A Novel Design of Hilbert Huang Based All Digital Phase Locked Loop Using FPGA

Velamarthi Spandana* and Chandra Sekhar Paidimarry
Department of Electronics and Communication, Osmania University, Hyderabad, Telangana-500007, India
Email: velamarthispandana@gmail.com* (V.S.); chandrasekharpaidimarry@yahoo.com (C.S.P.)

***Abstract*—Many applications for digital processing are steadily increasing in radio frequency signal processing from analog to digital. ADPLL plays a vital role in digital signal processing. Several ADPLL models were introduced in the past. However, in those models, the power dissipation gets increased due to the delay in the lock state. Due to the generated digital noise, the system leads the output signal with inherent phase noise and does not undergo the frequency division process. Therefore, a novel Hilbert-Huang based All Digital Phase Locked Loop (HH-ADPLL) was proposed to make the ADPLL into the locked state. The input reference signal is initially entered in the Hilbert-Huang transform to extract the analytic component and generate the up and down signals. By eliminating the higher frequency part from these signals, the up/down counter produces borrow and carry signals. The carry and borrow signals are fed into the increment decrement counter to produce the output signal. At last, input and output signal matching is carried out in the phase detector module, and ADPLL enters a locked state. The presented HH-ADPLL is implemented in Artix-7 FPGA, and the efficiency is validated in terms of stability, phase error, power dissipation, combinational delay, and performance improvement computed.**

***Index Terms*—All digital phase locked loop, digital controlled oscillator, field programmable gate array, Hilbert-Huang transforms, phase detector**

I. Introduction

Recently, Phase-Locked Loops (PLLs) have seen an increase in the application of communication systems for phase synchronization, modulation, and demodulation due to the advancement of wireless communication technologies [1]. The core component for controlling the loop of the connected devices with flexible utility is PLL [2]. The PLL performs the generation or recovery of the clock frequency. Applications for processing Radio-Frequency (RF) signals are increasingly transitioning from analog to digital as digital signal processing components get better and better [3]. So, to replace the traditional analog PLL, the new All Digital Phase Locked Loop is introduced (ADPLL) [4]. In PLL, there are four types. Here ADPLL was considered for implementation [5]. Generally, ADPLL consists of three modules: Loop Filter (LP), Digital Controlled Oscillator (DCO), and Phase Detector (PD). It has increased stability, easy to implement, and provides better jitter performance than the conventional PLL. Its working is the same as the conventional PLL, but the designed component is in digital format [6]. It also has advantages in power consumption and size of the design [7]. Due to the use of a digital LF and their improved adaptation to micrometer mechanisms, ADPLLs have stronger immunity against noise, a small surface, and better programmability [8]. However, sometimes it suffers from a tradeoff between frequency range and resolution. To overcome these issues, an ADPLL with lock stages is introduced to minimize the error and to gain high filtered output [9].

ADPLL can be classified into two types: single phase and three phases. Identifying the error in a single phase is complex, while the transformation function obtains the errors in three phases [10]. An ADPLL is fully realized on Field Programmable Gate Array (FPGA) for synchronizing a digital carrier [11]. An FPGA is a prototyping-based design that evaluates the design's performance and simulation for the system designer's work. An ADPLL was implemented on FPGA independently or inside the network to validate previously obtained results with the minimal cost of building an Integrated Circuit (IC) [12]. The ADC and DAC are also available in the FPGA. To simulate ADPLL, it used the z-transform of FPGA [13]. FPGA is created under the achievement of resolution of 15-ps Root Mean Square (RMS), less than 10-ps Integral Nonlinearity (INL) and 11-ps average bin size, and less than 10-ps Integral Nonlinearity (INL) [14]. It is the most common method used for communication technology and to compute the digital signal [15]. While there are certain drawbacks to prototyping on an FPGA, the most significant drawback is that the mixed-signal circuits are essential for an ADPLL operation which cannot be implemented on the FPGA. Also, in such a situation, complications occur in the jitter requirements; therefore, frequency stability must be attained for the higher clock frequency [16, 17].

The essential components of digital circuits are present. Still, the elements are not in accurate transistor level but are implemented by the lookup tables and additional multi-role FPGA structure. Sometimes the digital circuits produce a considerable rate of noise. As a result, the circuit's output signal has intrinsic phase noise and causes a vast phase uncertainty. Thus, phase noise becomes a major worry for a combination of two signals and could

Manuscript received December 26, 2022; revised February 17, 2023; accepted April 6, 2023.
*Corresponding author: Velamarthi Spandan

impair phase-alignment performance [18−20]. The power dissipation is high due to the long processing time of the ADPLL to attain the locked state. Considering these issues, the proposed model was created.

The further description of the presented model was aligned as follows: Section II details the related works. Section III describes the existing methodology. Section IV gives an elaborate explanation of the proposed methodology. Section V gives the details about experimental studies. Section VI concluded with the conclusion and future work, respectively.

II. Related Works

To achieve the signal in a locked state, a new ADPLL system was discussed by Radhapuram *et al*. [21]. This technique comprises four modules: Phase Frequency Detection (PFD), controller, ring- Digital Controlled Oscillator (DCO), and frequency divider. Initially, the PFD could detect the differences in frequency and phase among the two entered signals. Then, the controller performed the charge pump function, and the pulses at the ring DCOs frequency divided LF identified output. Finally, a program was generated for tuning the frequency of the ring DCO output, the same as the reference signal, and producing the precise phase difference detection for achieving the signal in a locked state. However, it could increase the time and complexity of the design.

Bharat and Kumar [22] presented architecture for generating true random numbers based on ADPLL to minimize the two signals' phase variation. This approach consists of three modules: PD, LF, and DCO. Initially, XOR-Gate was represented as the PD and interlocked among the input, output, and frequency phases. In this architecture, the difference in the signals is reduced by PD. Then, to eliminate noise or undesirable frequency using the loop filter K-counter. Next, DCO would change the frequency based on a signal generated at LF. Finally, the ID counter and K counter clock were matched to find whether it was similar; if the difference is high, it should be minimized. This technique requires low throughput and a low degree of randomness for generating the output signal.

The jitter behavior in a generalized ADPLL, an event-driven self-sampled model, was studied by Koskin *et al*. [23], and the discrete-time model simulation provided a steady-state analysis. Here, the latter optimized the jitter under the restriction of two control parameters. The measure of the ADPLL-produced jitter is described by averaging time error. Finally, the performance analysis of this model is carried out by the FPGA measurements. This model helps to find some starting estimates of the optimization parameters quickly. However, for the small programmable, the approximation is less accurate.

Koskin *et al.* [24] performed an event-driven ADPLL based on FPGA with an asynchronous control. They compared simulated results with the theoretical model through parameters such as phase plane representation, transient response, and order parameters. A digitally controlled oscillator and frequency detector are considered a local clock for modeling the ADPLL. In consideration of this local clock, the system produces the local signal. The model's position at the rising edges of the reference or local waveform is updated. The model has proved that FPGA maintains an honest relationship with manual predictions. However, if the ratio of programmable parameters of FPGA increases, the ADPLL model goes out of control.

The existing techniques to achieve faster locking range use some overloaded modules. So Dinesh and Marimuthu [25] introduced R2POS-DPLL, an Optimal Sampling Digital PLL combining high Precision Time to Digital Controller (TDC) and a wide range of high-Resolution DCO. This combination locking range has been enhanced, and the frequency resolution was maximized. Moreover, high power consumption is reduced by the modified bang band detector, which is used as a phase detector. However, during the fractional precision, unacceptable noise shape deterioration occurs.

The key contributions of this presented work are,

- Initially, HH-ADPLL was designed with three modules: PD, LF, and DCO, and they are implemented in Xilinx.
- Input signals are applied to the phase detector. The phase detector computes the phase difference between two signals.
- The up/down counter generates the carry and borrows signal and is given to the increment decrement counter.
- The increment decrement counter generated the signal I_{out}, which depends on the output of the up/down counter.
- The increment-decrement counter output signals match each other and put the ADPLL into a locked state.
- Subsequently, the performance parameters were measured and compared with other approaches.

III. System Model and Problem Statement

ADPLL contains digital blocks that use the control loop with negative feedback and accepts only the digital signal. This signal may be a single digital signal in collaboration with the parallel digital signal. This design provided three blocks such as PD, LF, and DCO. The designed ADPLL mainly aimed to interlock the frequency and phase of the input and output. Initially, the PD was used to diminish the difference between the two signals. The loop filter is the block that was used to eliminate the noise in the signal. Finally, the output signal matching the input signal is created by DCO. Also, for modifying the output signal frequency, it is divided by the N counter in the way of feedback. Here the obtained output signal is less in phase error and meets the locking range condition.

The main problem in designing the ADPLL was increasing the design time and complexity, which was too complicated for the design process. Also, it provides low throughput, and the degree of randomness was low to generate the output signal and provide delay for developing the output signal in the ADPLL design. The design of the system model is given in Fig. 1.

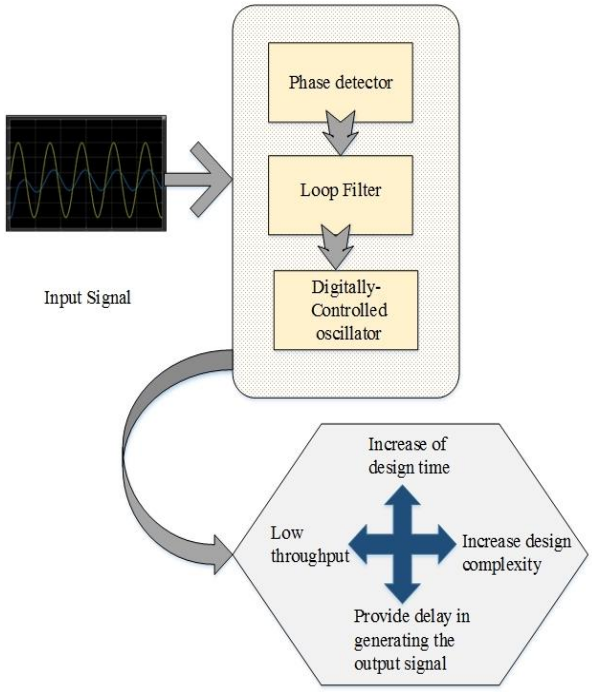

Fig. 1. System model with the problem.

IV. Proposed HH-ADPLL to Make ADPLL Locked State

A novel Hilbert-Huang-based ADPLL (HH-ADPLL) combined with FPGA is proposed to make the ADPLL in a locked state. The presented novel HH-ADPLL is defined in Fig. 2.

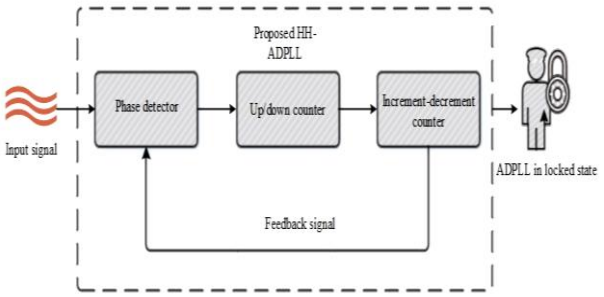

Fig. 2. Proposed HH-ADPLL.

The ADPLL consists of three modules in the presented model: PD, Up/Down Counter as LF, and Increment Decrement Counter as DCO. Initially, the phase detector used the Hilbert-Huang transform to generate an analytic signal. The ref signal and VCO signal are entered into the PD. The PD calculates the difference between these reference signals and the output of the ADPLL design. If the two signals' phase and frequency are matched, the ADPLL is locked. Here, the up/down counter is used as a loop filter to remove the entered signal's high-frequency parts and generate the carry and borrow.

Finally, the increment decrement counter is taken as DCO to generate the output signal concerning the carry, borrow and clock signal. The proposed HH-ADPLL design is implemented on FPGA.

A. Phase Detector

The first module in the proposed ADPLL design is a phase detector. The phase detector accepts two signals: input reference signal and feedback (VCO) signal from the DCO. The phase detector is also known as a phase comparator. Here D-flip-flop matches the phase and frequency difference between input and VCO signals. If the match occurs, the ADPLL will turn into a locked state. Initially, the input signal is given to the phase detector. The Hilbert-Huang transform activated in the phase detector module provides the input signal's analytical components. In the proposed ADPLL structure, phase detection is based on implementing HHT. The HHT is composed of two components. Empirical mode decomposition and Hilbert transform. Here the entered signals are decomposed into a limited and small amount of intrinsic mode function (IMFs) components that are adaptable and highly effective. This decomposition process is suitable for the entire non-stationary and nonlinear signal analysis. Also, the function of the Hilbert transform on the decomposed IMF components attains the important features of the signals, such as frequency, phase and amplitude. The obtained IMFs components are represented in the timescale parameter characterization. Thus the signal is in the combination of the IMF components and residues.

The input signal is composed of several analytical components and residue. It is explained in Eq. (1).

$$x(t) = \sum_{k=1}^{n} c_k(t) + r_s(t) \tag{1}$$

where $x(t)$ denotes the input signals, $c(t)$ represent the analytical and $r_s(t)$ indicate the residue. The process of extracting the analytical components from the input signal is given in Eq. (2).

$$\mathrm{HH}[x(t)] = x(t) - \sum_{k=1}^{n} c_k(t) \tag{2}$$

where HH[·] represents the Hilbert-Huang transform (HHT).

After applying HHT for the input signal in the PD, the input signal can be represented as in Eq. (3),

$$x(t) = \mathrm{Re}\sum_{k=1}^{n} \varphi_k(t) e^{j\int 2\pi\lambda_k(t)dt} \tag{3}$$

where $\lambda_k(t)$ and $\varphi_k(t)$ are the frequency and phase of the reference signal and Re denotes the real part of the complex. Utilizing the analytical components of the reference and the feedback pulse, the phase detector produces an up-and-down signal by comparing the signal's phase difference. The comparison of these two signals is given in Eq. (4),

$$p(t) = g_D[\varphi_k(t) - \varphi_o(t)] \tag{4}$$

where $p(t)$ is the phase detector output signal, g_D is the detector gain, and $\varphi_o(t)$ is the phase of the output VCO signal. These generated up and down signals are then given to the loop filter. In phase detection analysis, using

analytical components instead of the original pulse mitigates the matching estimation of the phase detector. It also removes the cross-term artifacts from the positive and negative frequency components and speeds up the detection process.

B. Loop Filter

The second module of the designed system is the loop filter. The loop filter excludes the high-frequency part of the PD-generated up and down signals. Here, the up/down counter is used as a loop filter and is given in Fig. 3. It works as an integrator. It can be adaptable for all types of phase detectors. The up and down signal from the phase detector is fed into the up/down counter.

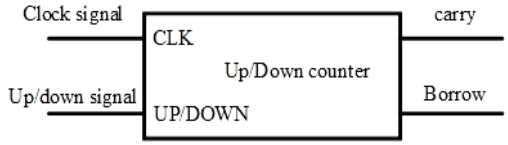

Fig. 3. Up/down counter.

The up/down counter multiplies the frequency of the up and down signal in the range of the clock signal. The clock signal ranges by the multiple of 8, 16, 32, etc., from the frequency of the reference signal. It is given in Eq. (5).

$$\lambda_k = 8N\lambda_{\text{clk}} \tag{5}$$

where λ_{clk} indicates the frequency of the clock signal and N=1, 2, ···, n. Using clock signal, the up/down counter generates a carry and borrow and is given as output. For higher-order applications such as telecommunication, the LF module must require. According to the filter types, the signals are generated. The up-down counter can design the simplest structure of the LF. Thus in the proposed architecture, this counter is used.

C. Digital Controlled Oscillator

The carry and the borrow waveforms generated from the up/down counter are then given to the Digital Controlled Oscillator. It is the third module of the presented model. It generates the output depending on the carry and borrow signal. Here the Increment Decrement Counter is used as a Digital Controlled Oscillator and is given in Fig. 4. The utilization of the Increment Decrement Counter in the DCO module achieves good control over the lock-in Range and hold Range.

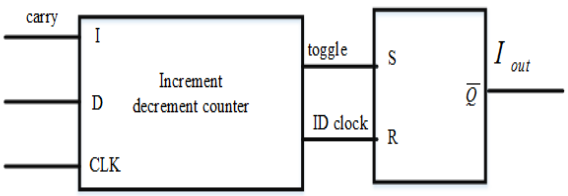

Fig. 4. Increment decrement counter.

The increment decrement counter consists of three gates: I, D, and clock pulse. Here the carry is assigned to the gate I, borrow is assigned to the gate D and the ID clock is taken as a clock pulse. The chosen ID clock is two times the frequency of the input reference signal. Processing these three signals, the Increment Decrement Counter generates a toggle signal. The toggle signal is created based on the carry and the borrow signal. It is given in Eq. (6),

$$l(t) = \lambda_m + K_{\text{DCO}} + v(t) + \text{cb}(t) \tag{6}$$

where λ_m represents the nominal frequency of the oscillator, K_{DCO} represents the oscillator gain, $v(t)$ indicates the input tuning voltage, and cb(t) generates the carry and borrow signal from the loop filter. Finally, the output signal I_{out} is generated by the SR flip-flop by processing the ID clock and toggle pulse.

The steps and processes presented in the designed model were detailed in Algorithm 1. The Xilinx code was executed based on these step processes, and the results were verified. The algorithm incorporated all mathematical function parameters in the pseudocode format.

Algorithm 1: HH-ADPLL

Start
{
//input is the reference signal
Phase detector ()
{
//accepts reference signal and VCO signal as input
// Hilbert-Huang transformation is activated
int $x(t), c(t)$
//Hilbert-Huang transform variables are initialized
$c(t) \rightarrow HH[x(t)]$
//input signal is transformed into an analytical representation
$p(t) \rightarrow \varphi_k(t) - \varphi_o(t)$
//up and down signals are generated
}
Up/down counter ()
{
Higher frequency parts of the signals are removed.
//carry and borrow signals are generated
}
Increment decrement counter ()
{
$I_{out} \rightarrow \overline{toggle + IDclock}$
//output signal is generated matching to the reference signal
}
}
stop

Here the Hilbert-Huang transform extracts only the analytical components of the signals. It provides sufficient parameters of the signal for further processing. So the processing time gets reduced, and power dissipation is also minimized. It makes the ADPLL in a locked state short period. The processes of the proposed HH-ADPLL are given in Fig. 5.

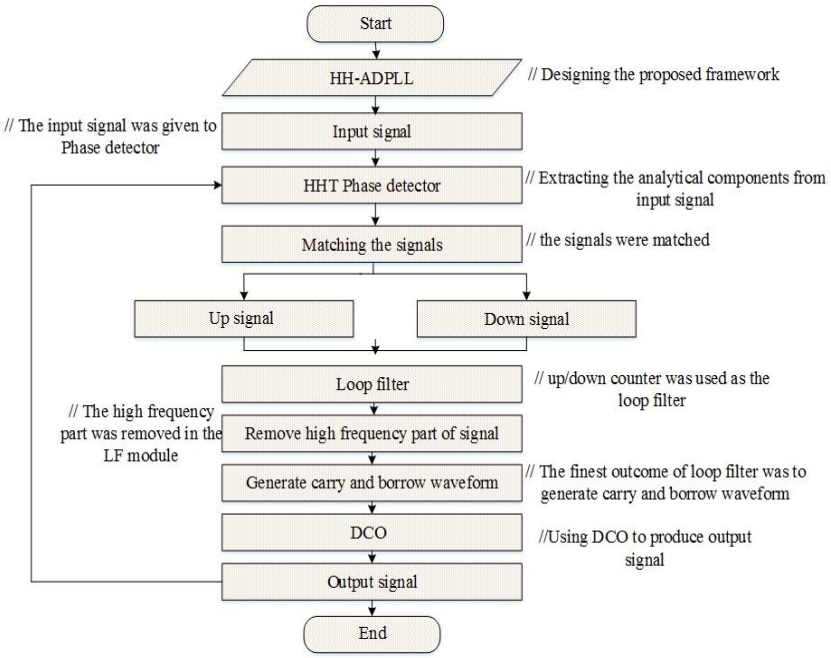

Fig. 5. Flowchart of HH-ADPLL design.

TABLE I: DESCRIPTION OF IMPLEMENTATION PARAMETERS

Designation of execution parameters	
Parameters	Description
Platform	Xilinx ISE
Version	14.7
OS	Windows 10

V. RESULTS AND DISCUSSIONS

The Hilbert-Huang phase detector initially provides the reference signal to extract the analytical component and produce the up and down signals. After, the high-frequency component of these created up and down signals has been filtered by the up/down counter and borrow and carry waveforms are generated. Then the increment decrements counters take these borrow and carry signals to align the output signal with the reference signal. In the phase detector module, input and output signal matching are finally completed, and the ADPLL enters a locked state. A test bench is developed to verify the design, and the simulation is done. In this suggested design, the reference signal is changed every 6 ns, and the output signal is changed every 7 ns. The number of taps used in the FIR filter for the Hilbert transform function is 23 taps. The performance was validated regarding stability, phase error, power dissipation, and combinational delay. The parameters required for implementing the presented model are given in Table I.

A. Case Study

A case study was conducted to study the working of the designed system. Here the initial input is the reference signal. Moreover, the resultant signals are given below after processing the three modules, PD, LF and DCO.

Here the Hilbert-Huang phase detector extracts the analytical components of the signals. The phase and frequency differences are calculated in this module and generate the up and the down signal. The simulation result of the PD is given in Fig. 6.

The high-frequency parts of the signals are neglected and produced a carry and borrow from the up-down counter. The result is in Fig. 7. The increment decrement produced the output signal using the SR flip-flop. The output of the increment decrement counter is given in Fig. 8.

Finally, the generated output signals are given to the PD module, and the matching of reference and the output signal is carried out to make the ADPLL into the locked state. The locked state of the proposed HH-ADPLL is given in Fig. 9.

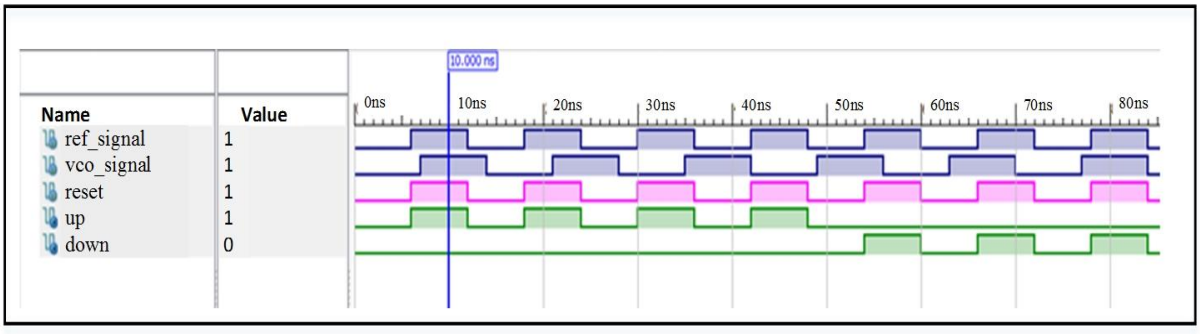

Fig. 6. Hilbert Huang transform-based phase-detector output.

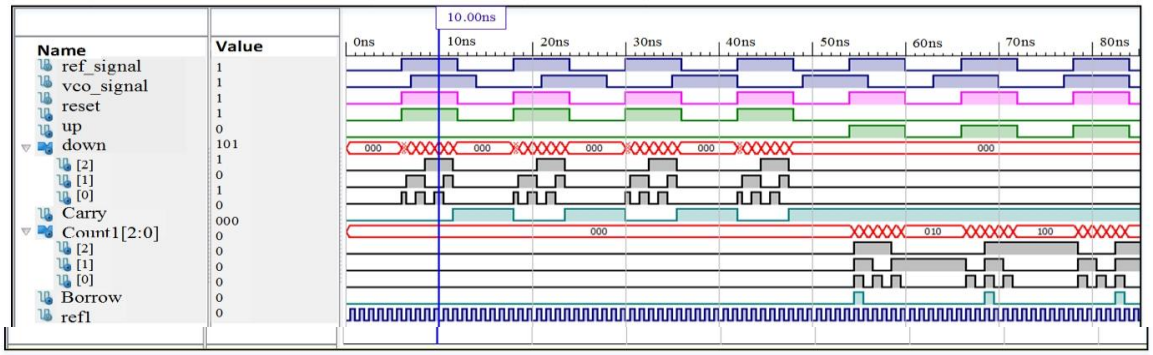

Fig. 7. Simulation result of Up-down counter.

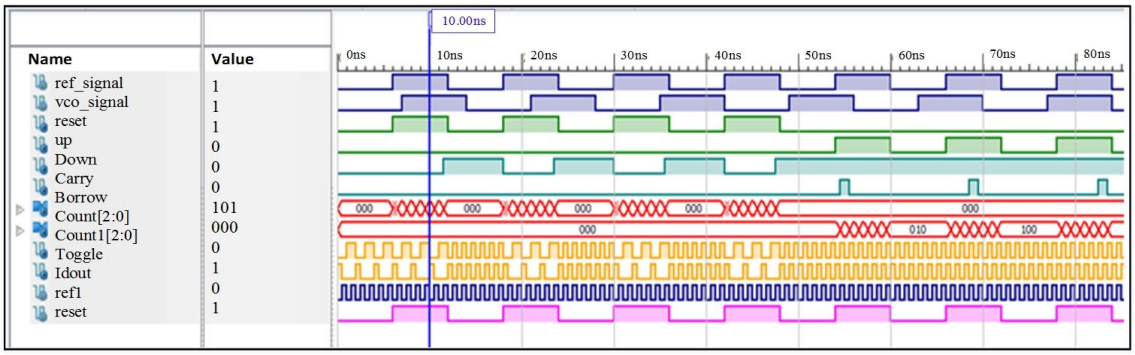

Fig. 8. Simulation result of increment-decrement counter.

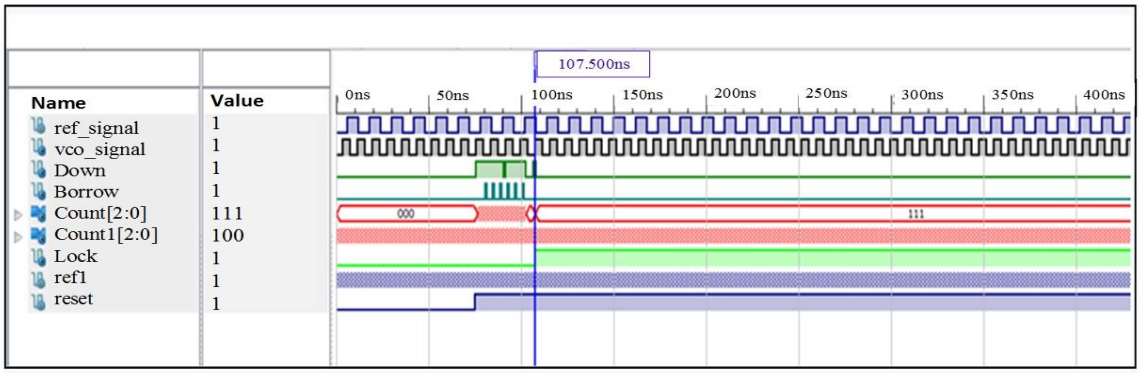

Fig. 9. Locked state of the HH-ADPLL.

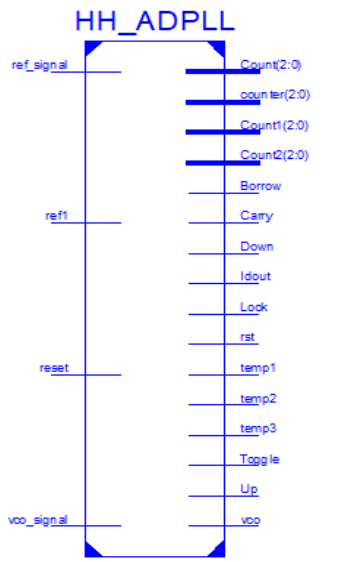

Fig. 10. ADPLL RTL schematic.

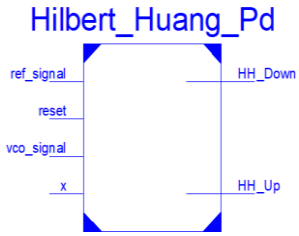

Fig. 11. PD schematic.

After the simulation of the proposed HH-ADPLL, RTL is generated and synthesized, shown in Fig. 10, and the schematic of PD is shown in Fig. 11. The proposed HH-ADPLL has been executed using Xilinx ISE 14.7. Compared to other versions, this version of Xilinx has updated features and enhanced performance. So the architecture is implemented in Xilinx ISE 14.7.

B. Performance Analysis

The presented model is designed using the Verilog HDL and implemented on Artix-7 FPGA evaluation

board. And the simulator used for simulating the Verilog HDL is Xilinx ISE 14.7, running on the windows 10 platform. Here a testing code is written for compilation on a PC to connect with the onboard testing circuit built in the FPGA. The FGPA accepts the test commands from the PC, and the execution parameters are adjusted according to the test modes. In addition, the results are recorded by monitoring the ADPLL architecture during operation. The terms such as phase error, combinational delay, power dissipation, power consumption and stability are computed to validate the performance of the proposed model with existing models such as 2.4 GHz Low Power All Digital Phase Locked Loop (2.4 LPADPLL) [26], Variable Phase Accumulator (VPAC) [27], All Digital Phase Locked Loop with Optimized Digital Controlled Oscillator (ADPLL-ODCO) [28], All Digital Phase Locked Loop with Varactorless LC Digital Controlled Oscillator (ADPLL-VLCDCO) [29] and ADPLL for True Random Number Generator (ADPLL-TRNG) [22].

1) Power Dissipation

Power dissipation is evaluated by the ratio of the total power supplied to the total power loss. It is the maximum power that is dissipated under certain thermal conditions. It describes the loss or waste of energy in heat format during a certain process. In the presented architecture, the power dissipation is calculated according to the utilized logic circuits, average operating frequency and load conditions. The basic formula for power dissipation is given in Eq. (7).

$$\text{power dissipation} = \frac{\text{total power supplied}}{\text{power loss}} \tag{7}$$

The power dissipation of the existing model, such as ADPLL-VLCDCO, scored 11 mW, LPADPLL scored 12 mW, VPAC scored 14.1 mW, and ADPLL-ODCO scored 2.83 mW. At the same time, the presented model scored power dissipation as 1.8 mW, which is lower than the other existing models. The comparEison of power dissipations with existing models is given in Fig. 12.

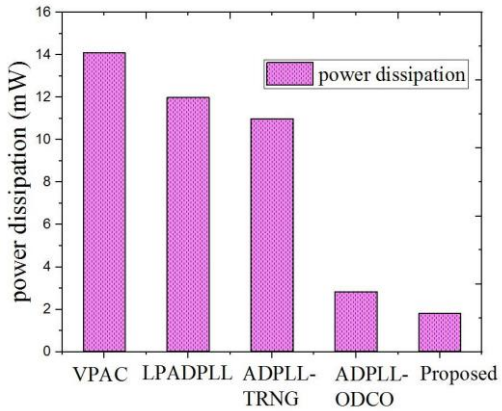

Fig. 12. Comparison of power dissipation with existing techniques.

2) Combinational Delay

Combinational delay is the time that must pass between the application of an input signal and its stabilization at the circuit's input before the circuit's output is stable enough to produce the intended output signal.

The combinational delay existing models such as 2.4 LPADPLL scored 2.068 ns, VPAC scored 4.77 ns, ADPLL-TRNG is 1.479 ns, and ADPLL-ODCO scored 12.997 ns. At the same time, the presented model scored power dissipation as 1.5 ns. The comparisons of combinational delays with existing models are given in Fig. 13.

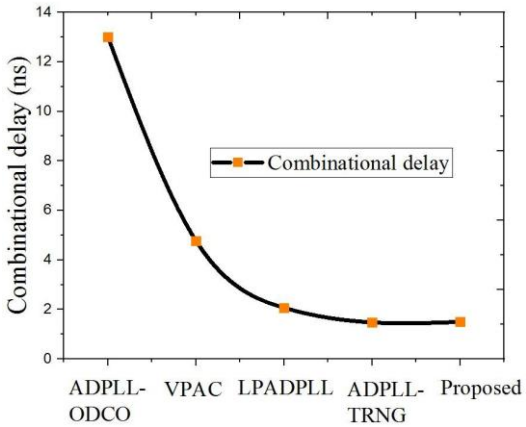

Fig. 13. Comparison of combinational delay with existing techniques.

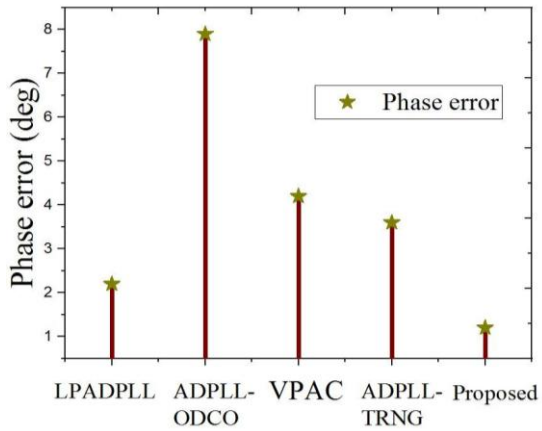

Fig. 14. Comparison of phase error with existing techniques.

3) Phase Error

The phase error is defined as the variation in the phase of the reference signal and measured output signal. The phase error results from the combinational delay. The phase error can be calculated by Eq. (8).

$$\text{Error} = \left[e(n) - \lambda\right] T_{\text{DCO}} \tag{8}$$

The presented model scored a phase error ±1.2° lower than the other existing models. The comparisons of combinational delays with existing models are given in Fig. 14.

The presented model achieved extremely low power dissipation, which is lower than the other current models and starts at 1.8 mW. Additionally, the developed circuit has a combinational delay of 1.5 ns, less than the current methods. The phase error in the developed model is ±1.2°. The proposed model's power dissipation is 1mW lower than the other existing approaches. This model achieves

the lock state faster. The overall comparison of HH-ADPLL is given in Fig. 15, and the comparison of other parameters such as power consumption and lock-in time is noted in Table II.

TABLE II: COMPARISON OF POWER CONSUMPTION AND LOCK-IN TIME

Method	Power consumption (mW)	Lock in time (μs)
LPADPLL	12	4.2
VPAC	14	2.8
ADPLL-ODCO	5.8	0.6
ADPLL-TRNG	4	0.28
Proposed	3.2	0.1075

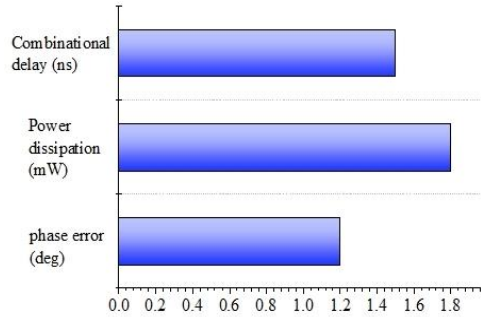

Fig. 15. Overall performance of the proposed method.

The main aim of the ADPLL is the lock state. The suggested HH-ADPLL design is utilized for the frequency applications. It provides better locking between the output and the reference signal. As getting the minimum phase error, the up and down signal is not generated, and the system gets into the locked state. Also, the power dissipation caused by the DCO is reduced due to the short lock-in time. These overall performance increases the efficiency of the suggested system to achieve the lock state quickly.

Here the performance is not calculated by RTL synthesis because, in the process, the circuits are not activated accurately for the entire architecture, and the major activity factors will change throughout the chip; hence this is not liable and is prone to error. Although the total power is calculated correctly using the selected activity factor, the breakdown of that power into logic, clock, memory, etc., is less precise. Also, the present FPGA is sufficient for evaluating the power dissipation etc., so we do not use the OpenLane.

C. Discussion

TABLE III: OVERALL PERFORMANCE

Overall Performance statistics	
Stability	Good
Phase error	±1.2°
Power dissipation	1.8 mW
Combinational delay	15 ns
Acquisition range	330.360 MHz
Lock-in-time	107.500 ns
Lock range	9302.32558 kHz
Phase noise measurement	−130.4 dBc/Hz at 100 MHz offset
Hold range	0.189 −0.250 MHz
Frequency hold during loss of reference	0−214 kHz
Reacquisition after loss of reference	300.360 MHz

Here, the phase detector initially employed the Hilbert Huang transform to create analytical components from the provided reference signal. The phase detector is fed with the ref signal and VCO signal. The PD computes these two signals' differences. ADPLL enters a lock state if the two signals' frequency and phase coincide. Here, the high-frequency components are removed using an up/down counter acting as an LF, and the carry and borrow signal is produced. Using these carry and borrow signals, the DCO generates the output signal matching the reference signal to get ADPLL into a locked state. The over-efficiency gained by the presented model is given in Table III.

VI. CONCLUSION

The novel Hilbert-Huang-based All Digital Phase Locked Loop (HH-ADPLL) is proposed to make the ADPLL in locked state. Initially, the input signal is inserted into the Hilbert-Huang PD to extract the analytic component and generate the up and down signals. These generated up and down signals are then given to the up/down counter to eliminate the high-frequency part, and the carry and borrow signals are generated. These carry and borrow signals are then fed into the increment decrement counter to match the output signal to the reference signal. At last, input and output signal matching is carried out in the phase detector module, and ADPLL goes into a locked state. The presented HH-ADPLL is implemented in Artix-7 FPGA, and the efficiency is validated in terms of stability, phase error, power dissipation, and combinational delay. The developed model attained very low power dissipation, which ranges from 1.8mW, which is lower than the other existing models. Also, the combinational delay of the designed circuit is 1.5 ns which are lower than the existing approaches. One of the existing methods scored a power dissipation of 2.83 mW. Comparing to 1mW of reduced in the proposed architecture power dissipation. So, the power dissipation in the proposed model is 1mW reduced. This model takes less time to attain the lock state. Here the suggested ADPLL architecture is not implemented in real-time. Also, acquiring the phase information eases the matching process of the PD to attain a quick lock state. In the future, ADPLL with efficient optimization techniques would improve the method and also for real-time implementation. Additionally, the present trend in applications for digital signal processing indicates a shift toward ultra-short waveband and above, and this ADPLL can still be altered and improved in the near future to meet the demands of upcoming requirements for such designs. Future improvements can be made to the suggested low power, high speed, and area-efficient ADPLL design compared to its analogue predecessor.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

Both the authors contributed equally.

REFERENCES

[1] S. Xu, L. Tang, and J. Yang, "Time-domain modelling and performance research of millimeter-wave all-digital phase-locked loop," *Journal of Physics: Conference Series*, vol. 2245, #012018, 2022.

[2] C. Zhang, S. Wang, R. He, Q. Zhao, and K. Wang, "Design and construction of a low cost all-digital phase locked loop based on field programmable gate array," *Journal of Physics: Conference Series*, 2021, vol. 1972, #012054, 2021.

[3] M. Kumm, H. Klingbeil, and P. Zipf, "An FPGA-based linear all-digital phase-locked loop," *IEEE Trans. on Circuits and Systems I: Regular Papers*, vol. 57, no. 9, pp. 2487-97, April 2010.

[4] D. H. Badarov and G. S. Mihov, "Stability study of the digital phase locked loop implemented on Xilinx FPGA," in *Proc. of 2018 IEEE XXVII Int. Scientific Conf. Electronics-ET*, 2018, pp. 1-4, doi: 10.1109/ET.2018.8549622

[5] S. Edway and R. K. Manjunath, "Design and simulation of FPGA based all digital phase locked loop (ADPLL)," in *Proc. of 2017 3rd Int. Conf. on Applied and Theoretical Computing and Communication Technology*, 2017, pp. 259-263.

[6] S. N. Ishak, J. Sample, Z. Yusoff, and M. Faseehuddin, "All-digital phase locked loop (ADPLL) topologies for RFID system application: A review," *Jurnal Teknologi*, vol. 84, no. 1, pp. 219-30, 2022.

[7] M. R. Rehman, A. Hejazi, I. Ali, M. Asif, S. Oh, P. Kumar, Y. Pu, S. S. Yoo, K. C. Hwang, Y. Yang, and Y. Jung, "An ultra-low-power 2.4 GHz all-digital phase-locked loop with injection-locked frequency multiplier and continuous frequency tracking," *IEEE Access*, vol. 9, #152984-92, October 2021.

[8] Z. Wang, H. Ding, H. Xu, C. Zhang, S. Hu, X. Ji, X. Xia, and Z. Cai, "An all-digital phase-locked loop with a PGTA-based TDC and a 0.6-V DCO," *IEICE Electronics Express*, vol. 15, no. 22, pp. 1-15, 2018.

[9] S. Selvaraj, E. Bayram, and R. Negra, "System-level model of a two-step locking technique applied in an all-digital Phase-locked loop," in *Proc. of 2022 11th Int. Conf. on Modern Circuits and Systems Technologies*, 2022, pp. 1-4, doi: 10.1109/MOCAST54814.2022.9837730

[10] P. Zhang, H. Fang, Y. Li, and C. Feng, "Fast single-phase all-digital phase-locked loop for grid synchronization under distorted grid conditions," *Journal of Power Electronics*, vol. 18, no. 5, pp. 1523-35, 2018.

[11] M. Kumm, H. Klingbeil, and P. Zipf, "An FPGA-based linear all-digital phase-locked loop," *IEEE Trans. on Circuits and Systems I: Regular Papers*, vol. 57, no. 9, pp. 2487-97, April 2010.

[12] C. Dooley, E. Blokhina, B. Mulkeen, and D. Galayko, "FPGA Based Modelling of an ADPLL Network," in *Proc. of 2019 16th Int. Conf. on Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuit Design*, 2019, pp. 289-292.

[13] L. Xiu, X. Wei, and Y. Ma, "A full digital fractional-$ N $ TAF-FLL for digital applications: demonstration of the principle of a frequency-locked loop built on time-average-frequency," *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, vol. 27, no. 3, pp. 524-34, Jan. 2019.

[14] Y. Tang, T. Townsend, H. Deng, Y. Liu, R. Zhang, and J. Chen, "A highly linear FPGA-based TDC and a low-power multichannel readout ASIC with a shared SAR ADC for SiPM detectors," *IEEE Trans. on Nuclear Science*, vol. 68, no. 8, pp. 2286-93, July 2021.

[15] H. B. Meitei and M. Kumar, "FPGA implementation of a wireless communication system for secure IR sensor data transmission using TRNG," *Int. Journal of Engineering Trends and Technology*, vol. 70, no. 7, pp. 220-237, 2022.

[16] R. Khalirbaginov and H. Alexandr, "Novel HDL design of digital controlled oscillator for ADPLL," in *Proc. of 2019 IEEE Conf. of Russian Young Researchers in Electrical and Electronic Engineering*, 2019, pp. 1678-1682.

[17] X. Chen, J. Breiholz, F. B. Yahya, C. J. Lukas, H. S. Kim, B. H. Calhoun, and D. D. Wentzloff, "Analysis and design of an ultra-low-power Bluetooth low-energy transmitter with ring oscillator-based ADPLL and 4 × frequency edge combiner," *IEEE Journal of Solid-State Circuits*, vol. 54, no. 5, pp. 1339-1350, February 2019.

[18] D. H. Badarov and G. S. Mihov, "All-digital phase locked loop with single reference frequency oscillator," in *Proc. of 2020 XXIX Int. Scientific Conf. Electronics (ET)*, 2020, pp. 1-4.

[19] M. Salarpour, F. Farzaneh, and R. B. Staszewski, "Synchronization-phase alignment of all-digital phase-locked loop chips for a 60-GHz MIMO transmitter and evaluation of phase noise effects," *IEEE Trans. on Microwave Theory and Techniques*, vol. 67, no. 7, pp. 3187-3199, May 2019.

[20] M. Salarpour, R. B. Staszewski, and F. Farzaneh, "A mm-wave MIMO transmitter with a digital beam steering capability using CMOS all-digital phase-locked loop chips," in *Proc. of 2018 IEEE MTT-S Int. Microwave Workshop Series on 5G Hardware and System Technologies*, 2018, pp. 1-3, doi: 10.1109/IMWS-5G.2018.8484613.

[21] S. Radhapuram, T. Yoshihara, and T. Matsuoka, "Design and emulation of all-digital phase-locked loop on FPGA," *Electronics*, vol. 8, no. 11, #1307, November 2019.

[22] H. B. Meitei and M. Kumar, "FPGA implementation of true random number generator architecture using all digital phase-locked loop," *IETE Journal of Research*, vol. 68, no. 3, pp. 1561-70, May 2022.

[23] E. Koskin, P. Bisiaux, D. Galayko, and E. Blokhina, "Jitter optimization in a generalized all-digital phase-locked loop model," *IEEE Trans. on Circuits and Systems II: Express Briefs*, vol. 68, no. 1, pp. 77-81, July 2020.

[24] E. Koskin, P. Bisioux, D. Galayko, and E. Blokhina, "FPGA Validation of event-driven ADPLL," in *Proc. of 2020 European Conf. on Circuit Theory and Design*, 2020, pp 1-4, doi: 10.1109/ECCTD49232.2020.9218367c

[25] R. Dinesh and R. Marimuthu, "A wide range high resolution digital controlled oscillator with high precision time to digital converter for optimal sampling digital PLL," *Microprocessors and Microsystems*, vol. 74, #103013, April 2020.

[26] L. Xu, S. Lindfors, K. Stadius, and J. Ryynanen, "A 2.4-GHz low-power all-digital phase-locked loop," *IEEE Journal of Solid-State Circuits*, vol. 45, no. 8, pp. 1513-21, July 2010.

[27] S. Saad, M. Mhiri, A. B. Hammadi, and K. Besbes, "An enhanced variable phase accumulator with minimal hardware complexity dedicated to ADPLL applications," in *Proc. of 2018 15th Int. Multi-Conf. on Systems, Signals & Devices*, 2018, pp. 1447-1452.

[28] D. C. Lee, K. Y. Kim, Y. J. Min, K. M. Kim, A. Abdullah, J. Park, and S. W. Kim, "A low power all-digital PLL with power-optimized digitally controlled oscillator," in *Proc. of 2010 IEEE Int. Conf. of Electron Devices and Solid-State Circuits*, 2010, pp. 1-4, doi: 10.1109/EDSSC.2010.5713739c

[29] S. Biereigel, S. Kulis, P. Moreira, A. Kölpin, P. Leroux, J. Prinzie, "Radiation-tolerant all-digital PLL/CDR with varactorless LC DCO in 65 nm CMOS," *Electronics*, vol. 10, no. 22, p. 2741, Nov. 2021.

Velamarthi Spandana received B.Tech degree from SLCs College of Engineering and Technology and M.Tech. degree from JNTU Hyderabad. She is now pursuing Ph.D. degree from Osmania University. She has published a paper ICPET-2016 on Implementation of Video Finger printing on compressed dual video watermarking for real time application. Her area of interests is Digital Systems, Signal Processing, digital and analog VLSI.

Chandra Sekhar Paidimarry received BE degree from Nagpur University in 1991, M.Tech degree from JNTU Hyderabad in 1999 and Ph.D. from Osmania University in 2009. He had been awarded with Post-Doctoral Fellowship by Shizuoka University, Japan for one year. Prior to joining in teaching, he has eight years of industrial experience of design and development of Embedded Systems. He has been working in the Department of Electronics and Communication Engineering, Osmania

College of Engineering, Osmania University, Hyderabad from 2001. He has been elevated as Professor of ECE in 2015. Currently; he is working as a Vice Principal for University College of Engineering, Osmania University. He has more than 50 research publications to his credit. He delivered more than 15 invited talks and guest lecturers in various conference and events. His research interests include Development of high performance computational Electromagnetic and efficient FPGA based signal processing algorithms and Design Automation.