[1]G. Vimala
[2]Dr. F. Vincy Lloyd
[3]K. Prasad Babu

FPGA Based Implementation of DDFS for PLL

Journal of Electrical Systems

Abstract: **-** Frequency synthesizers play a crucial role in providing stable and precise frequency sources for various electronic systems, contributing to the reliability and performance of communication and signal processing applications. A frequency synthesizer is an electronic circuit or device that generates output signals with a specified frequency. It is commonly used in communication systems, radio transmitters and receivers, radar systems, and various other electronic applications where precise and stable frequency sources are required. In this work a frequency Synthesizer is designed and developed for the specifications of output Frequency Range of 20 MHz to 100 MHz, Frequency Accuracy up to 10 Hz, High Switching Speed of 20 µSec, Low Phase Noise of 110 dBc/Hz at 10KHz from carrier, Frequency Modulation with Selectable Deviation, Frequency Chirp with Selectable Step Size, TDM mode up to 4 Pre Selected Frequencies, Fixed Frequency, FM and Chirp. The Synthesizer is digitally controllable with FTW, With the simulations Frequency Range achieved is 0-400MHz against 20 – 100 MHz, Resolution achieved is 2 Hz against 10 Hz, Phase Noise performance is 120 dBc @ 10kHz, against 110 dBc, Switching Time of 2µS against 20µS, and 4 Modes of Operations achieved successfully. Xilinx fpga 2v250fg256 is used for implementation. Number of Slices used are 1264 out of 1536 with 82% , Number of Slice Flip Flops used are 536 out of 3072 with 17% ,Number of 4 input LUTs used are 2238 out of 3072 with 72%, Number of bonded IOBs used are 52 out of 172 with 30%, Number of GCLKs used are 3 out of 16 with 18%. Compared to other DDFS implementations, this work ensured implementation of 32-bit FTW with various modes, better utilization, low power consumption, flexible coding.

Keywords: FTW, Low power-Frequency synthesizer, TDM, FF, FM, Chirp

I. INTRODUCTION

All Frequency Synthesizers play a very important role in Radio Communications. Military Communication Attack Applications require highly agile, very low phase noise and high resolution Synthesizers, which must be digitally controllable through computer with a simple set of commands. For such applications, Direct Digital Frequency Synthesis (DDFS) is proved to be the most viable solution since it provides many significant advantages over PLL based designs of Frequency Synthesizers. Fast Settling Time, Sub-Hertz Frequency Resolution, Continuous-phase Switching Response, and Low Phase Noise are the features easily obtainable in DDFS systems. The switching rate specification of the Frequency Synthesizers used for Military Communication Applications is of the order of microseconds, and is not possible to realize using Computer based controllers using commercial / embedded operating systems. FPGA based embedded controller is necessary in order to provide such high switching speeds. Also the FPGA based controller is designed to provide a simple and highly effective set of control commands, encapsulating the internal implementation details of the Synthesizer as a subsystem.

I.I Specifications

The specifications of the frequency synthesizer are as follows:

1. Output Frequency Range : 20 MHz to 100 MHz
2. Frequency Accuracy : 10 Hz
3. Switching Speed : < =20 µSec
4. Low Phase Noise :< = -110 dBc/Hz at 10KHz from carrier
5. Frequency Modulation : Deviations: 8kHz, 15kHz, 30kHz, 100 kHz,
   Input for frequency modulation: 14bit data generated by an external ADC.
6. Frequency Chirp : Step Sizes: 12.5 kHz, 25 kHz, 50 kHz
7. TDM mode : up to 4 preselected frequencies
8. Digital Controllability : Frequency, FM deviation, TDM and Chirp

In order to develop a Frequency Synthesizer to meet the above specifications, the ASIC AD9858 of Analog Devices Corporation is used as the DDFS core, with the Xilinx FPGA XC2V250 as the embedded controller. The IC AD9858 is a direct digital synthesizer (DDFS) with a clock of 1 GSPS, and 32-bit frequency tuning word. The FPGA based controller is developed for programming the AD9858 via parallel (8-bit) format, by receiving the commands from a PC based external controller. Precise timings of the order of nanoseconds over the command lines of the DDFS are implemented using the FPGA during the Frequency Modulation, Frequency Chirp and Time Division Multiplexed modes of operation. In the Fixed Frequency mode of operation, the FPGA receives frequency command from the PC based external controller in BCD format, computes the 32 bit

[1]* Research Scholar D23EC503, ECE, BIHER Tambaram, Tamilnadu, gandlavimala30@gmail.com[1]
[2]Professor, ECE, BIHER, Tambaram, Tamilnadu, vincylloyd.ece@bharathuniv.ac.in[2]
[3] Associate Professor, ECE, Ashoka Womens Engineering College, Kurnool, A. P. kprasadbabuece433@gmail.com[3]

frequency tuning word and programs it into the DDFS. In the Frequency Modulated mode of operation, the FPGA receives the center frequency and deviation commands from the external controller. It computes the frequency tuning word and programs it into the DDFS. It triggers the external ADC at a rate of 500 kHz and reads the ADC output to recalculate the frequency tuning word and reprogram the DDFS instantaneously. In the Frequency Chirp mode of operation, the FPGA receives the Start Frequency, Stop Frequency and Chirp Step commands from the external controller. It computes the frequency tuning words and Ramp Rate word and programs it into the DDFS. It retriggers the DDFS at regular intervals to return the DDFS to the Start Frequency. In the Time Division Multiplexed mode of operation, the FPGA receives upto 4 frequencies from the external controller. It computes the frequency tuning words and programs them into the DDFS. At regular intervals the FPGA changes the profile selection of the DDFS in order to switch between the frequencies. Apart from meeting the stringent programming timing cycle requirements, the complex computational requirements of the frequency tuning words and ramp rate words of the DDFS are successfully achieved using this FPGA, with huge savings in computation time and memory usage.

I.II Development Cycle

The FPGA based controller is implemented in VHDL on Xilinx Integrated Synthesis Environment version 6.3, with Modelsim Simulation Environment version 5.7 as the simulation tool. Software is developed on Microsoft Visual C++ 6.0 environment for the preparation of arrays for use in the FPGA and also to evaluate the Synthesizer for its specifications. Figure 1 shows the process of development of the FPGA based controller, from conception through evaluation for the specifications.

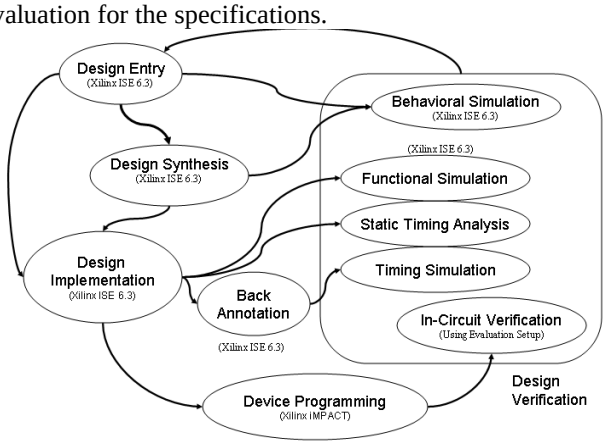


Fig 1 Process of development of the FPGA based controller.

The design process started with a set of drawings indicating the various functional blocks and state machines of the controller, with modular approach. Each of these modules is coded in VHDL on the Xilinx ISE 6.3 in its text editor. Simulations and corrections are carried out for every module using Modelsim SE 5.7 till the desired design performance is obtained. This process is repeated until the top level module is successfully simulated. Synthesis of the design is carried out using the Xilinx ISE 6.3 with the necessary constraints specified such as pin assignments. Using iMPACT tool of Xilinx, the resulting binary file is loaded into the config PROM of the FPGA through the JTAG port. The Synthesizer is then evaluated for its specifications though the controller software. The process is iterated till the desired specifications are achieved.

I.III Literature Survey

The following table represents the contributions of various authors on DDFS

Sl.No	Authors	Title	Year	Finding
1	Jyoti Sharma a b, Riyaz Ahmad	Design and optimization of phase frequency detector through Taguchi and ANOVA statistical techniques for fast settling low power frequency synthesizer	2024	In this work, a novel phase frequency detector (PFD) architecture using pass transistor logic is proposed. The circuit does not have a reset path, resulting in the elimination of blind zone and dead zone. The ϕ-V characteristics of the PFD were found to have better linearity across the range of − Π to Π due to the absence of blind and dead zones. The Taguchi and ANOVA statistical techniques were used to optimize the PFD. The optimized PFD exhibited a phase noise of −142.24 dBc/Hz, consumed 5.64 µW of power and had a maximum operating frequency of 5.25 GHz, and a delay of 10.65 ps. Using this PFD, a GHz-range synthesizer was designed, and its performance characteristics were obtained from circuit simulations using CADENCE Virtuoso. The synthesizer had a power consumption of 4.25 mW at a supply of 1.8 V, achieved a lock time of 2.95µs, and could generate frequencies ranging from 0.1 GHz to 4.75 GHz while occupying a chip area of 0.013 mm2.
2	Guangkun Guo , Chao Li	Analysis and Implementation of a Frequency Synthesizer Based on Dual Phase-Locked Loops in Cesium Atomic Clock	2023	A mathematical model of the frequency synthesis chain is established to estimate its performance. The phase-settling time and system stability are analyzed and studied in detail, and the obtained results are verified by experiments. An optimized realization of the frequency synthesizer shows that the phase-settling time can be adjusted within the range of 644.5 µs to 1.5 ms. Additionally, we measure the absolute phase noise values to be −63.7 dBc/Hz, −75.7 dBc/Hz, −107.1 dBc/Hz, and −122.5 dBc/Hz at 1 Hz, 10 Hz, 1 kHz, and 10 kHz offset frequencies, respectively
3	Arunima Sarkar, Asima Adak	Design and implementation of a frequency synthesizer using PLL	2022	An investigations has been made on the digital circuitry and the analog circuitry
4	Luca Bertulessi	Frequency Synthesizers Based on Fast-Locking Bang-Bang PLL for Cellular Applications	2021	The synthesizer has an output frequency from 3.59 GHz to 4.05 GHz. The integrated output jitter is 182fs and the power consumption of 5.28 mW from 1.2 V power supply leads to a FoM of −247.5 dB. This topology exploits a novel locking technique that guarantee a locking time of 5.6 s, for a frequency step of 364 MHz, despite the use of a single bit phase detector.
5	Rituparna Bhowmick, J. Manjula	Design of PLL based Frequency Synthesizer using Harmonic Extraction Techniques	2016	PLL is introduced which consists of proportional path and integral path. Furthermore Phase and Frequency Detector (PFD) is implemented using Gate Diffusion input (GDI) method. The proposed design is implemented using 90nm Cadence Virtuoso Analog Design Environment tool. Findings: As a result of using this technique, VCO achieves tuning range of 23.17% and phase noise of -85dBc/Hz @1MHz and -93dBc/Hz@10MHz with a small power dissipation of 50uW and Frequency Synthesizer achieves power dissipation of 4.8mW.

II. The Design Approach

Direct Digital Frequency Synthesizer (DDFS) provides many significant advantages over PLL approaches. Fast Settling Time, Sub-Hertz Frequency Resolution, Continuous-phase Switching Response, and Low Phase Noise are the features easily obtainable in DDFS systems. The schematic diagram of Direct Digital Frequency Synthesizer (DDFS) is shown in the figure 2

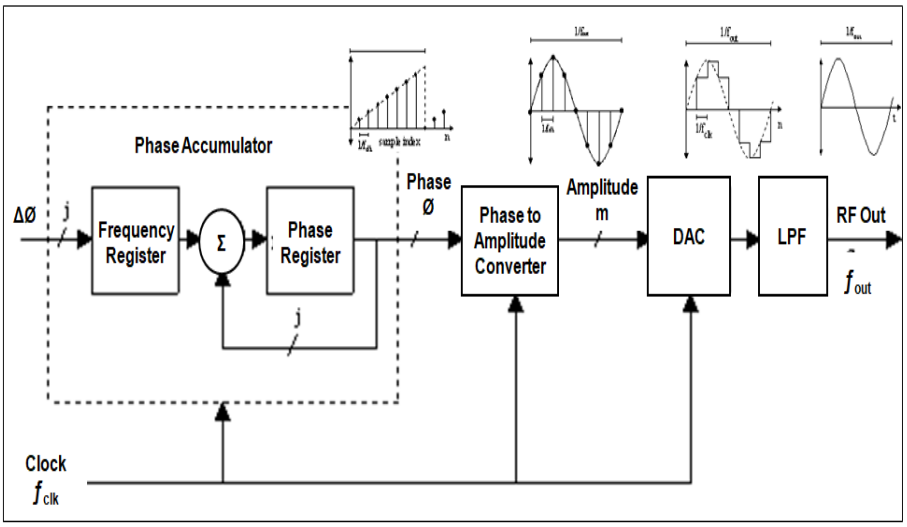


Fig 2. Direct Digital Frequency Synthesis

The DDFS has the following basic blocks: a Phase Accumulator, a Phase to Amplitude Converter (conventionally a sine ROM), a Digital to Analog Converter and a Filter. The Phase Accumulator consists of an N-bit Frequency Register which stores a Digital Phase Increment Word (alternately called Frequency Tuning Word (FTW)) followed by an N-bit full adder and a Phase Register. The Digital Phase Increment Word is entered in the Frequency Register. At each clock pulse this data is added to the data previously held in the phase register. The Digital Phase Increment Word represents a phase angle step that is added to the previous value at each $1/f$clk seconds to produce a linearly increasing digital value. The phase value Φ is generated using the modulo 2N overflowing property of an N-bit phase accumulator. The rate of the overflows is the output frequency

$$f\text{out} = \Delta\Phi \times f\text{clk} / 2N \;\forall\; f\text{out} \leq f\,\text{clk} / 2 \qquad (1)$$

where N is the number of phase accumulator bits, $f$clk is the clock frequency and $f$out is the output frequency. The constraint in above equation comes from the sampling theorem. The phase increment word in the equation is an integer, therefore the frequency resolution is

$$\Delta f\text{out} = f\text{clk}/2N \qquad (2)$$

The Read Only Memory (ROM) is a sine look-up table, which converts the digital phase information into the values of a sine wave. In the ideal case with no phase and amplitude quantization, the output sequence of the table is given by

$$\sin(2\pi\,\Phi(n) / 2N) \qquad (3)$$

where Φ(n) is the N-bit phase register value at the nth clock period. The numerical period of the phase accumulator output sequence is defined as the minimum value of Φe for which Φ(n) = Φ(n+Φe) for all n. The numerical period of the phase accumulator output sequence (in clock cycles) is

$$\Phi e = 2N / \text{GCD}(\Phi,2N) \qquad (4)$$

where GCD (Φ,2N) represents the greatest common divisor of Φ and 2N. The numerical period of the sequence samples recalled from the sine ROM will have the same value as the numerical period of the sequence generated by the phase accumulator. Therefore, the spectrum of the output waveform of the DDFS prior to a digital-to-analog conversion is characterized by a discrete spectrum consisting of Φe points. The ROM output is presented to the D/A-converter, which develops a quantized analog sine wave. The D/A-converter output spectrum contains frequencies n$f$clk ± $f$out, where n = 0, 1, …etc. The amplitudes of these components are weighted by a function

$$\text{sinc}(f/f\text{clk}) \qquad (5).$$

This effect can be corrected by an inverse sinc(f/$f$clk) filter. The filter that is after the D/A converter removes the high frequency sampling components and provides a pure sine wave output. As the DDFS generates frequencies close to $f$clk/2, the first image ($f$clk - $f$out) becomes more difficult to filter. This results in a narrower transition band for the filter. The complexity of the filter is determined by the width of the transition band. Therefore, in order to keep the filter simple, the DDFS operation is limited to less than 40 percent of the clock frequency. The Direct Digital Frequency Synthesis Technology keeping its relative merits in view is decided to be the most suitable option for the Frequency Synthesizer developed under this work. According to the specifications of this work, the range of frequencies that the Frequency Synthesizer is expected to generate is 20 – 100MHz at a step size of 10Hz. The CMOS Integrated Circuit AD9858 of Analog Devices is a Direct Digital Frequency Synthesizer having 32 bit Frequency Tuning Word operating on 1GHz clock is capable of meeting these specifications. Also the AD9858 is capable of generating Chirp over a band of frequencies at the specified

frequency steps. Having four profiles for frequency selection, the AD9858 is ideally suitable for the specification of upto four Time Division Multiplexed signals. AD9858 is controllable through a set of 20 command lines at a rate of 100MHz max. This meets the maximum data input rate required by the specifications for TDM and Chirp Signal generation. But since no Personal Computer with a Commercial Operating System can generate data at this high speed, a Field Programmable Gate Array (FPGA) capable of operating on 50MHz clock is required to control the AD9858.

XC2V250-5FG256I is chosen to control the DDFS. This FPGA is an Industrial Grade 250k gate FPGA of Xilinx Corporation that meets the requirements mentioned above. The specifications of this IC are featured in the reference along with the remaining components used in this work.

III. FUNCTIONAL SETUP

The image shown in figure3, represents the functional block diagram of Frequency Synthesizer.

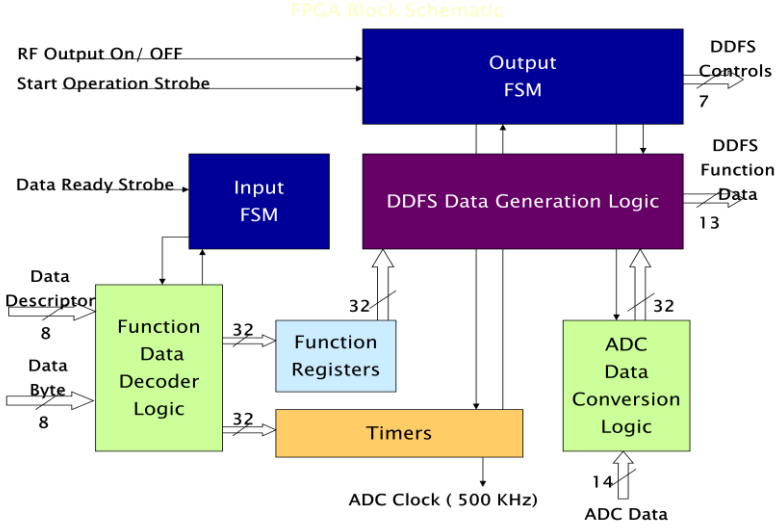


Fig 3. Functional block Diagram of Frequency Synthesizer.

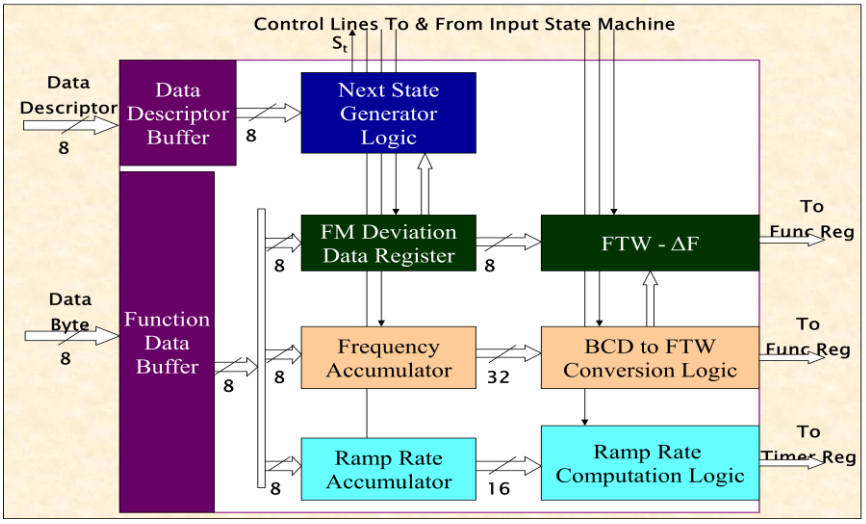


Fig 4. Data Decoder Logic

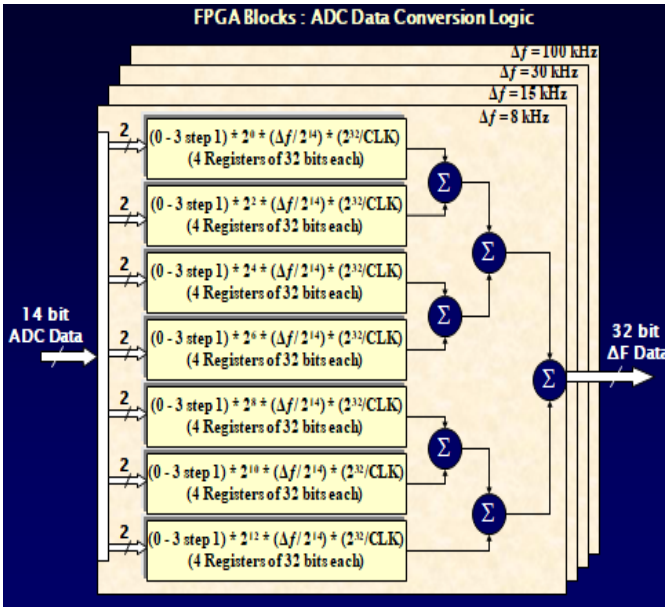


Fig 5. ADC data conversion logic

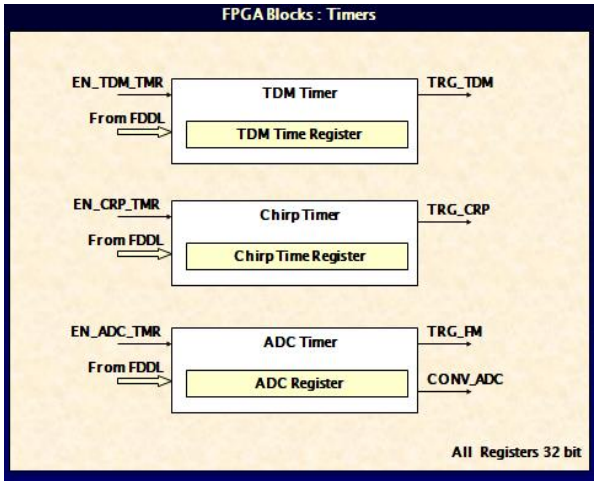


Fig 6. Functional Registers

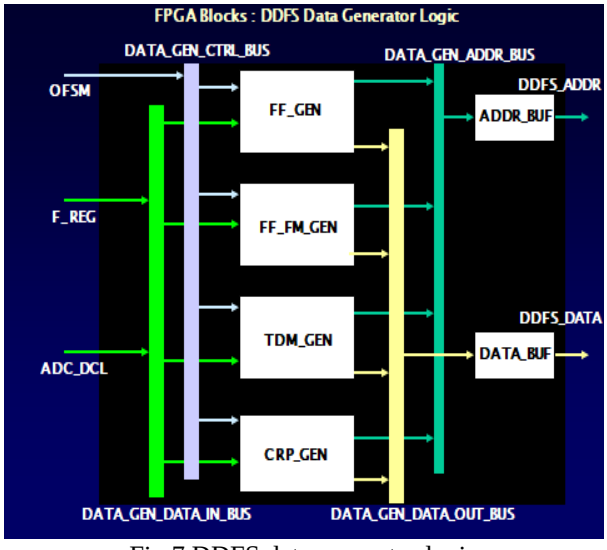


Fig 7 DDFS data generator logic

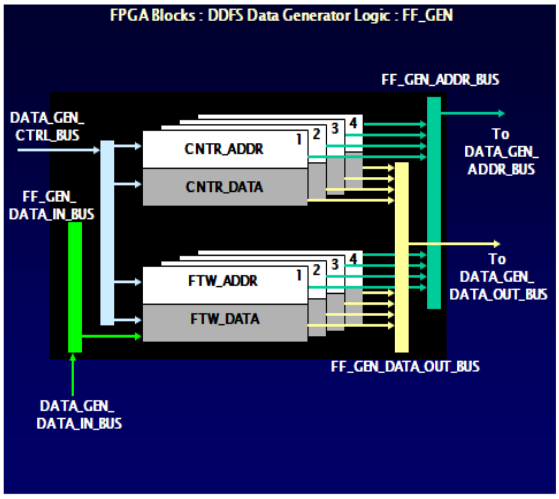
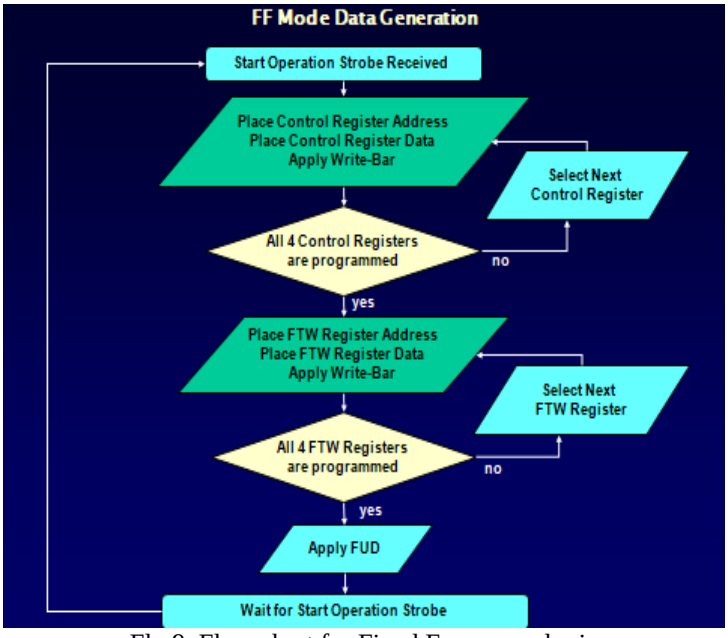


Fig 8 FF_GEN block



Flg 9. Flow chart for Fixed Frequency logic

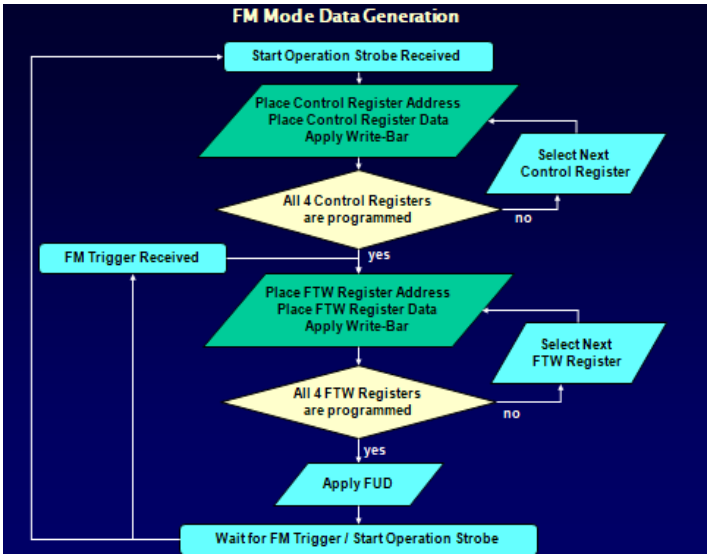


Fig 10. Flow chart for FM logic

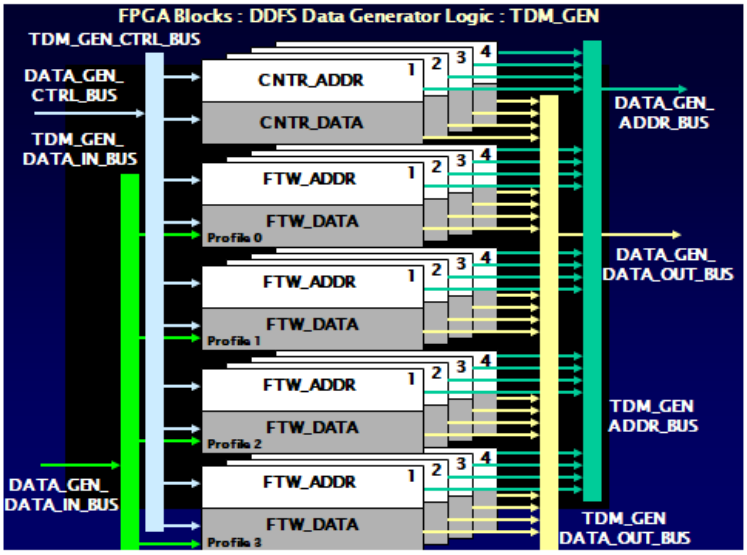


Fig 11. TDM Data Generator logic

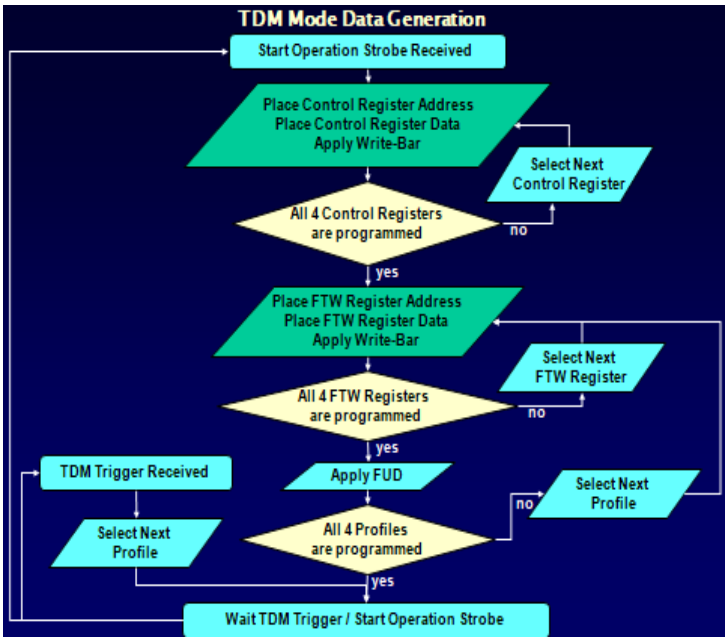


Fig 12. Flow Chart for TDM logic

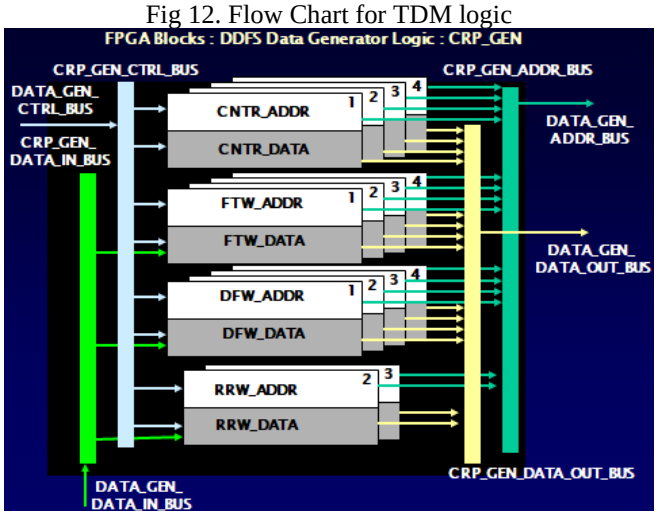


Fig 13 Chirp modulation generation logic

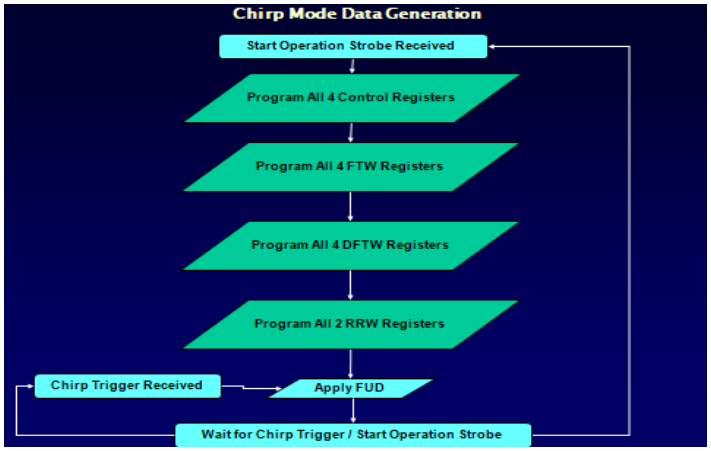


Fig 14 Flow chart for Chirp logic

IV. SIMULATION RESULTS

Before The results of evaluation as tabulated in table 1 indicate that the requirements of Frequency Accuracy, Resolution and Phase noise are successfully met. Switching time is tested by successively switching the output of the Synthesizer between two frequencies and the output is fed to the RF of a double balanced mixer. The LO of the mixer is fed from a Signal Generator generating one of the two frequencies. The output of the mixer transits between DC and RF, and the maximum value of this transit time is measured using the Storage Oscilloscope. Approximately 2µS of switching period is achieved against the specification of 20µS. Fig 4 presents the fixed frequency and phase noise performance of the synthesizer. The phase noise achieved is -120dBc/Hz at 10kHz against the specification of –110dBc/Hz. Figure 5 and 6show the frequency deviations of 15 and 100kHz of Frequency Modulated mode respectively. Fig 7 presents the TDM mode upto 4 frequencies and fig 8. presents the Chirp Mode of operation over a frequency band 45MHz to 65MHz.

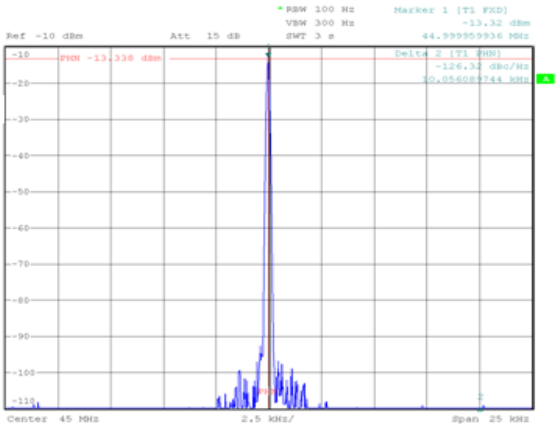
TABLE1

SNo	Frequency Set MHz	Frequency Measured MHz	Phase Noise dBc/Hz at 10KHz from the carrier	Spurious dBc at >5kHz from the carrier
1	20.00000	20.00000	<-120	<-120
2	25.11111	25.11111	<-120	<-120
3	42.12345	42.12345	<-120	<-120
4	52.88866	52.88866	<-120	<-120
5	61.00001	61.00001	<-120	<-120
6	76.33996	76.33996	<-120	<-120
7	82.00000	82.00000	<-120	<-120
8	19.000002	19.000002	<-120	<-120
9	16.000004	16.000004	<-120	<-120
10	100.000006	100.000006	<-120	<-120
11	124.001008	124.001008	<-120	<-120
12	320.50000	320.50000	<-120	<-120
13	0.000002	0.000002	Spectrum Analyzer Cannot Measure	Spectrum Analyzer Cannot Measure
14	400.00000	400.00000	<-120	<-120

Table 1 Results of Evaluation for Frequency Range, Accuracy and Phase Noise

IV.I FIXED FREQUENCY MODE:

Frequency Accuracy is measured using a Frequency Counter with a measurement accuracy of 1Hz. The Frequency Accuracy of the Synthesizer is measured upto 2Hz and is found to exceed the specified 10Hz. The Phase Noise of the Synthesizer is measured in the Spectrum Analyzer whose output is displayed in figure 4. From the Center frequency of 45MHz, the measured value of the Phase Noise at 10 kHz is -126dBc/Hz, which is much superior to the specified -110dBc/Hz. Spurious signal levels within the close vicinity of the Center Frequency are highly dependent on the DDFS Clock Source.



Fixed Frequency Accuracy and Phase Noise performance

Fig 15. Fixed Frequency and Phase Noise performance of the Synthesizer

IV.II FREQUENCY MODULATED MODE:

Figures 15 and 16 show the Frequency Synthesizer operating in Frequency Modulation mode with deviations of 15kHz and 100kHz respectively from the center frequency of 45MHz. The span of the Spectrum Analyzer is accordingly set for each of the frequency deviations.

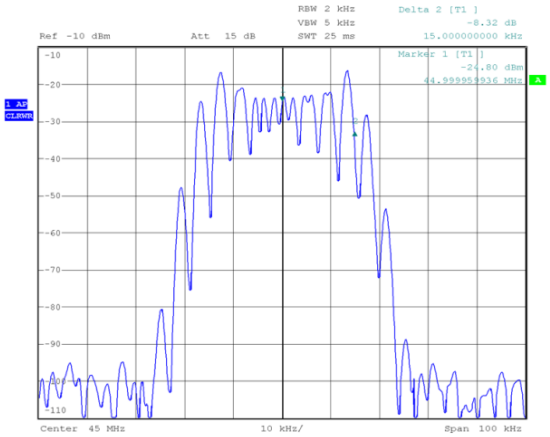


Fig 16. Frequency Modulation performance of the Synthesizer with 15kHz Deviation1kHz sine wave.

In figure 16 FM is applied as modulating signal to the external ADC. The ADC's CONVERT signal is applied at a rate of 500kHz by the FPGA. The digital output data of the ADC is read by the FPGA to compute the instantaneous frequency using the Split and Add Technique algorithm . The instantaneous FTW thus computed is loaded into the DDFS by the FPGA. During the evaluation, the Frequency Synthesizer is found to successfully operate in FM mode for all the modulating signal frequencies of up to 100kHz.

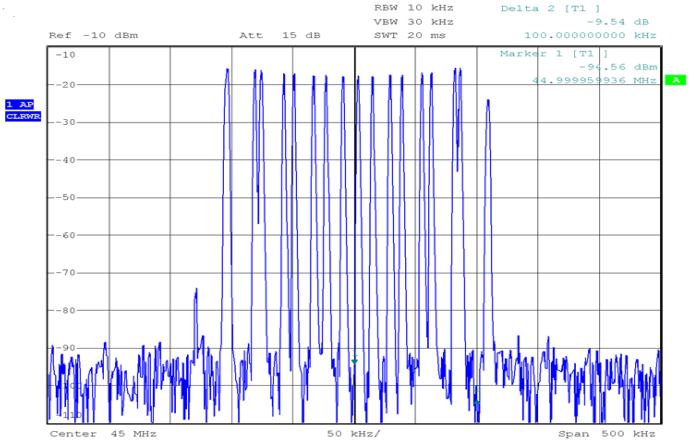


Fig 17. Frequency Modulation performance of the Synthesizer with 100kHz Deviation

IV.III TIME DIVISION MULTIPLEXED MODE OF OPERATION:

Figure 18. shows the TDM performance of the Synthesizer with 4 Frequencies switched at the minimum switching time. The Resolution Bandwidth and the Video Bandwidth of the Spectrum Analyzer are appropriately set in order to view clear pips of all the 4 frequencies. Due to the rapid switching, all the 4 frequencies of the Synthesizer appear to exist simultaneously as displayed in the spectrum. The spurious generated by this Synthesizer during TDM mode of operation is less than -50 dBc, which is far less than -30dBc generated by the state-of-the-art Synthesizers.

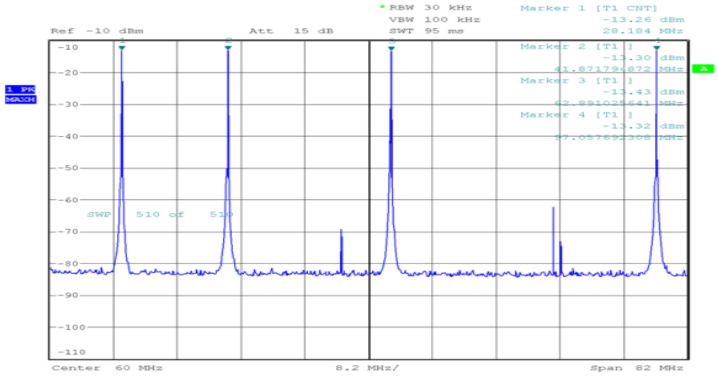


Fig 18. TDM performance of the Synthesizer with 4 Frequencies

IV.IV CHIRP MODE OF OPERATION:

The Chirp performance of the Synthesizer is shown in figure 8. between 45MHz and 65MHz. The output spectrum in the chirp mode of operation is very flat over the entire frequency range of operation. No spurious above -50dBc out of the Chirp Frequency Band is reported in the output of the Synthesizer.

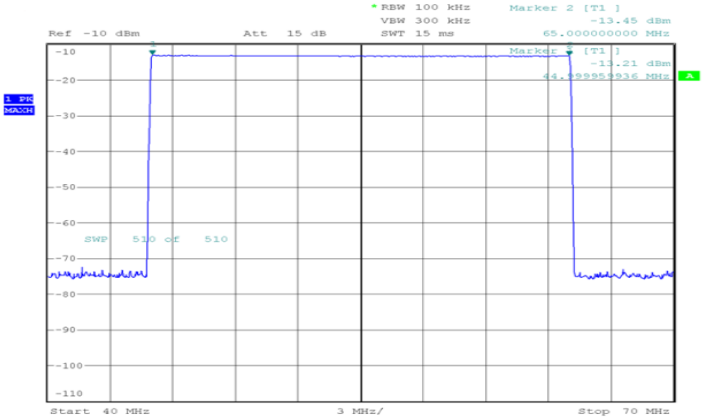


Fig 19. Chirp performance of the Synthesizer

The results achieved are thus beyond specifications, proving the selection of DDFS is appropriate for the Frequency Synthesizer developed under this work. An extract of the results of synthesis of VHDL code is presented as shown below.

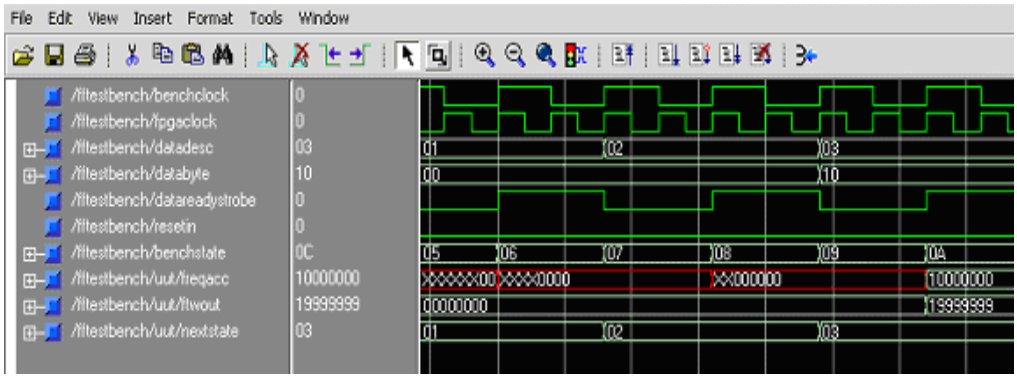


Fig 20.Simulation for FTW Generation

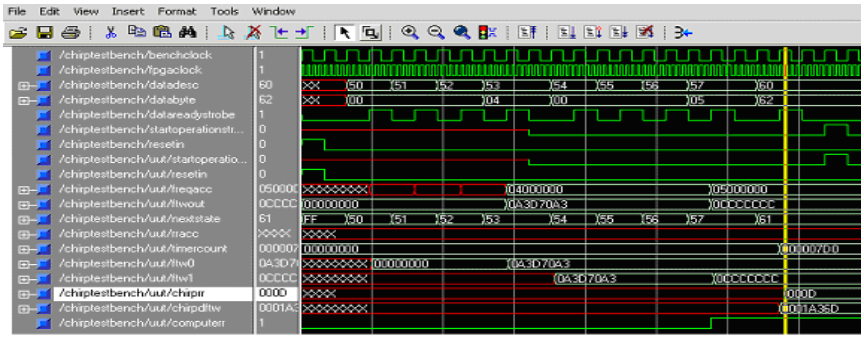


Fig 21 Simulation for Ramp Rate computation.

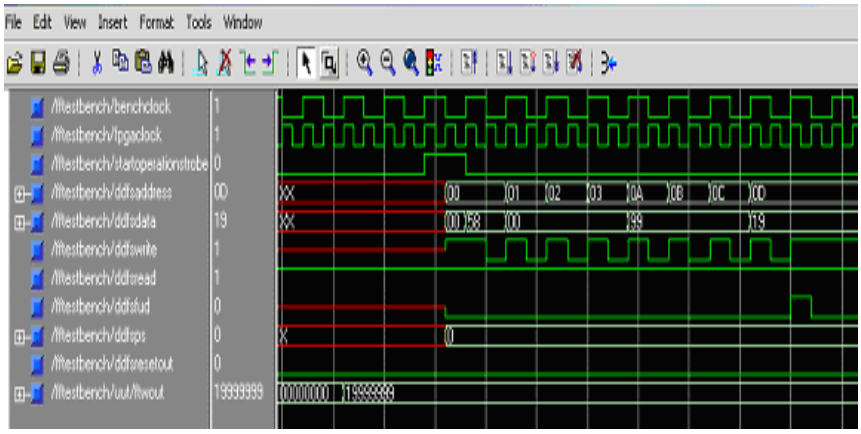


Fig 22. Simulation for DDFS Programming during FF Mode of operation

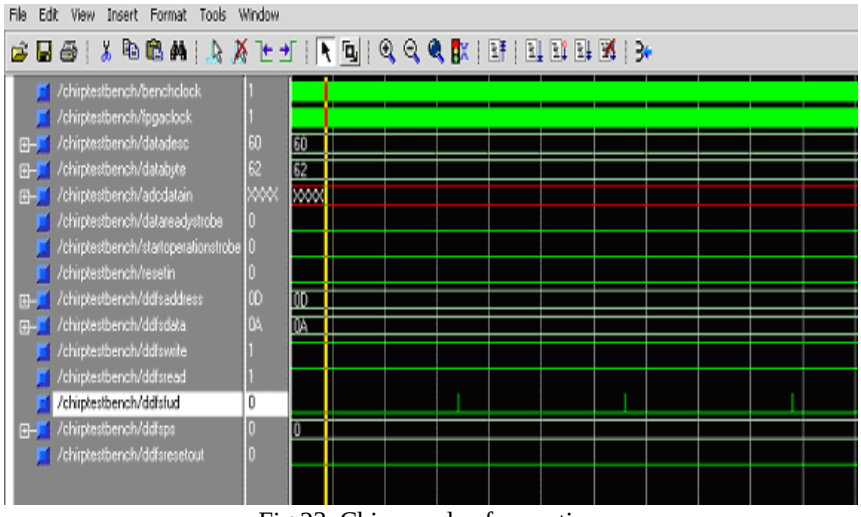


Fig 23. Chirp mode of operation

IV.V DEVICE UTILIZATION SUMMARY:

Selected Device : 2v250fg256-5
Number of Slices :1264 out of 1536 (82%)
Number of Slice Flip Flops: 536 out of 3072 (17%)
Number of 4 input Look Up Tables: 2238outof3072(72%)
Number of bonded IOBs : 52 out of 172 (30%)
Number of GCLKs : 3 out of 16 (18%)

4.6 TIMING SUMMARY:

Speed Grade : -5
Minimum period : 12.744ns
Maximum Frequency : 78.468MHz
Minimum input arrival time before clock : 30.907ns
Maximum output required time after clock: 6.127ns

Maximum combinational path delay: 5.064ns

From the device utilization summary, 82% of the 250k Gate FPGA is occupied by the code developed under this work. Maximum frequency of clock with which the FPGA can operate is 78.4 MHz as given in the timing summary, which lead to a decision of using a 50 MHz clock to the FPGA due to its immediate availability as well as computational simplicity while computing the timer counts for FM, TDM and Chirp modes of operation.

V. CONCLUSION

The requirements of fast switching time, high resolution and very low phase noise are successfully achieved through the FPGA based controller developed under this project. This indicates that the selection of Direct Digital Frequency Synthesis Topology is proper. Also, the range of frequencies that this DDFS can generate is 40% of its clock (1GHz) i.e. 400MHz, which is far beyond the specification. The output frequency range achieved is 0-400MHz against the specification of 20-100MHz. The switching speed (2μS) achieved is far beyond the specification (20μS). The resolution achieved is 2 Hz against the specification of 10 Hz. The phase noise achieved is -120dBc/Hz at 10kHz against the specification of –110dBc/Hz.The requirements of Data Conversion from BCD to Frequency Tuning Word (FTW) of the DDFS and its Command Rate of 50MHz during Chirp mode of operation are successfully achieved using the Field Programmable Gate Array (FPGA) thus establishing the correctness of the design approach.

REFERENCES

[1] Jyoti Sharma a b, Riyaz Ahmad ,Design and optimization of phase frequency detector throughTaguchi and ANOVA statistical techniques for fast settling low power frequency synthesizer, 2024

[2] Saurabh Kumar, Yatendra Kumar Singh, A low-jitter and low-phase noise switched-loop filter PLL using fast phase-error correction and dual-edge phase comparison technique 2024

[3] Guangkun Guo , Chao Li , Analysis and Implementation of a Frequency Synthesizer Based on Dual Phase-Locked Loops in Cesium Atomic Clock, 2023

[4] Chandra Keerthi Pothina ,Ngangbam Phalguni Singh ,Jagupilla Lakshmi Prasanna,Chella Santhosh and Mokkapati Ravi Kumar Design of Efficient Phase Locked Loop for Low Power Applications 2023

[5] Kewei Hu, Junwen Zhang,Kai Huang; Xiaoyun Kuang; Kaitian Huang; Xiaopeng YuA Low Power PLL Circuit with Area-Saving Loop Filter Applied in MEMS 2022

[6] Suneui Park, Seojin Choi Seyeon Yoo; Yoonseo Cho; Jaehyouk Choi, An Ultra-Low Jitter, Low-Power, 102-GHz PLL Using a Power-Gating Injection-Locked Frequency Multiplier-Based Phase Detector 2021

[7] Vaishali A V, Dr. Kiran Agarwal GuptaLow Power Phase Locked Loop Design Techniques 2020

[8] Aniruddha Kailukea , Pankaj Agrawalb , R. V. Kshirsagara, Design of Low Power, Low Jitter PLL for WiMAX Application in CMOS process 2019

[9] Nabihah Ahmad, Nur Atikah Binti Ishaimi2 and M. Hairol Jabbar, Charge Pump and Loop Filter for Low Power PLL Using 130nm CMOS Technology 2018

[10] Tun-Ju Wang, Tzu-Hsuan Liu; Tsung-Hsien Lin, An ultra-low power 169-nA 32.768-kHz fractional-N PLL 2017

[11] Xuan Guo, Danyu Wu; Lei Zhou; Huasen Liu; Jin Wu; Xinyu Liu A 4-GHz 32-bit Direct Digital Frequency Synthesizer in 0.25μm SiGe HBT with SFDR > 46 dBc up to Nyquist bandwidth2016

[12] Supraja Batchua Jayachandra Prasad Talarib Ravi Nirlakallac, Analysis of Low Power and High Speed Phase Frequency Detectors for Phase Locked Loop Design 2015.