

Performance Realization of CORDIC based GMSK System with FPGA Prototyping

Renuka Kajur¹

Assistant Professor, Department of Electronics and Communication Engineering, PESIT, Bengaluru, India

K V Prasad²

Professor and Head, Department of Electronics and Communication Engineering, BIT, Bengaluru, India

Abstract—The Gaussian Minimum Shift Keying (GMSK) modulation is a digital modulation scheme using frequency shift keying with no phase discontinuities, and it provides higher spectral efficiency in radio communication systems. In this article, the cost-effective hardware architecture of the GMSK system is designed using pipelined CORDIC and optimized CORDIC models. The GMSK systems mainly consist of the NRZ encoder, Integrator, Gaussian filter followed by FM Modulator using CORDIC models and Digital Frequency Synthesizer (DFS) for IQ Modulation in transmitter section along with channel, the receiver section has FM demodulator, followed by Differentiator and NRZ decoder. The CORDIC algorithms play a crucial role in GMSK systems for IQ generation and improve the system performance on a single chip. Both the pipelined CORDIC and optimized CORDIC models are designed for 6-stages. The optimized CORDIC model is designed using quadrature mapping method along with pipeline structure. The GMSK systems are implemented on Artix-7 FPGA with FPGA prototyping. The Performance analysis is represented in terms of hardware constraints like area, time and power. These results show that the optimized CORDIC based GMSK system is a better option than the pipelined CORDIC based GMSK systems for real-time scenarios.

Keywords—GMSK; CORDIC algorithm; FPGA; DFS; Gaussian Filter; pipelined; integrator; differentiator; channel

I. INTRODUCTION

In the current era of high-speed communication, the prime objective of the system is to achieve modulation that has a power spectrum with a constant amplitude and adequate bandwidth. Out of all, some of the efficient techniques are Minimum Shift Keying (MSK) and Gaussian Minimum Shift Keying (GMSK). They are derived from Constant Phase-FSK family of modulation operating at a constant envelope. As a modulated signal has the characteristic of constant amplitude, the power consumption can be minimized by using a Class C RF amplifier. Employing an amplifier is necessary for battery operated units. Before modulation, the MSK signal along the I-Q component allows via half-sine shaper to pass the modulation signal. GMSK incorporates the same technique as that of MSK. The only variation is, instead of the shape of the half-sine pulse the inputs bits form the shape of the Gaussian bell curve. The realization of such shapers is carried using different digital or analog circuits of PCM LUTs [1-7]. The present wireless digital communication systems use different types of modulation techniques to achieve high performance with efficient usage of available spectrum efficiency. The MSK provides the constant envelope signals for power amplification

which reduces the problems caused by non-linear distortion, and MSK encodes each bit is like a half sinusoidal. The spectrum is not close enough to aware of the data rate approaches in RF channel Bandwidth. The GMSK overcomes the problems of MSK. GMSK is used in many wireless applications, and it limits the spectral bandwidth by using Gaussian filtering [1-2].

The GMSK is widely used in mobile communications and associated modulation scheme for GSM standards, and most of the GSM Mobiles have long battery life with greater efficiency and because of RF-Power amplifiers. The software-defined radio is a simplified approach which is adopted in most of the cellular standards. The American /European countries use GMSK modulation schemes in cellular phones as per GSM standards. The GMSK supports non-linear modulation, which is used for easy implementation on hardware [3-5]. If the Radio signal frequency increases in mobile communication and it is difficult to use by conventional technology like DSP, Microcontroller and Microprocessor in the real-time conditions due to its computational complexities. The GMSK system based transmitter and receiver is designed with timing synchronization using the Mueller and Muller method. The timing synchronization is necessary for real-time usage for GMSK systems, GMSK receiver consumes larger chip area and affects the overall system performance [6]. The GMSK modulation for EDGE mobile communication is established with a new digital technique of linear approximation based on Inter-symbol-interference (ISI) and partial response signaling. The GSM/Edge systems support narrow bandwidth, reasonable power spectrum usage, lesser the impulse response overshoot, higher immunity to noise interface and constant filter output pulse with a modulation index of 0.5[7]. The GMSK system is mainly used in Automatic identification systems (AIS) [8], OFDM communication systems [9] and digital image processing [10] and other applications.

Section II discusses the existing approaches of the GMSK system and its application usage and also the review of the new CORDIC algorithms and problem identification with research gaps. Section III describes the CORDIC algorithm principles along with pipelined and optimized CORDIC hardware architecture. The GMSK system is elaborated in section IV with GMSK principles, GMSK modulation, and Demodulation architectures. The results and performance analysis of Pipelined CORDIC and optimized CORDIC based GMSK systems are represented in section V. section VI concludes the overall work of GMSK systems with constraints improvements.

II. EXISTING WORKS

In this section, the existing approaches towards GMSK systems and its application usage and also a few existing CORDIC based approaches are elaborated. Munir et al. [11] present Cube Sat AIS receiver module using GMSK Modulation, the received AIS signals from the transmitter are at low power, and it is challenging to analyze operating frequency. So by deploying the GMSK modulation, the signals are analyzed in appropriate conditions. Poletaev et al. [12] present phase distortion variation estimation of GMSK modulated signals under Very low frequency (VHF). The phase distortion detector is designed using VCO's for IQ modulation and arc tan module for IQ generations for the generation of phase drift of VLF signals. Rakesh et al. [13] present the software-based approaches for BER analysis of GMSK system under AWGN channel condition. The Simulink tool is used for GMSK systems which include an encoder, GMSK Modulator followed by AWGN channel and GMSK Demodulator along with Viterbi decoder. The GMSK signals are generated for mobile radio telephony using Simulink modeling by George et al. [14]. The performance analysis of the GMSK Model for different fading channel conditions along with hard decision and soft decision decoding. The soft decision decoding techniques give better BER than hard decision decoding. Ghnimi et al. [15] present GMSK modulation under radio mobile propagation conditions based on the Matlab environment and its performance realization including AWGN along with one path and four path channel fading are analyzed.

The FPGA Based GMSK Modulator is designed by Nitin et al. [16] for GSM system which includes differential encoder-decoder, ROM based sine and cosine wave (IQ) generation using phase trajectory, phase concatenation followed by phase accumulation module. Gupta et al. [17] present GMSK transceiver on the FPGA platform for VHF waveform generation. The transmitter design includes Gaussian filter followed by FM Modulator with Direct digital synthesizer (DDS) up converter using XILINX system generator tool. Similarly, for GMSK receiver, DDS followed by Cascaded-integrator-comb (CIC) decimator filter and non-coherent Viterbi decoder. The GMSK carrier phase recovery loop for non-linear analysis are evaluated by jhaidri et al. [18], and the system includes precoder, GMSK Modulator followed by AWGN channel, along with coherent demodulation and carrier phase recovery on Matlab simulation environment. The FPGA based GMSK Demodulator using CORDIC engine is designed by Kumar et al. [19], and demodulator includes IQ generation using 1024 x16 ROM Module followed by 6-stage CORDIC module along with differentiator and decision synchronizer module. The design consumes a huge amount of chip area and power.

The floating-point Arithmetic processor based on CORDIC algorithm is designed by bingyi et al. [20] which includes mantissa bits generation by pre-processing unit, followed by reconfigurable CORDIC rotation unit and exponent bits for normalization using post-processing model. The CORDIC rotation unit is a unified structure for multiplication, division, and square roots. The Hoang et al. [21] presents 32-bit Floating point FFT twiddle factor calculation unit using adaptive

CORDIC module. The CORDIC module generates the sine and cosine values are used for FFT twiddle factor calculation for each iteration. Madi et al. [22] present a hardware implementation of sine and cosine generation using CORDIC module. The phase inputs are generated from the ROM table followed by CORDIC5.0 is adopted form Xilinx IP Core is modeled using Xilinx system generator. The Mousetrapped-radix-2 CORDIC module is designed by chagela et al. [23], which includes asynchronous pipelined architecture to improve the throughput and power-delay product.

Research Gap: It has been noticed from the review of existing works carried on GMSK systems are based on software and few on hardware-based approaches. The recent existing approaches towards GMSK systems are not optimized yet and facing computational complexity, performance degradation, more resource utilization on hardware. The existing GMSK Modulation uses Voltage controlled oscillator (VCO), or local oscillators for IQ modulation or ROM based LUT, or current waveform generation approaches for IQ generation, which affects the overall chip area and system performance on GMSK systems. Few approaches are used CORDIC based designs for IQ generation but are facing problems on angle rotation convergence, more iteration stages to achieve the sine and cosine values, not pipelined and LUT based memory for tangent values updating. These research gaps are overcome by using the Proposed CORDIC based GMSK system in next sections.

III. CORDIC DESIGNS

The CORDIC algorithm is an algorithm to calculate the trigonometric and hyperbolic operations, and it is also known as volder's algorithm. CORDIC typically intersect with one digit per iteration. In this section CORDIC algorithm principle and hardware architecture of pipelined and optimized CORDIC is explained in detail.

A. CORDIC Algorithm Principles

The system receives a vector and rotates by an angle θ for each iteration is

$$\begin{aligned}x_A &= x_i \cos\theta - y_i \sin\theta \\y_A &= x_i \sin\theta + y_i \cos\theta\end{aligned}\quad (1)$$

If we consider, $x_i = 1$ and $y_i = 0$, after the first rotation, it will be

$$\begin{aligned}x_A &= \cos\theta \\y_A &= \sin\theta\end{aligned}\quad (2)$$

To calculate the sine and cosine angles through rotation mode. Simplify the equation (1) to

$$\begin{bmatrix} x_A \\ y_A \end{bmatrix} = \cos\theta \begin{bmatrix} 1 & -\tan\theta \\ \tan\theta & 1 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix}\quad (3)$$

One rotation is performed using the equation (3), and it requires four multiplications along with addition and subtractions. To avoid the multiplications, use small arbitrary

angles by $\tan \theta_i = 2^{-i}$ for $i = 0, 1, \dots, n$. Simplify the equation (3) to i^{th} iterations is

$$\begin{aligned} x_{i+1} &= x_i - S_i 2^{-i} y_i \\ y_{i+1} &= y_i + S_i 2^{-i} x_i \\ z_{i+1} &= z_i - S_i \tan^{-1}(2^{-i}) \end{aligned} \quad (4)$$

The angle of the final iteration values accumulated by equation (4) and compared with an initial value z_0 . The sign S_i belongs to +1 for counterclockwise direction and -1 for clockwise direction.

B. Pipelined CORDIC Model

The CORDIC model is an iterative structure. So it will take a longer time to process an algorithm for the number of iterations. So use the pipelined registers between the iteration stages to achieve the fast computations. The sine and cosine angles have been computed in a row using a pipelined structure. The hardware architecture of the pipelined CORDIC model is represented in figure 1. For the rotation mode, set the initial values to $x_0 = 1$ and $y_0=0$ and $z_0= \theta$. The pipelined structure has a 6-stage ($i=6$) iterative process. The 8-bit CORDIC model has pipelined registers, shifters, adder/subtractor along with constant tangent values with counter mode. The accuracy of the computation will be increased based on the number of iterative stages. These initial values are shifted by i -bits, where i is an integer and shifted up to 0 to 5. The division of X_i and Y_i by 2^i is taken place by the shifter for each stage. The new vector values are generated at intermediate stages for the given vector, and it is iteratively rotated to get the desired angle Z_i . The Z_i will decide the sign for addition and subtraction. The selection of the sign S_i set by

$$S_i = \begin{cases} -1 & \text{if } Z_i < 0 \\ 1 & \text{if } Z_i > 0 \end{cases} \quad (5)$$

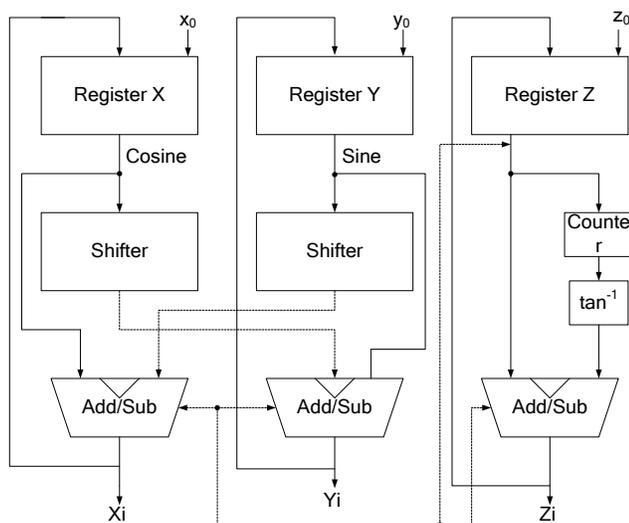


Fig. 1. Hardware Architecture of the Pipelined CORDIC Model.

If the initial stage is set to $x_0 = 1$ and $y_0=0$, then results are in the form of discrete sine and cosine values. It is difficult to realize on FPGA. So varies the X_i and Y_i values to realize on hardware and to avoid the fractional values. In order to improve the accuracy, multiply the constant $K= 0.611$ with 2^6 , so the initial values will set to $X_i = 38 + x_0$ and $Y_i=0+ y_0$ for six iterations. The addition/ subtraction outputs are feedback to pipelined registers to get each stage CORDIC outputs of sine and cosine values.

The tangent values are generated based on the following equation.

$$ZT_i = 2^6 \tan^{-1}(2^i) \quad (6)$$

The tangent values (ZT_i) are assigned for successive iteration using the counter to generate the Z_i values. The counter will count till 5 and reset to 0.

C. Optimized CORDIC Model

The improvised version of the pipelined CORDIC model is designed in this section and calls it as optimized CORDIC model. The significant difference between the above pipelined and this optimized CORDIC is quadrant mapping for proper angle rotation using preprocessing and post-processing, and separate pipelined structure for sine and cosine calculation and also delay module is used to synchronize the quadrant mapping with pipeline structure. The optimized CORDIC model is designed and represented in figure 2. The pipelined structure has a part of optimized CORDIC model is represented in figure 3.

The optimized CORDIC is overcome the rotation angle range problem, computational complexity, and avoiding the ROM table usage for tangent calculation, which saves the chip area of the CORDIC model. This optimized CORDIC is one of the best solutions for sine and cosine signal (IQ) generation to improve the speed and accuracy for GMSK systems and its applications.

The optimized CORDIC model has three stages namely, preprocessing, pipelined structure, and post-processing. The preprocessing stage receives the 8-bit phase input and for quadrant mapping, chooses the MSB 2-bit phase input [7:6] and transformed to the first quadrant. The same 2-bit is passed to the delay unit. The MSB of phase 2-bits [7:6] is "00", then assign the 8-bit phase input (PI) into phase register as a first quadrant. Similarly, if the bits are "01", assign PI-90°, for "10", assign PI-180° and for "11", assign PI-270° into phase register as a first quadrant. This 8-bit phase register is input to the pipelined CORDIC structure.

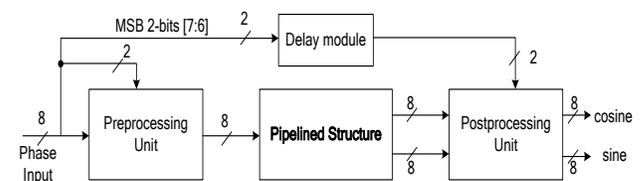


Fig. 2. Optimized CORDIC Model.

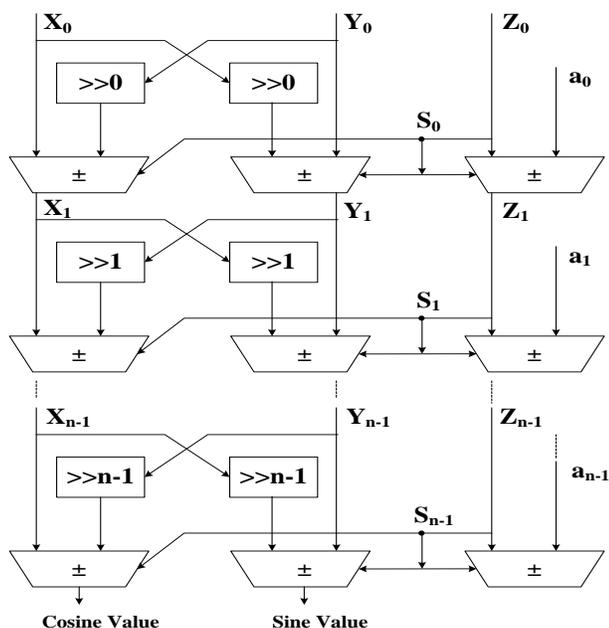


Fig. 3. Pipelined Structure used in Optimized CORDIC Model..

The pipelined structure is proportional to the accuracy of the angles and provides the high-speed calculation. In pipelined structure, Initialize the $X_0= 38$ (Decimal value), $Y_0 = 0$ and $Z_0 = 8$ -bit phase register value as angle input before the iteration starts. The pipelined structure has six stages, and set $n=6$ is shown in figure 3. The phase register input Z_i [7] decides the sign(S) for the corresponding iterations stages. The constant tangent values (a_0 to a_5) are assigned directly while performing the iterations. The adder/subtract or of each level completes the corresponding iteration stages. The pipelined registers are placed (not shown in figure 3) after each addition/subtraction operation except the final iteration stage. In six stage, the sine and cosine values are generated.

The delay register receives the MSB 2-bit phase input and generates the delayed 2-bit quadrant bits results parallelly with pipelined structure outputs to maintain the synchronization problems. The post-processing unit converts the first quadrant of the phase input to next quadrants along with pipelined outputs. The post-processing quadrant 2 bit generates the final cosine and a sine value from 6th stage pipelined outputs is as shown in table 1.

TABLE. I. QUADRANT MAPPING

| Quadrant | Angle range | Sine value | Cosine value |
|----------|--------------------------|------------------|------------------|
| 00 | $(0, 90^\circ)$ | $\sin\theta$ | $\cos\theta$ |
| 01 | $(90^\circ, 180^\circ)$ | $\cos\theta$ | $\sim\sin\theta$ |
| 10 | $(180^\circ, 270^\circ)$ | $\sim\sin\theta$ | $\sim\cos\theta$ |
| 11 | $(270^\circ, 360^\circ)$ | $\sim\cos\theta$ | $\sin\theta$ |

IV. PROPOSED GMSK SYSTEM

In this section, the proposed GMSK system methodology is addressed using the pipelined CORDIC and optimized CORDIC model and its design flow is represented in figure 4. The proposed work main aims to design cost-effective hardware architecture of the GMSK system using pipelined CORDIC and optimized CORDIC models on a single chip. The CORDIC models are incorporated for IQ generation, which speeds up the GMSK systems in real time scenarios. The GMSK Systems mainly consists of Non-return to- Zero (NRZ) encoder-decoder, Integrator- Differentiator, Gaussian filter, FM Modulator, and FM Demodulator along with Channel. The GMSK design has 1-bit input data feed serially one after other, and process the GMSK operation, generates the 1-bit output, which is similar to GMSK input data.

A. Algorithm Principle

The mathematical representation of the GMSK modulated signal is explained in this section.

1) Consider the binary data stream $a(n), a \in (0,1)$ and convert to an antipodal sequence $b(n), b \in (0,1)$ and a stream of rectangular pulses $r(t)$ is

$$r(t) = \sum_n b(n) p(t - nTb) \tag{7}$$

Where pulse $p(t) = 1, t \in (0, Tb)$ $Tb = 1/ fb$ and is $p(t) = 0, otherwise$ symbol interval.

2) The integrator is single-pole infinite impulse response with unity feedback coefficient, and in general, it is represented as

$$i(n) = i(n - 1) + r(t) \tag{8}$$

Where $n = 0, 1, 2,$ etc., many delay elements used in the integrator.

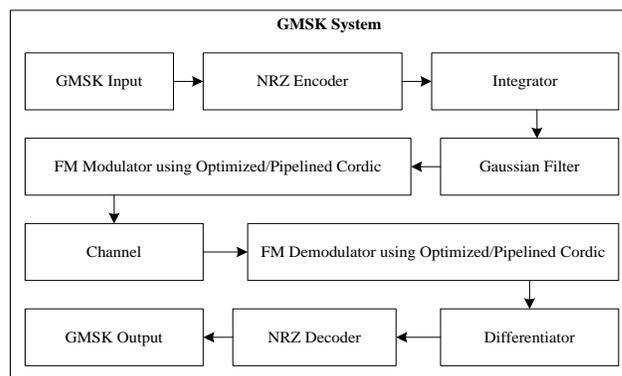


Fig. 4. Methodology of the Proposed Work.

3) The Gaussian filtering is one of the filtering technique which minimizes the group delay, and Gaussian function is used as an impulse response of the Gaussian filter. The Gaussian filter is designed using the FIR method and is expressed as

$$g(t) = \sum_{t=0}^{M-1} h(t).i(t) \quad (9)$$

Where $i(t)$ is filtered pulse, $M=8$ for tap filter and Gaussian filter is specified by BT product, and impulse response $h(t)$ is

$$h(t) = \frac{1}{\sqrt{2\pi}.\delta T} e^{-\frac{t^2}{2\delta^2 T^2}} \quad (10)$$

Where standard deviation $\delta = \frac{\sqrt{\ln 2}}{2\pi BT}$ and $BT = 0.3$ for GSM

4) The modulation of the carrier wave frequency by $g(t)$ around a center frequency f_c implies modulating its phase by integrating the filtered output. so

$$\phi(t) = W_c t + 2\pi f_m \int_0^t g(t).dt \quad (11)$$

Where the phase has been normalized to '0' at $t=0$. The instantaneous frequency of the modulating signal will be

$$f(t) = \frac{1}{2} \frac{d\phi}{dt} = f_c + f_m g(t) \quad (12)$$

Where f_c is carrier frequency and f_m is peak frequency deviation.

5) The peak frequency deviation is measured by the bit rate, and the GSMK need a one-bit interval of the period T_b contains 'N' cycles of frequency $f_c - f_m$ and $N + \frac{1}{2}$ cycles of $f_c + f_m$. This leads to $f_m = \frac{1}{4T_b} = \frac{f_b}{4}$.

6) The modulated carrier $y(t)$ is generated using the DFS method, and it can produce

$$y(t) = \cos(\phi(t)) \quad (13)$$

7) For maintaining the adequate sampling rates with digital methods requires at higher operating frequencies, So use Quadrature implementation approach to generate the modulated signal. The GSMK modulated signal $y(t)$ is

$$y(t) = \cos(W_c t) \cos(2\pi f_m \int g[T].dT) - \sin(W_c t) \sin(2\pi f_m \int g[T].dT) \quad (14)$$

Where $W_c = 2\pi f_c$ and Simplified version of the modulated signal $y(t)$ is

$$y(t) = I(t) \cos(2W_c T) - Q(t) \sin(W_c T) \quad (15)$$

B. Hardware Implementation

The GSMK Modulator mainly consists of NRZ Encoder, Integrator Gaussian filter followed by FM Modulation using CORDIC model's for IQ generation and DFS for IQ modulation and the hardware architecture of the GSMK Modulator is represented in figure 5. The GSMK modulator process 1-bit GSMK input serially and generates 12-bit GSMK modulated output.

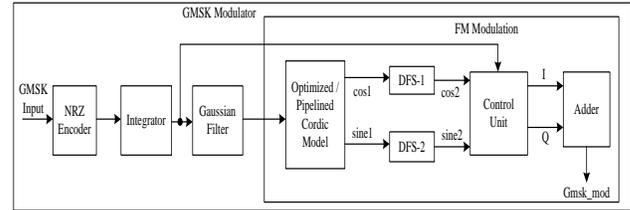


Fig. 5. Hardware Architecture of GSMK Modulator.

The NRZ Encoder receives the 1-bit GSMK input and generates the 1-bit output according to the NRZ logic. The NRZ encoder mainly contains the XOR module and data-Flip-flop. The XOR Module receives the GSMK input along with D-FF feedback output to process the transactions. The NRZ encoder is mainly used for slow speed transmission which interfaces for the synchronous and asynchronous process. The NRZ is to Logic '1', means the bit is set as a high and Logic '0', means the bit is set as a low value. The NRZ encoded output will be inverted and is input to Integrator, and It is mainly used for high data rate conversion with flexible multiplier less filter. The Integrator is used in the design is single-pole infinite impulse response (IIR) with unity feedback value. The integrator uses four data registers, and the NRZ encoded data is input to the first register. The first and fourth register outputs are added to generate the 1-bit integrated output.

The Gaussian filter is used to minimize the group delay in the GSMK process, and Gaussian function is the impulse response of the Gaussian filter. To design a Gaussian filter, the Low power FIR filter is used. The FIR design is an 8-tap filter, which is having 8- Gaussian filter coefficients, is multiplier with integrated output and generates the 8 temporary output values. These output values are stored in the eight registers parallelly. Using adder, add the corresponding register outputs individually, and 7th adder output is Gaussian filter output.

The FM Modulation mainly consists of pipelined CORDIC or optimized CORDIC model for IQ generation, two -Digital frequency synthesizer (DFS) modules for IQ Modulation, Control unit to generates I, Q separately, and an adder unit.

The pipelined CORDIC or optimized CORDIC models are explained in section III. The pipelined CORDIC or optimized CORDIC models receive the phase (angle) input and generate the In-phase (cos1) and Quadrature (sine1) phase waveforms. The DFS is used to convert the phase value to the sine wave. The DFS Module mainly contains phase accumulator which generates the slope value, followed by complementor which generates the triangle wave values, The Multiplexor-tree generates the off-wave (one-side) values, and finally, format converter is used to generate the sine waveform. The DFS

outputs (cos2 and sine2) are input to the control unit. Based on the integrator output values, the IQ output values are generated. The adder unit adds both the IQ values to generate the final GMSK modulated output.

The Channel is used to generate the noise using random sequences. The random sequences are generated using the LFSR module. For rapid implementation in Hardware, Galois field LFSR is used. The 5th order generator polynomial is used to design the Galois LFSR. The polynomial is $G(x) = x^5 + x^2 + 1$. The Galois LFSR hardware design uses the 5 adders, 6 multipliers, and 6-data flip-flops. Overall two LFSR modules of the same polynomials are considered and add the two LFSR outputs to frame the random sequence. These random sequences are the same as standard AWGN generation in real time consideration. The GMSK Modulated output is XOR with channel output to produce the corrupted output values which are input to GMSK Demodulator.

The GMSK Demodulator is used to recover the similar GMSK inputs, and it mainly consists of FM demodulator using pipelined or optimized CORDIC models for IQ generation and DFS for IQ modulation, along with Differentiator and NRZ decoder. The hardware architecture of the GMSK Demodulator is represented in figure 6.

The FM demodulator receives the corrupted data (demod_in) as an input to the control unit. The pipelined or optimized CORDIC modules generate the cosine (cos1) waveform based on the phase input values. The CORDIC output is input to the DFS process, which generates the arbitrary waveform as cos2. The Delay unit is used to synchronize the DFS Modulated output with demodulated input (demod_in). The control unit is used compare the delay DFS output and demodulated input which generates the 1-bit FM demodulated output. The differentiator process the FM demodulated output using four registers subtract or. The fourth register output is subtracted from the first register output to generate the differentiator output, and it is input to the NRZ decoder. The NRZ decoder decodes the differentiator output using the delay unit and the XOR Module. The differentiator output and delayed differentiator output are inputs to the XOR module. The XOR output is inverted and generates the 1-bit GMSK demodulated output which is almost similar to the original GMSK input data sequence.

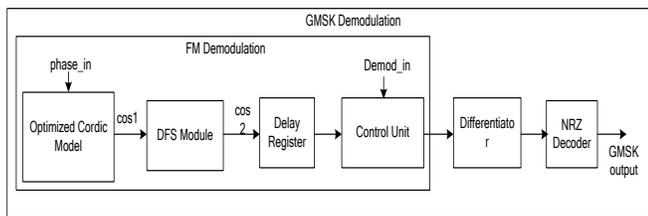


Fig. 6. Hardware Architecture of GMSK Demodulator.

V. RESULTS AND ANALYSIS

The pipelined CORDIC and Optimized CORDIC based GMSK system results are analyzed in the below section. The two different designs are modeled over Xilinx 14.7 platform using Verilog-HDL and simulated on Modelsim6.5f and finally prototyped on Artix-7 FPGA.

The resource utilization comparison of both the pipelined CORDIC and optimized CORDIC based GMSK system are tabulated in table 2, and the graphical representation is in figure 7. The comparative results are analyzed in terms of area, time and power. The optimized CORDIC based GMSK system utilizes only 145 slice registers, working at 259.477 MHz, and utilizes 0.09W total power consumption on Artix-7 FPGA.

The optimized CORDIC based GMSK system improves the area overhead around 41.53% in slice Registers, 44.75% in slice LUTs, and 38.13% in LUT-FF pairs than pipelined CORDIC based GMSK system. The optimized CORDIC based GMSK system improves timing overhead around 38.8% in operating frequency and reduction of 2.17% in total power utilization than pipelined CORDIC based GMSK system.

The summary of the hardware synthesis results shows that the optimized CORDIC based GMSK system is better than the pipelined CORDIC based GMSK systems. The optimized CORDIC model uses Quadrature mapping along with pipeline structure, and constant tangent values are assigned directly while processing in the iteration stages. But in the Pipelined CORDIC model, the constant tangent values are assigned based on the counter method.

The hardware resource comparison of both the pipelined CORDIC and optimized CORDIC models is tabulated in table 3, and the graphical representation is in figure 8. The pipelined CORDIC Model utilizes 146 slice registers, 331 Slice LUTs and 124 LUT-FF pairs on FPGA which is quite higher than the optimized CORDIC Model. The Pipelined CORDIC operated at 157.588 MHz maximum frequency with the minimum period of 6.346ns on Artix-7 FPGA.

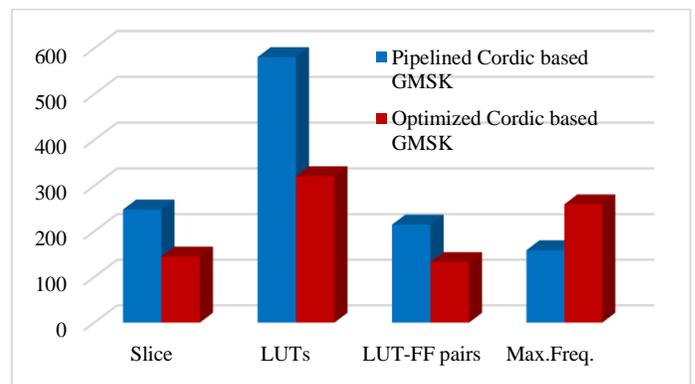


Fig. 7. Resource utilization Comparison.

TABLE. II. RESOURCE UTILIZATION OF BOTH PIPELINED AND OPTIMIZED CORDIC BASED GMSK SYSTEM

| Resource Utilization | Pipelined CORDIC based GMSK System | Optimized CORDIC based GMSK system |
|----------------------|------------------------------------|------------------------------------|
| Area | | |
| Slice Registers | 248 | 145 |
| Slice LUTs | 581 | 321 |
| LUT-FF pairs | 215 | 133 |
| Time | | |
| Minimum Period (ns) | 6.294 | 3.854 |
| Max.Frequency (MHZ) | 158.887 | 259.477 |
| Power | | |
| Dynamic power (W) | 0.01 | 0.008 |
| Total power (W) | 0.092 | 0.09 |

TABLE. III. RESOURCE UTILIZATION OF BOTH PIPELINED AND OPTIMIZED CORDIC MODELS

| Resource Utilization | Pipelined CORDIC Model | Optimized CORDIC Model |
|-----------------------|------------------------|------------------------|
| Area | | |
| Slice Registers | 146 | 98 |
| Slice LUTs | 331 | 114 |
| LUT-FF pairs | 124 | 81 |
| Time | | |
| Minimum Period (ns) | 6.346 | 1.962 |
| Max.Frequency (MHZ) | 157.588 | 509.71 |
| Speed | | |
| Latency(Clock cycles) | 8 | 7 |
| Throughput(Mbps) | 157.588 | 582.52 |

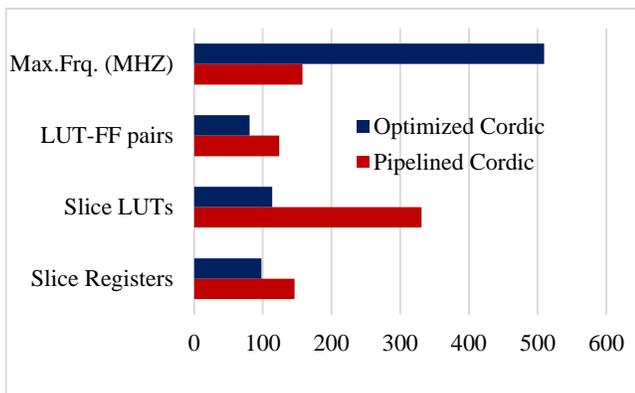


Fig. 8. Resource Comparison of Both Pipelined and Optimized CORDIC Models.

The optimized CORDIC model improves the area overhead around 32.87% in slice Registers, 65.55% in slice LUTs, and 34.67% in LUT-FF pairs than pipelined CORDIC based GMSK system. The optimized CORDIC based GMSK system improves timing overhead around 69 % in operating frequency than the pipelined CORDIC based GMSK system.

The speed comparison optimized CORDIC based GMSK system in terms of latency and throughput is improved around 12.5% and 72.89% respectively than pipelined CORDIC based GMSK system.

VI. CONCLUSION AND FUTURE WORK

The proposed GMSK system is designed using pipelined CORDIC and optimized CORDIC individually. The CORDIC Model is used to generate the IQ waveforms with low latency, which improves the speed of the GMSK system in real time scenarios. The pipelined CORDIC models is designed using shift and add method with pipeline registers along with tangent values used by the counter method. The optimized CORDIC model is designed using quadrant mapping and pipeline structure. Both the CORDIC models are designed for six stages. The hardware architecture of the GMSK system is implemented on Artix-7 FPGA with prototyping. The performance analysis of two CORDIC models based GMSK system is synthesized and summarize the hardware constraints like area, time and power. The optimized CORDIC based GMSK system improves the area (slices) around 41.53%, 38.8% in operating frequency and 2.17% in power utilization than pipelined CORDIC based GMSK system. Similarly, the optimized CORDIC model utilizes fewer resources in terms of the area (slices) around 32.87%, operating frequency around 69%, 12.5% in latency and 72.89% in Throughput than pipelined CORDIC model. In the future, adopt the proposed GMSK system in GSM standard for real-time usage and also for further spectral efficiency enhancements.

REFERENCES

- [1] Bax, Walt T., and Miles A. Copeland. "A GMSK modulator using a/spl Delta//spl Sigma/frequency discriminator-based synthesizer." *IEEE Journal of Solid-State Circuits* 36.8 (2001): 1218-1227.
- [2] Gudovskiy, Denis A., Lichung Chu, and Shinhaeng Lee. "A novel nondata-aided synchronization algorithm for MSK-type-modulated signals." *IEEE Communications Letters* 19, no. 9 (2015): 1552-1555.
- [3] Shah, Avni, and Sunil B. Somani. "An FPGA based generalized parametrizable modulator." In *2012 12th International Conference on Hybrid Intelligent Systems (HIS)*, pp. 402-407. IEEE, 2012.
- [4] Sen, Du, Yang Yuhong, Cui Penghui, and Guo Bin. "Research on the separability of GMSK mixed signals based on modulation parameters." In *2015 IEEE 5th International Conference on Electronics Information and Emergency Communication*, pp. 175-177. IEEE, 2015.
- [5] Xiaopeng, Gao, and Dong Deping. "Accelerate GMSK demodulation using GPU." *System Science, Engineering Design and Manufacturing Informatization (ICSEM)*, 2011 International Conference on. Vol. 2. IEEE, 2011.
- [6] Lee, Juan Felipe Medina, Juan Felipe Patarroyo Montenegro, Catalina Muñoz Morales, Alexander López Parrado, and Juan José Giraldo Gutiérrez. "Implementation of a gmsk communication system on fpga." In *2011 IEEE Second Latin American Symposium on Circuits and Systems (LASCAS)*, pp. 1-4. IEEE, 2011.
- [7] Liu, Shugang, and Hongli Liu. "A linear approximation of GMSK modulation for GSM/EDGE mobile communications." In *2008 4th International Conference on Wireless Communications, Networking and Mobile Computing*, pp. 1-4. IEEE, 2008.
- [8] M. A. Adarsh, K. Adithya and A. Nimbai, "FPGA Implementation of Space-Based AIS," *2018 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*, Chennai, 2018, pp. 1-5.
- [9] S. P. Majumder and M. A. Jumana, "Performance analysis of a space-frequency block coded OFDM wireless communication system with MSK and GMSK modulation," *2014 International Conference on Electrical Engineering and Information & Communication Technology*, Dhaka, 2014, pp. 1-5.
- [10] S. R. Udawant and S. S. Magar, "Digital image processing by using GMSK," *2016 International Conference on Automatic Control and Dynamic Optimization Techniques (ICACDOT)*, Pune, 2016, pp. 695-698.
- [11] Munir, Achmad, Nazmi Febrian, and Antrisha Daneraici Setiawan. "Signal analysis of GMSK modulation-based CubeSat automatic identification system receiver." In *2016 3rd International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE)*, pp. 439-442. IEEE, 2016.
- [12] Poletacv, Alexander S., Alexander G. Chensky, and Dmitrii A. Tokmachev. "A Method for Central Frequency Phase Variations Measurements of GMSK Modulated Signals." In *2018 XIV International Scientific-Technical Conference on Actual Problems of Electronic Instrument Engineering (APEIE)*, pp. 101-104. IEEE, 2018.
- [13] Rakesh, N. "Performance Analysis of BER for GMSK Signal for GSM Frequency Transmitted Through AWGN Channel." In *International Conference on Intelligent Data Communication Technologies and Internet of Things*, pp. 619-626. Springer, Cham, 2018.
- [14] George, Gejo, and Srija Unnikrishnan. "Performance Evaluation of GMSK for Digital Mobile Radio Telephony." *International Conference on Advances in Computing, Communication, and Control*. Springer, Berlin, Heidelberg, 2013.
- [15] Ghnimi, Said, AdnenRajhi, and Ali Gharsallah. "BER performance of GMSK modulation under radio mobile propagation environments." *Mediterranean Microwave Symposium (MMS)*, 2011 11th. IEEE, 2011.
- [16] Babu, KM Nitin, and K. K. Vinaymurthi. "GMSK modulator for GSM system, an economical implementation on FPGA." *Communications and Signal Processing (ICCS)*, 2011 International Conference on. IEEE, 2011.
- [17] Gupta, Subhashini, Vikas Bhatia, and L. C. Mangal. "An efficient FPGA implementation of GMSK (BT= 0.3) transceiver with non-coherent sequence detection for tactical V/UHF waveforms." In *2012 National Conference on Communications (NCC)*, pp. 1-5. IEEE, 2012.
- [18] Jhaidri, Mohamed Amine, Christophe Laot, and Alain Thomas. "Nonlinear analysis of GMSK carrier phase recovery loop." In *2016 International Symposium on Signal, Image, Video, and Communications (ISIVC)*, pp. 230-235. IEEE, 2016.
- [19] L. Kumar, D. K. Mittal, and R. Shrestha, "VLSI-design and FPGA-implementation of GMSK-demodulator architecture using CORDIC engine for low-power application," *2016 IEEE Annual India Conference (INDICON)*, Bangalore, 2016, pp. 1-6.
- [20] Li, Bingyi, Linlin Fang, Yizhuang Xie, He Chen, and Liang Chen. "A unified reconfigurable floating-point arithmetic architecture based on CORDIC algorithm." In *2017 International Conference on Field Programmable Technology (ICFPT)*, pp. 301-302. IEEE, 2017.
- [21] Hoang, Trong-Thuc, Duc-Hung Le, and Cong-Kha Pham. "VLSI Design of Floating-Point Twiddle Factor Using Adaptive CORDIC on Various Iteration Limitations." In *2018 IEEE 12th International Symposium on Embedded Multicore/Many-core Systems-on-Chip (MCSoc)*, pp. 225-232. IEEE, 2018.
- [22] Madi, A. Ait, and A. Addaim. "Optimized Method for Sine and Cosine Hardware Implementation Generator, using CORDIC Algorithm." *International Journal of Applied Engineering Research* 13, no. 1 (2018): 21-29.
- [23] Changela, Ankur, Mazad Zaveri, and Anurag Lakhani. "FPGA implementation of asynchronous mousetrap pipelined radix-2 cordic algorithm." *2018 International Conference on Current Trends towards Converging Technologies (ICCTCT)*. IEEE, 2018.