoded Bit Stream Used

Classification assigned	***Encoded bit stream's bit rate range***
High bit rate	>15kbps
Medium bit rate	5 to 15kbps
Low bit rate	2 to 5 kbps
Very low bit rate	<2kbps

The encoded bit rate is commonly called the bit rate of the speech coder. Waveform coding is used for high bit rate coding (>32 kbps) [5].
Signal to Noise Ratio (SNR) is another metric that is used to determine the quality of the coded data by the waveform coders.
The following speech coding algorithms are classified as waveform coders:

- Pulse Code Modulation (PCM) and its variants
- Adaptive PCM (APCM)

B. Parametric Coders

This class of speech coders does not aim to preserve the shape of the inputted waveform. So Signal to Noise Ratio (SNR) cannot be used as a metric to determine the quality of this class of speech coders. The speech signal is assumed to be generated from a modelled system that is controlled by certain parameters. Therefore the encoded bit stream must have the parameters of the inputted bit stream. Consequently the encoding process of this class of speech coders involves determining these parameters. Therefore the quality of the encoded data depends on the accuracy and sophistication of the model used. So increasing the bit rate does not translate to a better quality [5].
This form of coding is used for low bit rates (2 to 5kbps).

The following speech coding algorithms are classified as parametric coders:

- Linear Prediction Coding (LPC)
- Mixed Excitation Linear Prediction (MELP)

C. Hybrid Coders

This class of speech coders combines the methodologies employed in parametric coders and waveform coders. Therefore hybrid coders use a speech production model and the parameters of the model are determined in the encoding process. However additional parameters are optimized such that the decoded waveform of the time domain representation of the original signal [5].
This form of coding is used for medium bit rates.

The following speech coding algorithms are classified as hybrid coders:

- Code Excited Linear Prediction (CELP) and its variants

D. Single Mode Coders

This class of speech coders applies a fixed encoding mechanism at all times. Therefore the encoded bit rate is constant [5].

The following speech coding algorithms are classified as single mode coders:

- Pulse Code Modulation (PCM)
- Regular-Pulse-Excited Long-Term Prediction (RPE-LTP)

E. Multimode Coders

This class of speech coders has varying encoded bit rates with each mode having fixed bit rates. The mode selected is based on the local statistics of the inputted speech signals. Multimode speech coders could further be subdivided into open loop and closed looped coders. The encoded outputs of each mode is taken into account for the final decision in closed loop multimode speech coding. The input signal is analyzed then a mode is selected for open loop multimode speech coding.

Since the encoded bit rate is flexible, the efficiency with which data is transferred is better as there should be a reduction in the average bit rate [5].

The following speech coding algorithms are classified as multimode coders:

- TIA IS96
- Variable Bit Rate CELP Coder
- Adaptive Multirate (AMR) Coder standardized by ETSI 1999 (ETSI AMR ACELP)

3. Overview of Existing Speech Processing Standards

The demand for speech communication is increasing. As a result, speech coding technology has received much attention with regard to standardization [5].The speech processing standards that exists with regard to speech coding is with respect to the particular algorithm/technique used for speech coding. These standards exist as a reference for everyone.

The following sub-sections explains the principles behind speech coding standards that currently exist.

F. Pulse Code Modulation (PCM)

PCM is a standard that is classified under waveform coders and single mode coders [5]. The international standard for non-uniform PCM is the ITU-T G.711 [7]. In this algorithm, each speech sample is quantized using a logarithmic scale which allows the lower amplitudes to have more quantization levels than the higher amplitudes [3].

Non-uniform PCM yields better performance than uniform PCM since the signal to quantization noise ratio (SQNR) for uniform PCM is low at low amplitudes but for Non-uniform PCM SQNR is equal across all amplitudes. Hence non-uniform PCM is more commonly used.

The ITU-T G.711 standard governs the use of two types of non-uniform PCM: μ law and A law. Each sample of is represented using 8 bits. The following diagram shows the format of a sample of data.

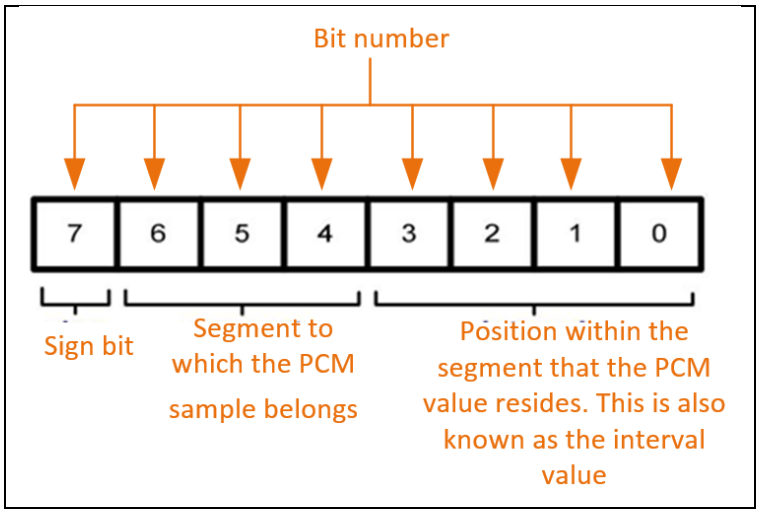

Figure 2. The Byte Format for the Representation for a Sample of Data for Non-Uniform PCM [3]

PCM is used mostly because of its simplistic nature which effectively results in low algorithmic delay and decoded speech quality. However, the encoded segment has a relatively high bit rate [3].

G. Differntial Pulse Code Modulation (DPCM)

This technique codes the difference between adjacent samples. According to [5], a prediction error is determined through a closed loop mechanism within which it is also quantized and combined with a predicted value. So the current value of the sample depends on previous samples rather than just the absolute value of the sample itself. Therefore DPCM is used when the sources changes slowly so that the samples correlate. This makes it possible for the DPCM coded sequence to occupy less bits than a PCM coded sequence improving the efficiency of transmission.

H. Adaptive Pulse Code Modulation (APCM)

APCM is a standard that is classified under waveform coders. It is the process where the step size vary based on the changes to the input signal [4]. These are used when the signals inputted are not stationary. If the non-stationary signal has a wide dynamic range, it is more efficient to use an APCM technique instead of a fixed quantizer since the SNR is greatly improved [5]. There are three types of APCM techniques that are covered in this section as explained by [5]:

- Forward Gain Adaptive Quantizer
- Backward Gain Adaptive Quantizer
- Adaptive Differential Pulse Code Modulation (ADPCM)

1. Forward Gain Adaptive Quantizer

In this speech coder, the gain level of the input sequence to be quantized is accurately controlled, but side information is sent to the decoder when the decoding process is warranted. The gain is controlled so as to normalize the amplitudes inside the frame so that high amplitude and low amplitude frames could be quantized using the same fixed quantizer making the process more optimal.

2. Backward Gain Adaptive Quantizer

In this speech coder, the gain level is estimated based on the output sequence of the quantizer. Therefore the gain does not need to be explicitly retained or transmitted. However, if there is an error in a gain value calculated, this error could be propagated to other samples. The gain is controlled so as to normalize the amplitudes inside the frame so that high amplitude and low amplitude frames could be quantized using the same fixed quantizer making the process more optimal.

3. ADPCM

ADPCM is a standard that is classified under waveform coders and single mode coders [5]. It codes the difference between two consecutive samples of PCM [4]. The principle behind this technique is using past values to determine future values so the error signal could be determined.

The international standard for ADPCM is the ITU-T G.726. Since this is technique is a form for Differential PCM, it has the advantages of PCM with a reduced bit rate. The adaptive aspect of ADPCM further adds to the efficiency of compression as it controls the gain in accordance with the time varying nature of the input.

Depending on the number of bits used in this technique for representing the coded data, the bit rate may differ *e.g.*, bits represented using 5, 4,3 and 2 bits correspond to bit rates of 40 kbps, 32 kbps, 24 kbps and 16 kbps respectively [4]. According to [8], if the encoded bit rates for 8 bit non-linear PCM is compared with that of ADPCM in 32kbps operation, there is a reduction of a factor of 2 showing the improvement in the efficiency of compression when ADPCM is used.

I. MPEG Audio

The standards for this technique could be found in ISO/IEC 11172-3 and ISO/IEC 13818-3 [10]. For MPEG Audio only Layer 1, 2 and 3 standards apply.

The following are the steps for compression for this MPEG Layer 1 and Layer 2 coding as explained by [9]:

- A QMF filter bank is used to transform the inputted signal into 32 sub-band signals that are uniformly distributed of frequency. These sub-band signals could be viewed as critically down sampled.
- The sub band signals are grouped into an allocation frame. There are 384 and 1152 sub-band samples per group for MPEG Layer I and II respectively.
- The signals are then quantized using APCM so as to achieve the MPEG-1 bit stream.
- For decoding, the encoded bit stream is decoded into the sub-band samples. These sub-band samples are then fed into an inverse QMF filter bank

MPEG Layer 3 coding uses a power law quantizer which is very similar to the non-linear PCM techniques discussed. The quantized values are then Huffman coded which is a form of lossless compression. The best gain for a given block, bit-rate and output from the perceptual model is usually done by two nested iteration loops [9].

J. Low Delay CELP (LD- CELP)

This is a hybrid coder. The parameters of the model used for the coder are selected carefully to facilitate fixed-point implementation. This coder consists of a codebook that is searched during encoding to locate the best codevector for a particular speech sub-frame. The ITU-T G.728 governs the standards for this speech coding technique.

This process involves passing the inputted speech signal through a perceptual weighting filter. The resultant signal is thin combined with the zero-input response and gain to produce a target vector. The optimal codebook indices are the encoded bits. The

impulse response of the cascaded filters are also accounted for through the Linear Prediction Coefficients (LPCs). The bit stream is represented using 10 bits of the excitation index transmitted every 0.625ms and so a bit-rate of 16 kbps is achieved. So in summary this technique involves comparing the locally decoded signal against the original signal and the coder parameters are selected such that the mean-squared weighted error between the original and reconstructed signal is minimized [11].

The design is highly robust against channel errors. Therefore this technique achieves low delays (about 2ms) while maintaining toll quality speech at a relatively low bit-rate of 16 kbps. However the complexity of this coder is higher when compared to other low bit rate coders.

K. Algebraic CELP (CS- ACELP)

This is a hybrid coder.The ITU-T G.729 governs the standards for this speech coding technique. The most distinct aspect of this speech coder is that the structure of the codebook used involves the excitation codebook vector. The inputted signal is sampled at a sampling rate of 8000 Hz is used. The speech signal is analyzed for speech frames of 10ms corresponding to 80 samples. The inputted frame is divided into two 5ms sub-frames which allows better tracking of the pitch andgain parameters and reduces the complexity of the codebooksearches [11].

The encoding principle of this technique includes five significant stages: the pre-processing stage, the LP analysis stage, the open-loop pitch search, the closed-loop pitch search, and the algebraic codebook search [12].

This increased complexity of this algorithm boosts the quality of synthetic speech. This format was intended to be computationally efficient at the expense of small inaccuracies such as in the area of short pitch areas [5].

4. Literature Survey

This section evaluates some speech processing applications for which micro-processors are used. If there is also a proposed implementation of the solution using FPGAs, these solutions are compared against each other.

With regard to ADPCM implementation, [13] shows the use of micro-processors to achieve this. The implementation was successful and efficient results were obtained. However, the context with which this method was proposed supports the preference to FPGA systems. In order to optimize the output from the micro-processor, a hardware based approach was taken which is the basis of FPGA design. Additionally, the implementation had concerns of cost even though the initial aim was to reduce cost by making the discrete signal processor (DSP) unnecessary for certain applications. FPGAs actually been successfully implemented for ADPCM coding in [4] and its implementation required significantly less computational cycles than that of a software implementation for micro-processors. So, for the same application, FPGAs had less latency than if micro-processors were used.

From the literature surveyed, there has been implementations of sections of the MPEG coding process via FPGAs to improve the efficiency of the sections of that process e.g. in [14] which uses the FPGA for the modified discrete cosine transform (MDCT) for MPEG layer 3 applications for the high data throughputs. This effectively improves the speech coding process. These findings point to the possibility that FPGAs may not necessarily fully replace micro-processors but could be used together with them to improve the speech coding process. Therefore, they may not completely replace micro-processors.

With regard to CS-ACELP there are micro-processor based implementations as in [16] as well FPGA implementations as with [15]. The implementation of the CS-ACELP speech compression algorithm of [15] which was implemented on FPGA had latency that was at least 200 times faster than software counterparts.

Since the beginning of the 21st century there has been a significant increase of use of FPGAs [18]. Even though the literature survey it was seen that the FPGA outperformed the micro-processor based counterpart for the same algorithms, FPGAs have not replaced them completely them in the area of speech coding. For instance, LD-CELP has been implemented using DSPs, a specialized micro-processor, in [17] but at this point, no implementations using FPGAs have been found.Additionally, due to the advancement in micro-processor technology as well, it is hard to say they will be irrelevant.

5. Converting to FPGA Implementation of Speech Processing

With regard to the general speech applications, FPGAs and micro-processors are being used. However, depending on the algorithm used, the current implementation might be specific to micro-processors or FPGAs. Microprocessors are easier to interface and program and so are mostly preferred. Even though FPGA implementation may be more complex, the resultant system would be more efficient.

Consider the general application of speech enhancement. FPGAs and micro-processors have been used in this application. Different algorithms are used to enhance speech. One algorithm that has been implemented for speech enhancement that has not been implemented using FPGAs is the DWT for De-Noising as implemented in [19]. The dsPIC, a micro-processor, has been used for this application [19].The clear issues seen in this implementation is the usage of memory and the practicality of the resultant system.

How can an FPGA be used to improve the performance of this system? The FPGA could essentially replace the dsPIC in the processing stage. It is even possible to interface an FPGA with MATLAB as done in [19]. The processing stage simply requires mathematical operations, such as matrix multiplication, which can be done more efficiently on the FPGA since it can be used to implement parallel processing and hence reduce the time taken for the computations. These operations could be done more efficiently through pipelining in FPGAs. An FSM could be designed to perform all control operations for this application [21]. Once the design steps of [19] is implemented through the steps suggested by [21], the re-implementation should be successful without any issues. Since FPGAs could perform calculations in fewer clock cycles than micro-processors, then the memory requirement may be reduced. Further, the implementation of the algorithm may be an issue for hearing aids which is a suggestion for its use. This is because the dsPIC may be too bulky and for mass production it would be too costly. The implementation of the algorithm on the FPGA could be transferred to an Application Specific Integrated Circuit (ASIC) which should be less bulky than a dsPIC and cheaper to implement in hearing aids.

The research done in [3-4] and [15] clearly shows that there is a significant reduction in latency when a software implementation is re-implemented using FPGAs. For example, [4] shows that with regard to ADPCM, the FPGA based encoder was 3621 times shorter than the latency of software encoder. These re-implementations suggested in [3-4] and [15] all fall under the speech coding aspect of the area of VoIP (Voice-Over Internet Protocol). This therefore begs the question: Could the QoS of VoIP be improved if other aspects of its architecture are implemented using FPGAs? The research done by [22] shows the implementation of a VoIP gateway where the whole system is mapped onto an FPGA. This implementation consists of both hardware and software segments. With regard to the hardware aspect, 50% of BRAM memories is used which is the most used resources, while the use of the other resources remains low. The achievements in [22] shows that FPGAs could be used to implement portable VoIP systems. From the accomplishments seen in the area of VoIP using FPGAs, it is possible to predict a better QoS for VoIP could be achieved by implementing FPGAs to perform functions of network elements that process VoIP data.

In the general speech processing application of encryption, FPGAs can be used as well. Current software based encryption schemes could be re-implemented in a more efficient

manner with FPGAs as computations usually require less clock cycles in FPGAs. Further, hardware based encryption are secure against malicious code unlike the software counterparts [23]. Therefore, this suggests that the use of FPGAs in speech encryption provides a very secure communication environment. Consider the FPGA based secured speech communication system suggested by [23]. Compression, watermarking and encryption are applied to the speech signals through the FPGA. The resultant system is secure against brute force attacks, parallel attacks, cold boot attacks, malicious code, man-in-the-middle attacks and differential cryptanalysis [23]. The delay for the proposed encryption method in [23] is about 5ns which is very small. This shows that FPGAs could be used for efficient secure communications.

The findings so far show that there is a lot of potential in the use of FPGAs and so we should expect even greater utilization of this technology in times to come.

6. Conclusion

The impact of the use of FPGAs in the 21st century has been investigated and it is clear that they are being utilized significantly more than before especially in the area of speech processing. With the advent of newer platforms such as 3D FPGAs and Multi-FPGA systems, the performance of speed processing systems can be further improved in the future, hence fostering further development of the speech processing arena in the 21st century.

Acknowledgements

The authors are grateful for all assistance afforded in development of this paper. Additionally, many thanks to the authors of the published papers that were referenced. These papers were very important in the conclusions made from this investigation.

References

[1] L. Rabiner, “Digital Speech Processing—Lecture 1: Introduction to Digital Speech Processing”, Department of Electrical and Computer Engineering University of California at Santa Barbara, **(2016)**.

[2] T. W. Hong and M. Yusoff, “Speech processing in FPGA with C-to-RTL compiler technology”, 2008 International Conference on Electronic Design, **(2008)**.

[3] G. Tomar and M. George, “Hardware Implementation of Pulse Code Modulation Speech Compression Algorithm”, AJMAHS, vol. 2, no. 1, **(2012)**, pp. 19-26.

[4] M. George and G. Tomar, “Hardware Implementation of Adaptive-Differential Pulse Code Modulation Speech Compression Algorithm”, International Journal of Smart Device and Appliance, vol. 2, no. 2, **(2014)**.

[5] W. Chu, “Speech coding algorithms”, Hoboken, N.J.: J. Wiley, **(2003)**.

[6] L. Rabiner, “Digital Speech Processing— Lecture 14A: Algorithms for Speech Processing”, Department of Electrical and Computer Engineering University of California at Santa Barbara, **(2016)**.

[7] International Telecommunication Union (ITU), Pulse Code Modulation (PCM), of Voice Frequencies. G.711, **(1972)**.

[8] International Telegraph and Telephone Consultative Committee (CCITT). 1990. 40, 32, 24, 16 kbit/s Adaptive-Differential Pulse Code Modulation (ADPCM). G726.

[9] MPEG", Mpeg.chiariglione.org, 2016. [Online]. Available: http://mpeg.chiariglione.org/standards/mpeg-1/audio. [Accessed: 28- Jul- 2016].

[10] "MPEG-1 and MPEG-2 Layer II Audio Encoding", Digitalpreservation.gov, 2016. [Online]. Available: http://www.digitalpreservation.gov/formats/fdd/fdd000338.shtml. [Accessed: 28- Jul- 2016].

[11] R. Salami, C. Laflamme, J. Adoul, A. Kataoka, S. Hayashi, T. Moriya, C. Lamblin, D. Massaloux, S. Proust, P. Kroon and Y. Shoham, "Design and description of CS-ACELP: a toll quality 8 kb/s speech coder", IEEE Transactions on Speech and Audio Processing, vol. 6, no. 2, **(1998)**, pp. 116-130.

[12] V. Togadiya, N. Shah and R. Rathod, “Comparative Review Between Celp And Acelp Encoder For Cdma Technology”, International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, vol. 2, no. 1, **(2013)**, pp. 603-608.

[13] A. Sharma and C. Ravikumar, "Efficient implementation of ADPCM codec", VLSI Design. Wireless and Digital Imaging in the Millennium. Proceedings of 13th International Conference on VLSI Design, **(2000)**.

[14] H. Kim, S. Kim and K. Chung, "FPGA implementation of unified kernel structure for MDCT/IMDCT in audio coding schemes", 2011 International SoC Design Conference, **(2011)**.

[15] G. Tomar and M. George, "Linear Prediction Analysis and Quantization for the Conjugate-Structure Algebraic-Code-Excited Linear-Prediction Speech Compression Algorithm", AJMAHS, vol. 2, no. 1, **(2012)**, pp. 35-46.

[16] R. Salami, C. Laflamme, J. Adoul, A. Kataoka, S. Hayashi, T. Moriya, C. Lamblin, D. Massaloux, S. Proust, P. Kroon and Y. Shoham, "Design and description of CS-ACELP: a toll quality 8 kb/s speech coder", IEEE Transactions on Speech and Audio Processing, vol. 6, no. 2, **(1998)**, pp. 116-130.

[17] M. L. George and G. S. Tomar, "Hardware Implementation of Adaptive-Differential Pulse Code Modulation Speech Compression Algorithm", International Journal of Smart Device and Appliance vol. 2, no. 2, **(2014)**, pp. 1-10.

[18] Z. Xueying, "Implementation of a 12.8 kbit/s LD-CELP speech codec", Proceedings of the 3rd World Congress on Intelligent Control and Automation (Cat. No.00EX393).

[19] S. Trimberger, "Three Ages of FPGAs: A Retrospective on the First Thirty Years of FPGA Technology", Proceedings of the IEEE, vol. 103, no. 3, **(2015)**, pp. 318-331.

[20] A. Uriz, J. Moreira, R. Hidalgo, P. Aguero, J. Tulli and E. Gonzalez, "An Implementation in dsPIC of a Denoising Algorithm Based on the Discrete Wavelet Transform", 2012 VI Andean Region International Conference, **(2012)**.

[21] R. G. Osuna, "Introduction to Speech Processing", 1st ed. Computer Science & Engineering- Texas A & M University, **(2016)**.

[22] M. George and G. Tomar, "Hardware Design Procedure: Principles and Practices", 2015 Fifth International Conference on Communication Systems and Network Technologies, **(2015)**.

[23] F. Abid, N. Izeboudjen, M. Bakiri, S. Titri, F. Louiz and D. Lazib, "Embedded implementation of an IP-PBX /VoIP gateway", 2012 24th International Conference on Microelectronics (ICM), **(2012)**.

[24] H. D. Kabir and S. Alam, "Hardware based realtime, fast and highly secured speech communication using FPGA", 2010 IEEE International Conference on Information Theory and Information Security, **(2010)**.

