International Conference on Computational Modeling and Security (CMS 2016)

Area and Power Optimized DTMF Detection
By using different FPGA’s

S Nagakishore Bhavanam¹, Dr. P. Siddaiah², Dr. P. Ramana Reddy³

¹Research Scholar, ²Associate Professor, ³Associate Professor, JNTUA, Ananthapuramu, A.P., India

Abstract

Rapid improvements in communication technologies, different applications in communications by FPGA technology. This paper explains about the Dual tone multi frequency detection (DTMF) by using different FPGA’s. Fast Fourier Transform (FFT) is one of the approach to detect the DTMF tones. It requires more hardware and also power consuming. So this paper proposes Split Goertzel Algorithm is optimized solution compared to Fast Fourier Transform. By using resource sharing approach (RSA) better optimized solution is achieved compared to Split Goertzel Algorithm. For functional verification Mentor graphics Modelsim software is used. Xilinx ISE 14.7 is used for Simulation & Synthesis respectively. To test the FPGA inside results the Xilinx Chopscope is used. The Spartan 3e, Spartan 6 & Zybo zynq 7000 series family FPGA boards are used in this paper and the corresponding results are showed as a table.

Keywords: DTMF, FFT, FPGA, RSA, Xilinx ISE;

1. Introduction

DTMF is widely used signaling standard in telecommunications. Each key you press on your phone generates two tones of specific frequencies, the fig.1 shows that, how the frequencies are organized. In embedded equipments separate Application Specific integrated circuits are needed for DTMF detection. For FPGA based DTMF detections we have to put external ASIC or to use some complex algorithms. FFT is one of the approach in DTMF detection by FPGA.

* S Nagakishore Bhavanam, Tel.: +91-9989541444;
  E-mail address: kishore.ece.anu@gmail.com

© 2016 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

Peer-review under responsibility of the Organizing Committee of CMS 2016
Architecture wise area efficient solution is achieved by using Split Goertzel algorithm. Further area optimized solution is achieved by using resource sharing approach.

![Fig.1 Keypad having low and high frequencies](image1)

### 2. Proposed Architecture Design

Mainly this paper divided into three phases. Each phase have three models. So totally this paper shows about nine models. Phase-I explains about DTMF detection by using Spartan 3e FPGA. Phase-II explains about DTMF detection by using Spartan 6 FPGA. Phase-III explains about DTMF detection by using Zybo Zynq 7000 series FPGA. All the Three phases have three different models. Those are FFT, Split Goertzel Algorithm & Resource Sharing Approach based ones.

**Phase-I: DTMF detection by using Spartan 3e Family FPFA**

*Model 1:*
FFT based DTMF detection by using Spartan 3e Family FPFA.

*Model 2:*
Split Goertzel algorithm based DTMF detection by using Spartan 3e Family FPFA.

*Model 3:*
Resource Sharing Approach based DTMF detection by using Spartan 3e Family FPFA.

**Phase-II: DTMF detection by using Spartan 6 Family FPFA**

*Model 1:*
FFT based DTMF detection by using Spartan 6 Family FPFA.

*Model 2:*
Split Goertzel algorithm based DTMF detection by using Spartan 6 Family FPFA.

*Model 3:*
Resource Sharing Approach based DTMF detection by using Spartan 6 Family FPFA.

**Phase-III: DTMF detection by using Zybo Zynq 7000 series family FPGA**

*Model 1:*
FFT based DTMF detection by using Zybo Zynq 7000 series family FPGA.

*Model 2:*
Split Goertzel algorithm based DTMF detection by using Zybo Zynq 7000 series family FPGA.

*Model 3:*
Resource Sharing Approach based DTMF detection by using Zybo Zynq 7000 series family FPGA.
Model 1:
FFT based DTMF detection by using Zybo Zynq 7000 series family FPGA.

Model 2:
Split Goertzel algorithm based DTMF detection by using Zybo Zynq 7000 series family FPGA.

Model 3:
Resource Sharing Approach based DTMF detection by using Zybo Zynq 7000 series family FPGA.

To implement FPGA based DTMF detection, First of all FFT based DTMF detection is implemented. The area & power results are analyzed. Second Split Goertzel Algorithm based DTMF detection is implemented and the area & power results are analyzed. Thirdly to achieve better optimized results of area and power, the resource sharing approach is used compared to FFT and Split Goertzel Algorithm. The proposed architecture is shown below.

![DTMF Detection Module](image)

The above figure can represent DTMF Detection General module. It consists of six modules such as: 4x4 Hex key pad (input), Dual Tone Multi Frequency test signal generator, Additive white Gaussian noise, Frequency Detection block, Magnitude or index estimator and Frequency to digit look-up table.

Hex Keypad can give an input to the DTMF module. It’s an external component. The signal from column is taken as an input. Row and display are the corresponding output signals. By using PC we can analyze the corresponding results. FTSG block generates that, the carrier frequencies necessary. It consists of two blocks those are FWS and DDS Core. This paper gets the output as a noise bits. This output can give as an input to FDM i.e. Frequency Detection Module. The Output of the block is indices that, magnitudes.

As per the scope of the project, there are 3 variants of FDB. Those are FFT core, Goertzel algorithm & Resource Sharing Approach. To test the FPGA inside results Spartan 3e, Spartan 6 & by Zybo Zynq 7000 series family FPGA boards are used.
3. Hardware Setup

To implement DTMF detection this paper explained about three different hardware setups. Those hardware families are Spartan 3e, Spartan 6 & Zybo Zynq 7000 series FPGAs. The below figures can represents the different hardware setups.

3.1. Hardware Setup for DTMF detection by using Spartan 3e Family FPGA

![Fig.3 Hardware Setup by using Spartan 3E family FPGA](image)

3.2. Hardware Setup for DTMF detection by using Spartan 6 family FPGA

![Fig.4 Hardware Setup by using Spartan 6 family FPGA](image)
3.3. Hardware Setup for DTMF detection by using Zynq 7000 series family FPGA

![Hardware Setup by using Zynq 7000 series family FPGA](image)

4. Simulation Results

Simulation is the imitation of the operation of a process or system over time. So it represents the operation of the system. The below figures can represent the simulation results for FFT, Goertzel and Resource sharing Approach for DTMF detection. Modelsim Xilinx Edition (MXE) is used for functional simulations for the designs.

The Reset (rst) will be 1 initially because to clear the previous values, unknown values, based on next rising edge of the clock operation will be going on. Master clock is for synchronization purpose. Hex_key value is a input. Hf_group_freq_word represents the high frequency & lf_group_freq_word represents the low frequency words. Tone 1 is a first DDS IP core output. Tone 2 is a second DDS IP core output. Sig_out represents combination of tone 1 and tone 2. Noise_bits represents the White Gaussian Noise. Noise_tones is a combination of sig_out and White Gaussian Noise. The FFT inputs are Xn_real it represents input is noise tones and Xn_imaginary we can give as Zero. Then the FFT output is Xk_real. Xk_real and Xk_img are the inputs to the Mag_out. Which is having high value is magnitude output.

Finally Found_digit represents the DTMF output. However by using FFT finding the magnitude of the input from hex_key pad which key as given input and finally got the output on found_digit out. Example you can observe on hex key value same as found digit out from the results below.

The fig.6 explains about the FFT based DTMF detection. Here we are giving Hex_Key as a input i.e 0001, 0010, 0011, 0100, 0101, 0110, 0111, 1000, 1001, 1010, 1011, 1100, 1101, 1110, 1111 as a binary format and 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F as a Hexadecimal format respectively. For the corresponding input we are getting corresponding output.
4.1 FFT based DTMF detection

The Fig.7 & Fig.8 represents the Phase increment value for the frequencies in terms of decimal and hexadecimal format.

For Ex: Choose the frequency = 697.
The decimal Value is 9354975.642 & Hex Value is 8EBEDF. How?

Sol: The sampling frequency $fs=5K$ Hz.
No. of samples $N = 2^7 = 128$.
Resolution = $fs/N = 5K/128 = 39.0625$.
For DDS input $p=26$ i.e fixed. So $2^{26}=67108864$.
Decimal Value for 697 = $(697 \times 2^{26}) / 5K$
= **9354975.642**.

Convert Decimal Value to Hexadecimal Value: go to calculator paste the value 9354975.642. Click on Hex. We are getting **8EBEDF** as Hex value.
Fig. 7 FFT based DTMF detection: Screenshot of high frequency group frequency word

Fig. 8 FFT based DTMF detection: Screenshot of low frequency group frequency word

Fig. 9 FFT based DTMF detection: Adding of white Gaussian noise to noise tones.
4.2 Split Goertzel Algorithm based DTMF detection:

The Goertzel algorithm is a digital signal processing (DSP) technique for identifying frequency components of a signal. While the general Fast Fourier transform (FFT) algorithm computes evenly across the bandwidth of the incoming signal, the Goertzel algorithm looks at specific, predetermined frequencies. Some applications require only a few DFT frequencies. Here by using Goertzel algorithm finding the magnitude of the input from hex key pad which key as given input and finally got the output on found digit out. You can observe hex key value same as found digit out.

The fig.11 explains about the Goertzel Algorithm based DTMF detection. Here we are giving Hex_Key as a input i.e 0001, 0010, 0011, 0100, 0101, 0110, 0111, 1000, 1001, 1010, 1011, 1100, 1101, 1110, 1111 as a binary format and 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F as a Hexadecimal format respectively. For the corresponding input we are getting corresponding output.
Fig. 12 Goertzel Algorithm based DTMF detection: Screenshot of high frequency group frequency word

Fig. 13 Goertzel Algorithm based DTMF detection: Screenshot of low frequency group frequency word

Fig. 14 Goertzel Algorithm based DTMF detection: Adding of White Gaussian noise to noise tones.
4.3 RSA based DTMF detection:

By using Goertzel algorithm finding the magnitude of the input from hex key pad which key as given input and finally got the output on found digit out. Example you can observe on hex key value same as found digit out.

Resource sharing is the assignment of similar block of operations (for example, +) to a common netlist cell. Netlist cells are the resources—they are equivalent to built hardware. Resource sharing reduces the amount of hardware needed to implement block operations. Without resource sharing, each operation is built with separate circuitry.
Fig. 17 RSA based DTMF detection: Screenshot of high frequency group frequency word

Fig. 18 RSA based DTMF detection: Screenshot of low frequency group frequency word

Fig. 19 RSA based DTMF detection: Adding of White Gaussian noise to noise tones.
From the figures 10, 15 & 20 we observe on hex key value same as found digit out from the results above. Clearly for corresponding inputs 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F (as a Hexadecimal format) we are getting corresponding output.

5. Chipscope Results

ChipScope is extremely useful because they allow viewing and manipulating signals directly from hardware during run-time. Since they are real VHDL modules and net lists, they get incorporated, synthesized, and implemented into design just like any other VHDL code that is written.
The fig.21 represents the FPGA based DTMF detection by Various FPGAs like Spartan 3e, Spartan 6 and Zynq 7000 series family FPGAs. Here for the Corresponding input “A” we are getting corresponding output as “A”. i.e nothing but for binary input “1010” we are getting chipscope output “1010” respectively. So the key is detected successfully by hardware. Similarly all 9 modules got these type of tone detection by hardware.

6. Results Analysis

The below table represents various comparisons between three phases of FPGAs for area and power.

| TABLE: I comparison between three models of Spartan 3e Based FPGA |
|---------------------|---------------------|---------------------|
| Name | Phase-I | Phase-II | Phase-III |
|------|--------|--------|--------|
| Selected Device | 3s500efg320-4 | 3s500efg320-4 | 3s500efg320-4 |
| Number of Slice Registers | 2700 out of 4656 57% | 1640 out of 4656 35% | 1322 out of 9312 14% |
| Number of Slice LUTs | 4303 out of 9312 46% | 2904 out of 9312 31% | 2207 out of 9312 23% |
| Number used as Logic | 2624 out of 17600 15% | 2385 out of 17600 13% | 2034 out of 17600 11% |
| Number used as Memory | 688 out of 6000 11% | 54 out of 6000 0% | 54 out of 6000 0% |

| TABLE: II comparison between three models of Spartan 6 Based FPGA |
|---------------------|---------------------|---------------------|
| Name | Phase-I | Phase-II | Phase-III |
|------|--------|--------|--------|
| Selected Device | 6slx16csg324-2 | 6slx16csg324-2 | 6slx16csg324-2 |
| Number of Slice Registers | 3555 out of 18224 19% | 1910 out of 18224 11% | 1433 out of 18224 7% |
| Number of Slice LUTs | 4247 out of 9112 46% | 2377 out of 9112 26% | 2088 out of 17600 11% |
| Number used as Logic | 2494 out of 9112 27% | 2323 out of 9112 25% | 2029 out of 9112 22% |
| Number used as Memory | 1753 out of 2176 80% | 54 out of 2176 2% | 54 out of 2176 2% |

| TABLE: III comparison between three models of Zybo Zynq 7000 series Based FPGA |
|---------------------|---------------------|---------------------|
| Name | Phase-I | Phase-II | Phase-III |
|------|--------|--------|--------|
| Selected Device | 7z010clg400-3 | 7z010clg400-3 | 7z010clg400-3 |
| Number of Slice Registers | 2321 out of 35200 6% | 1880 out of 35200 5% | 1403 out of 35200 3% |
| Number of Slice LUTs | 2388 out of 17600 13% | 2439 out of 17600 13% | 2088 out of 17600 11% |
| Number used as Logic | 1700 out of 17600 9% | 2385 out of 17600 13% | 2034 out of 17600 11% |
| Number used as Memory | 688 out of 6000 11% | 54 out of 6000 0% | 54 out of 6000 0% |

| TABLE: IV comparison between three phases of FPGAs |
|---------------------|---------------------|---------------------|
| FPGA Family | FFT Based | SGA Based | RSA Based |
|---------------|--------|--------|--------|
| Spartan 3E | 0.088 | 0.083 | 0.083 |
| Spartan 6 | 0.032 | 0.029 | 0.029 |
| Zynq 7000 Series | 0.101 | 0.101 | 0.101 |

From the above tables this paper concludes that Zynq 7000 series FPGA based DTMF detection gives the better optimized results interns of area and power.

7. Conclusions

Due to rapid improvements in Communication Technologies, The FPGA technology is increasing its applications in communication technologies. This research work implemented FPGA based efficient DTMF detection using Split Goertzel Algorithm with optimized Resource Sharing Approach.

Here In the first phase of the project work the FFT based DTMF detection using Xilinx FFT core is implemented.
The area, timing and power results are analyzed. In the second phase the split Goertzel algorithm is implemented and analysis is carried out. The area, timing and power results are analyzed. In the next phase the resource sharing approach is studied and suitable state machine based scheduling will be carried with limited resources to implement split Goertzel algorithm. It will be demonstrated that the novel resource sharing based approach consumes less power and can still efficiently detect the DTMF tones. To test the project at various stage DTMF tone generator module also will be implemented with digital carrier generators.

Modelsim Xilinx Edition (MXE) and Xilinx ISE is used for simulation and synthesis respectively. The Xilinx Chipscope tool is used to test the FPGA inside results while the logic running on FPGA. The Spartan 3E, Spartan 6 & Xilinx Zynq 7000 series Family FPGA development board is used in this project.

Acknowledgements

I would like to thank the higher authorities of JNTUA and Acharya Nagarjuna University to use ISRO Data Anaysis Centre and R & D Laboratories.

References

1. S Nagakishore Bhavanam, P. Siddaiah, P. Ramana Reddy “Zynq 7000 series Fpga based DTMF detection” IEEE International Conference on Computational Intelligence and Computing Research (ICCIC 2014), IEEE Xplore, IEEE Catalog Number: CFP1420J-ART, ISBN: 978-1-4799-3975-6, CD-ROM: IEEE Catalog Number: CFP1420J-CDR, 978-1-4799-3972-3, Print: IEEE Catalog Number: CFP1420J-PRT, 978-1-4799-3974-9.

2. S Nagakishore Bhavanam, P. Siddaiah, P. Ramana Reddy “FPGA based efficient DMTF detection using Split Goertzel Algorithm with optimized Resource Sharing Approach” 11th International Conference 11th IEEE and IFIP International Conference on wireless and Optical communications Networks (WOCN2014), IEEE Xplore, IEEE Catalog Number: CFP12604-CDR, ISBN: 978-1-4799-3155-2 and ISSN: 2151-7703.

3. Sonal Singhal , Piyush Kuchhal “Network on Chip for DTMF Decoder and TDM Switching in Telecommunication Network with HDL Environment” IEEE 2012, pp.1582-1588.

4. Kamal Shaterian, Hossein gharaee “DTMF detection with Goertzel Algorithm using FPGA, a resource sharing approach” IEEE 2010, pp.196-199.

5. Zhang Xinyi ” The FPGA Implementation of Modified Goertzel Algorithm for DTMF Signal Detection” IEEE 2010, pp.4811-4815.

6. S Nagakishore Bhavanam “Goertzel Algorithm Based DTMF Detection”, American American International Journal of Research in Science, Technology, Engineering & Mathematics (AJIRSTEM), Issue 6, Vol.1, March-May 2014, ISSN (Print): 2328-3491, ISSN (Online): 2328-3580, ISSN (CD-ROM): 2328-3629, pp 6-12.

7. S Nagakishore Bhavanam, P. Siddaiah, P. Ramana Reddy “FFT based DTMF detection by using Spartan 3E FPGA”. American International Journal of Research in Formal, Applied and Natural Sciences (AIRFANS), Issue 7, Volume 1, pp. 01-06, June-August, 2014, ISSN (Print): 2328-3777, ISSN (Online): 2328-3785, ISSN (CD-ROM): 2328-3793.

8. J. Nagi*, K. S. Yap, S. K. Tiong at al. “Intelligent Detection of DTMF Tones using a Hybrid Signal Processing Technique with Support Vector Machines”, IEEE 2008.

9. Thierry Capitaine, Valery Bouhary at al “Signal processing algorithm on a low resource processor: DTMF strings identification on a DIL8 package microcontroller” IEEE 2008, pp.1050-1054.

10. J. Nagi, S. K. Tiong, “Dual-tone Multifrequency Signal Detection using Support Vector Machines” IEEE 2008, pp.350-355.

11. Li-Te Shen and Shaw-Hwa Hwang” A New Algorithm For Dtmf Detection” IEEE 2009, pp.350-355.

12. Ge Jinzhao, Zhang Lulin, Qian Yumei, “A new signal detection method based on Goertzel algorithm”,Communication Technology, volume 9, pp16-18, 2002.

13. Felder D, Mason C, “Efficient Dual-Tone Multi-frequency Detection Using the Nonuniform Discrete Fourier Transform”,IEEE Signal Processing Letters, 5(7): pp160-163, 1998.

14. Mehran Yahyavi, Arian Ghajarehadeh at al. “An Improvement of MIPS Rate in Detection of DTMF Signals of 64 Subscribers Using GOERTZEL’s Algorithm” IEEE 2011, pp.132-136.

15. Thiagrajan Viswanathan, “Telecommunication switching system and Networks”. 35th Reprinting August 2011, PHI Publishing pp.183-215.

16. Texas Instruments. “Modified Goertzel algorithm for DTMF using the MS320C80™”, application report spra066. 1996.

17. Jaquenod A.G., Villagarcia H.A., De giusti M.R. “efficient tone detection solution using programmable logic devices™”. Argentina: UNLP.

18. Dulik T. “an FPGA implementation of Goertzel algorithm™”Springer- Verlag Berlin Heidelberg 1999.

19. Smith, G. L. “Dual-Tone Multi-frequency Receiver Using the WEDSP16 Digital Signal processor™”. AT&T application note.

20. Oppenheim, Alan v. discrete-time signal processing. 1998. 0-13- 754920-2, pp.633-634.