International Journal of Engineering Trends and Technology
ISSN: 2231 – 5381 / https://doi.org/10.14445/22315381/IJETT-V70I7P223

Volume 70 Issue 7, 220-237, July 2022

Original Article

FPGA Implementation of a Wireless Communication System for Secure IR Sensor Data Transmission using TRNG

Huirem Bharat Meitei[1], Manoj Kumar[2]

[1,2]*Department of ECE, NIT, Langol, Manipur, India*

[1]thinktank453@gmail.com

Received: 10 May 2022 Revised: 08 June 2022 Accepted: 10 July 2022 Published: 20 July 2022

Abstract *- This article implemented an application of True Random Number Generators based (TRNG) design using All Digital Phase lock loop (ADPLL) to send secure IR (Infrared) sensor data wirelessly using Bluetooth HC-05 Module. This unique approach for secure wireless transfer of IR sensor data is realized on the Field Programmable Gate Array and designed with Xilinx Vivado. TRNG using ADPLL is made with NOR gate-based ring oscillators (RO) and flip-flops(FF) which are used to generate different forms of entropy. The key for the IR sensor information encryption algorithm in this study is comprised of random bits generated by a TRNG. Transferred active low-IR sensor data encrypted using TRNG to another device wirelessly via Bluetooth module to achieve secure wireless data transmission. The Arduino UNO board is used to interface encrypted/decrypted IR sensor data, HC05 Bluetooth enslaver/slave data, and display the data. The scientific novelty of this study is developing an application that uses an ADPLL-based TRNG as a cryptographic system to transmit safe IR sensor data wirelessly via a Bluetooth module. Unlike prior approaches, this work demonstrates that a viable FPGA-based wireless application utilizing TRNG may be produced by primarily utilizing the onboard ADPLL and ring Oscillator, as well as a few basic logic parts (1 LUT (Lookup Table) for Transmitter Tx and 2 LUTs for Receiver Rx). Passing the NIST test demonstrates the unpredictable and unique nature of the output TRNG bitstreams employed in the architectures. As a result, the suggested wireless application utilizing the TRNG cryptographic approach is suitable for use in various fields, including security network systems and industrial applications.*

Keywords *- TRNG, ADPLL, HC05, IR-Sensor, FPGA.*

1. Introduction

Humans have grappled with the issue of data security from the beginning of the 21st century. As a result of their perspective, numerous ways of protecting ICs (integrated circuits) have been developed. Almost all cryptographic operations require random numbers. A randomly generated sequence of numbers is required for all encrypted structures, including initialization vectors, block padding, nonces, and keys. Since most of these random bit sequences were broadcast in the open domain, a passive hacker has a decent chance of evaluating the RNG's output and exploiting any detected vulnerabilities. As a result, RNGs (Random Number Generators) used in encryption methods should always be viewed as a critical component of the overall encryption process. A vulnerability or flaw in the RNG can result in the system's complete failure [1]. The famed Netscape V2.0 browser hack exemplifies a successful attack on a weak random number generator [2]. With an increasing reliance on data from various apps, smart devices, and sensors, the integrity of the transmission network gets critical. Consumer confidentiality must be appropriately secured using a secure yet robust RNG, such as the TRNG. Random number generators are indispensable units of all encryption techniques since these were used in block ciphers, digital signatures [3], and one-time padding [9].

Additionally, the topic of RNG discusses Deterministic random bit generators (DRBGs) based on hash functions and the Secure Hash Algorithm SHA-256 cryptographic approach [4]. With rapid digitalization, data protection is becoming more important in many digital applications, and cyber security threats are attracting more attention worldwide [5]. In recent years, as network (wired or wireless) communications of digital technologies have become more popular, the security of data transmission, which includes information data, voice, images, and video, is becoming extremely significant [6]. As a result, the demand for secure encrypted communication systems is rapidly increasing. Numerous cryptographic techniques have been developed to overcome these limitations.

In this case, cryptography's fundamental difficulty is the generation of lengthy, unpredictable random numbers. This requirement may generally be satisfied by merging different software or hardware designs capable of generating randomly generated permutations and providing the necessary secret keys for efficient encryption algorithms [7]. The greater the random bitstream's unpredictability, the higher the quality of the random number, and the less likely an attacker will be able to retrieve its information [8]. Today's growing demand for wireless communication users has heightened the importance of security and protecting information transferred by the user over an unsecured network from unauthorized access. Because data is shared across a wireless network, it must be authenticated and well protected. The most critical aspect of wireless communications is the security between different devices. This study aims to propose a secure connection network that uses a TRNG as the encryption architecture and is based on ADPLL. Bluetooth technology is used to create the wireless connection, whereas TRNG, based on a single ADPLL [10], establishes a secure algorithm for data exchange. Building a network using the embedded Bluetooth technology allows the equipment to communicate with one another in low-power standalone mode anywhere. This wireless technique is highly advantageous in the residential area since there is minimal hardware for smart equipment communication. It could be cost-effectively used for residential automation. Working at the unlicensed, widely used 2.4 GHz frequency, it can connect electronic technology inside a radius of 10 meters (expandable to 100 meters by enhancing the signal strength) at a rate of 1 Mbps [11]. TRNG is more secure due to the absence of similarity between the last generated bitstream and the preceding one [8]. The paper proposes a secure mechanism for defending against physical attacks on the security and confidentiality of applications and sensitive data. FPGA implementation of IR sensor data transmission and reception using a TRNG-based wireless communication system is used to establish a configurable framework for safe wireless data transfer utilizing a TRNG with a single ADPLL as the cryptographic approach. An efficient implementation of the complex cryptographic algorithms employing TRNG was accomplished on the Artrix-7 (XC7A35T-CPG236-1) FPGA board. This technology is developed for extensively utilized in portable electronics, where power usage is a critical factor to consider.

Moreover, emulation of the ADPLL in FPGA can minimize design duration and complication while also improving the efficiency and performance of the ADPLL [12]. Three key design objectives for hardware implementation are reduced area or cost, reduced latency to reduce the time required to encrypt a single data sequence, and maximum throughput to encode many cipher codes concurrently [13]. These performance criteria include a balance between area, speed, and cost. Bluetooth technology was preferred above others because it is already built into most smart devices, is inexpensive to integrate, requires little energy, and gives security over short distances [14]. Whereas FPGAs are a suitable candidate for incorporating programmable hardware in cellular communication systems. FPGAs are commonly used in digital signal computation and telecommunications technology. FPGAs' properties, such as high parallelism, large gate counts, and low-power packaging, allow for significant savings in memory usage, processing complexity, and energy consumption [16].

Pandey et al. [15] used the PRESENT lightweight block cipher technique to complete the encryption and decryption operations. The designed PRESENT system handled 64-bit inputs and a key size of 80/128 bits. The OTF (Open Telecommunications Framework) design was also used to calculate the initial key for the dynamic keys. The encryption/decryption operation was then completed using the created intermediate keys. The iterative approach mentioned in the decryption is applied to obtain a better compromise between time and area. The energy usage of the PRESENT design is significant at a lower frequency when a substantial number of keywords (128 bit) were processed. Cherkaoui et al. [17], drawing inspiration from Sunar et al., developed a unique concept wherein the ROs were substituted with a Self-Timed Ring (STR). An STR is a multistage synthesizer capable of maintaining a constant phase difference between its output signals. As a result, this architecture is resilient to the most often exploited flaws in TRNGs with ROs. Cherkaoui et al. TRNG design . It appears robust (no known vulnerabilities) but consumes a substantial portion of the power and board area. Paper [18] presents a novel wireless digital cryptography hyperchaotic network infrastructure based on radio frequency (RF) networking standards for protected real-time data or information transmission. Here the acquired simulation results indicated the utility of integrating the Zigbee (Xbee or Wireless Fidelity) standard, which is recognized for its strong noise resistance, with hyperchaotic communications security. But due to the high power consumption by its wireless network, this system is unsuitable for many portable standalone devices.

SensoRNG is an experimental framework that serves as a concept for evaluating the performance of a sensor-based RNG, as demonstrated in [19]. In their paper, the authors provide an exploratory analysis of random number generation on sensor-equipped devices and techniques for their implementation. The author validated the claims using a prototype model entitled SensoRNG, which employs sensor noise as the seed for entropy to produce random numbers. [7] Discusses the development and operation of a TRNG design on a Xilinx XCKU040 FPGA. Entropy is generated by jitter from the (Phase-locked Loop) PLL and metastability of the Flip-Flop. In addition, the results demonstrate that a TRNG design with an output of 100 Mbps bitrate may be constructed using just eight D-Type FF, seventeen LUTs, two

counters, and a lesser number of Configurable Logic Blocks (CLB). Due to the huge number of resources, the design becomes more complex and consumes a lot of energy. [20] Proposed a high-speed chaos-based RNG be developed with field programming gate array based on their chaotic design. The architecture operates at 293 MHz, provides an output bit rate of 58.76 Mbps, and meets both NIST 800-22 and Federal Information Processing Standard (FIPS 140-1) specifications [21]. Presented a TRNG entirely dependent on a ring oscillator's metastability entropy. Moreover, according to the paper, a digitized TRNG depending on a Meta-Ring oscillator generates the highest level of entropy, resulting in a significant increase in the nominal 140 Mbps throughput. The entropy seeds generated from the ring oscillator's metastability are not enough to produce high quality unique, unpredictable random sequences. Hence this design can be readily attacked at a different level while embedded in the wireless communication system. The ring oscillator RNG scheme was initially presented by [22] and utilized some simple ring oscillators XORed and processed by a simple D-type flip-flop (D-FF). The disadvantages of this approach, despite the simplicity of the design, are that it requires a great deal of area on an IC and consumes a great deal of energy. Moreover, this circuit needs to be protected from external changes due to the power supply and substrate signal; otherwise, attacking this generator becomes easier. The author of [23] proposes a TRNG with an FIR as the loop filter of the ADPLL. Here the design used 3rd order low pass filter as the loop filter instead of a conventional K-counter, which reduced power consumption to 0.072w for FAT-1. But one of the major disadvantages of the design is the low throughput of 200Mbps, which will be very difficult in wireless transfer-based applications like IoT and other designs.

The proposed wireless system for secure IR sensor data transfer application using ADPLL-based TRNG architecture has key advantages over the existing designs. TRNGs used in our wireless application for encrypting based on the digitized ADPLL have various advantages over PLL-based TRNGs, including low power consumption, reduced area requirements, ease of synthesizability, and the ability to be entirely redesigned in less time. Moreover, TRNGs based on a programmable FPGA architecture is more adaptable, quicker, and simpler to operate than TRNGs based on analog circuits [24]. It is completely independent of the FPGA manufacturer, does not require human placement, and is routed automatically during manufacturing, resulting in a very portable gadget. Additionally, they build a network with the embedded Bluetooth technology, allowing appliances to communicate with one another in low-power standalone mode anywhere. Due to its presence in most consumable electronics devices, the Bluetooth base system can be integrated at a low cost. Most importantly, it provides security through its short-range applications and pairing function [52]. This wireless approach is especially suitable in household scenarios, where smart device connecting infrastructure is limited. This design flexibility permits the use of a diverse set of algorithms to accomplish certain tasks, as well as the creation of acceptable designs based on simulation outputs. The wireless communication system with a secured IR sensor data transferred based on TRNGs is divided into the following sections: 1. Introduction 2. Proposed Wireless communication system for secure IR sensor data transmission 3. Description of the used TRNG designs with a single ADPLL. 4. Implementation of the proposed transmitter and receiver architecture.5. Experimental results for the proposed TRNG-based wireless communication system. 6. AT command mode for HC-05 module configuration. 7. Comparison and Discussion with the existing design. 8. Discussion 9. Conclusion and Future Work.

2. Proposed wireless Communication systems for secure IR sensor data Transmission

Fig. 1. depicts the system's block diagram, split into two primary sections: the transmitter and the receiver. In the transmitter architecture, the artix7 FPGA board XORs (exclusive OR) the original active low IR sensor data with TRNG for encrypting it. In our suggested system, data transfer is accomplished by employing infrared sensors. It consists of a transmitted LED and a receiver photodiode that detect invisible rays emerging from a digitally encoded item. The Arduino UNO board can be used as a rapid development tool for Very Large-Scale Integration (VLSI) test benches, particularly for sensors. By integrating Bluetooth, the Arduino UNO and the FPGA device become more suitable for designing wireless signal applications. The FPGA is the most versatile technology available for developing hardware for virtually any purpose. The FPGA allows genuine scalability and flexibility with its inbuilt, fully adjustable soft processing capacity.

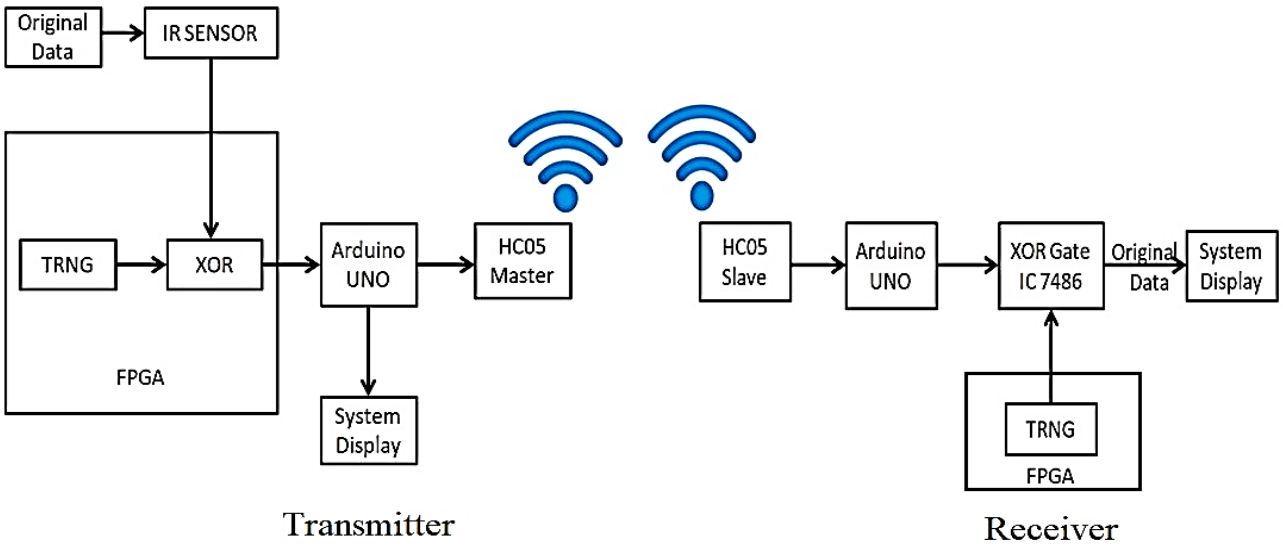
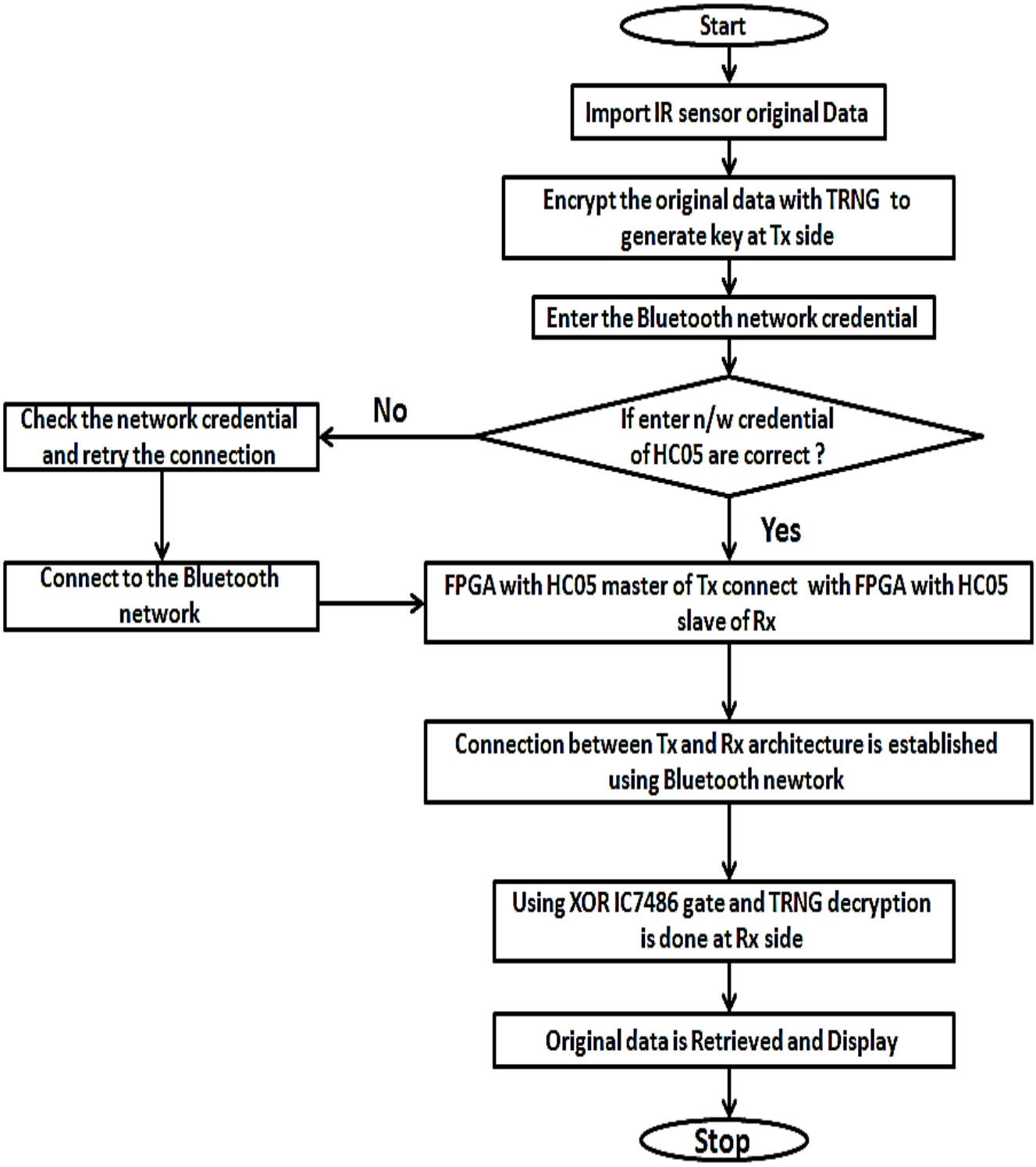


Fig. 1 Block diagram of the TRNG-based proposed wireless system

Additionally, it may satisfy specific requirements by combining programmable hardware and software on a single chip. In this project, the Bluetooth module used is the HC-05, a simple Bluetooth Serial Interface Communication device module. It is designed to facilitate the establishment of seamless wireless serial communication. Bluetooth Serial Interface Version 2.0 is approved and supports data transfer rates of up to 3Mbps [25]. Model number 05 denotes that the Bluetooth device can be used as an enslaver or an enslaved person [25]. Wireless signal transfer is achievable once the FPGA board and an Android smartphone are connected. Wireless technology is likely the most promising and extensively used technology for remotely controlling equipment. So the work provided in this paper can be used to construct effective remote control applications with simple tweaks and upgrades. The Infrared sensor is a type of electronic device which used to identify particular environmental parameters by producing or detecting infrared light. Such sensors could also sense or monitor the temperature and movement of an object. The IR sensor module includes five essential parts Tx (transmitter), Rx (receiver), operational amplifier, trimmer pot (variable resistor), and output LED [26]. IR sensor data can be used for object detection and as a source of entropy in TRNG architecture.

The 74HCT86 is a CMOS logic quad 2-input XOR Gate with a fast speed. EOR gates and EXOR gates are two other names for XOR gates. Logic gates use silicon gate (complementary metal-oxide-semiconductor) CMOS technology to achieve similar operating speeds as LS-TTL gates while using less power. Buffer circuitry, controlled inverter circuits, and other circuits regularly use it. Simply connect Vcc (pin 14) and ground (pin 7) to power the 74HCT86 XOR Gate IC. The typical working voltage of the IC is +5V. The outputting voltage on pin Y of the IC will be the same as the IC's operational voltage [27]. The XOR gate IC 74HCT86 is employed in our proposed system to decode the encrypted sensor data sent by the transmitter, allowing the raw data to be successfully retrieved at the receiver end.



Design Flowchart. 1 Overall System process flowchart for proposed TRNG-based wireless system

3. Description of the used TRNG design with a single ADPLL

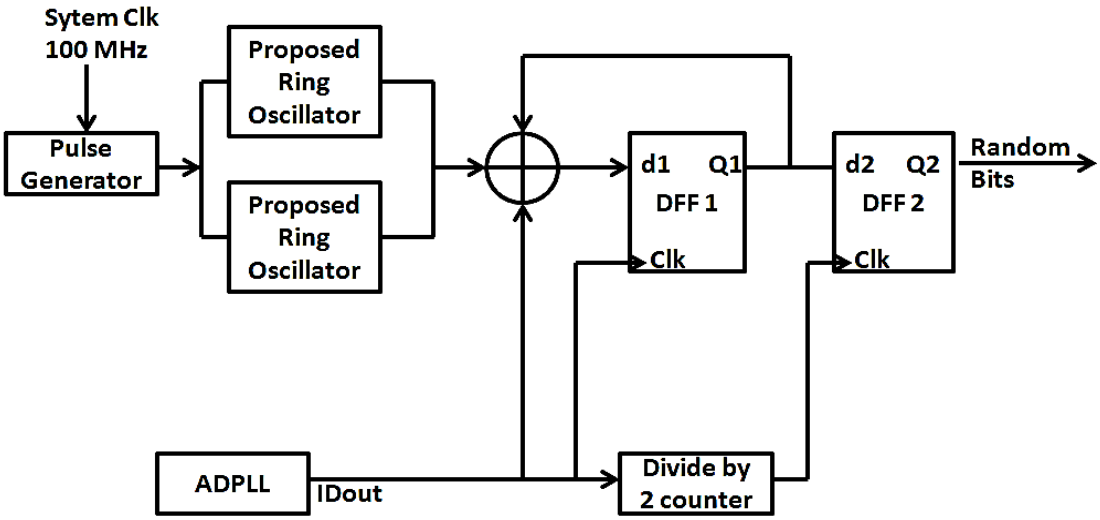


Fig. 2 Used ADPLL-based TRNG architecture in the proposed wireless communication system [10]

Fig.2 illustrates the single ADPLL-based TRNG architecture for encrypting the transmitted IR sensor data for securing data transmission used in our proposed wireless communication system.

3.1. Details used of ADPLL for implementing TRNG Architecture

We cascade our IR sensor with an ADPLL-based design architecture with ring oscillators and a flip-flop (FF) to generate the entire entropy source necessary for producing the random binary string with the suggested TRNG-based wireless communication system. ADPLL is a digital circuitry system that can be efficiently replicated on an FPGA [28]. ADPLL is a digital implementation of PLLs [29]. It is made up of three parts: (i) phase detectors (PDs); (ii) loop filters (LFs); and (iii) DCO.

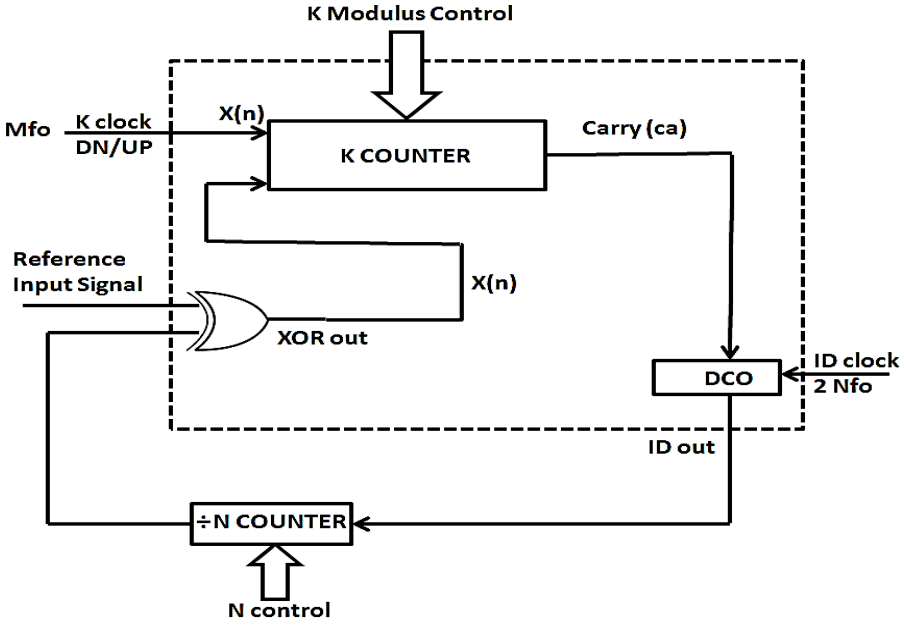


Fig. 3 Circuit diagram of an ADPLL [10]

The basic ADPLL block design is shown in Fig. 3, and most of the blocks are represented digitally [30]. XOR-Gate [31] is the phase detector. An ADPLL is used to establish a link between the input phase and the output phase, as well as the frequency. As a result, PD is utilized in ADPLL [32] to minimize the differential between the two signals.

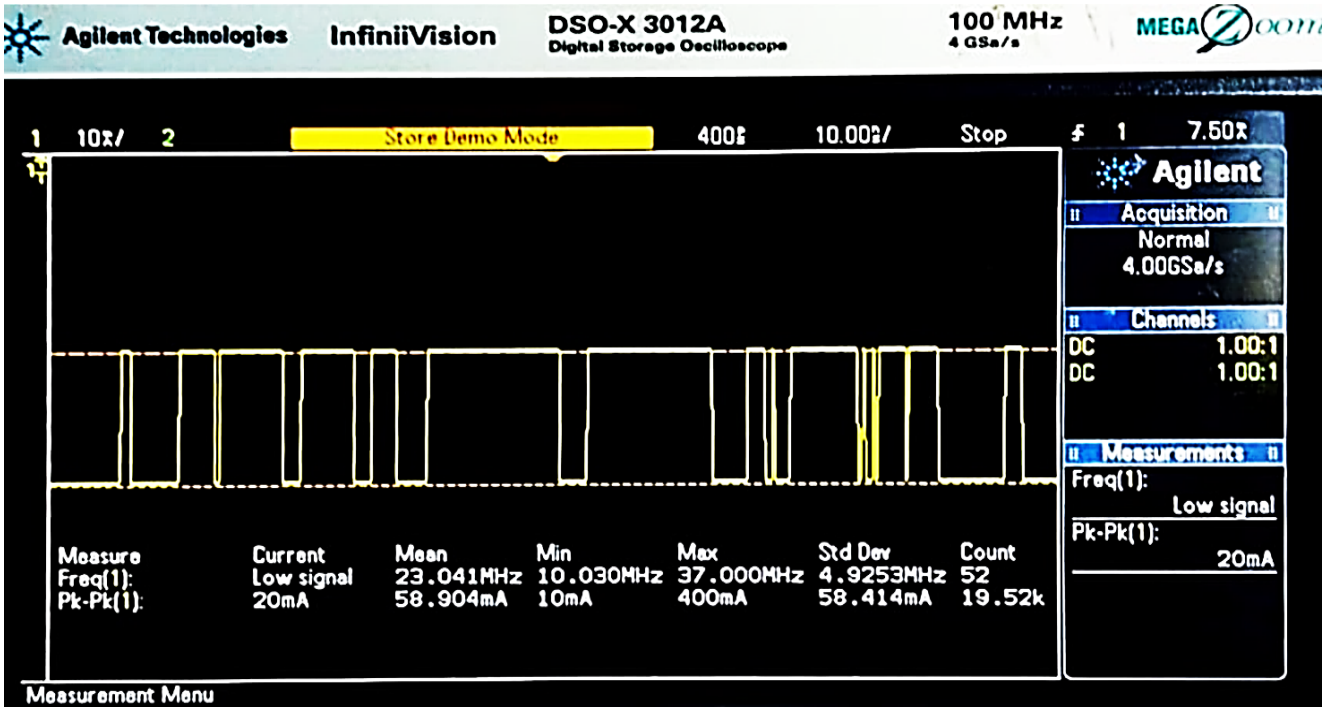


Fig. 4 DSO output waveform used in TRNG based on ADPLL [10]

Figure 4 shows the TRNG based on single ADPLL output patterns obtained by DSO. The TRNG architecture generates random bits saved in a text file using VHSIC Hardware Description Language (VHDL)**.** NIST tests are run on MATLAB software to ensure that the resulting bitstream is unpredictable and stochastic. To check the randomness of the generated output bitstreams, NIST tests are performed on the series of a sequence using MATLAB version R2015a so that it can be used in protecting our information data while transferring wirelessly across the channel. The NIST test evaluation of the project TRNG is shown in Table 1, proving that the output sequence is stochastic.

3.2. Statistical analysis by NIST (National Institute of Standards and Technology's) test

Several analytical tests employing conventional test suites were developed to study the statistical properties of random number generation. Using the NIST SP 800-22 standard quantitative assessment kit [33], probabilistic evaluations of generated TRNG-based cryptographic algorithms are generally done. The results of a performance test on the suggested TRNG generator employed in our proposed wireless architecture are presented in this subsection. Empirical tests are widely utilized to evaluate if a binary series exhibits any of the qualities of a truly random sequence. The measured values of P for the NIST statistical test are summarised in Table 1. As a result, we may conclude that the suggested application with ADPLL-based TRNG meets all NIST requirements and creates keys with appropriate randomness. As a result of these observations, it is obvious that the proposed ADPLL-based TRNG keystream is advantageous in limiting the adverse dynamic deterioration impact induced by the finite accuracy of the digital hardware implementation. A combination with a P-value higher to or similar to 0.001 satisfies the NIST requirement for randomness [34] with a confidence level of 99.9 percent. A total of 150 random bits are used in the test. The NIST findings for the proposed TRNG are listed in the table below, demonstrating that the produced series is truly random data, which improves the data security of wireless communication systems when combined with IR sensor data.

Table 1. NIST Study Results for TRNG based on ADPLL.

NIST Test	P-value	Result (P=pass ,F=Fail)
Frequency	0.0136	P
Block Frequency	0.9134	P
Run	0.000594*	F
Rank	0.9945	P
DFT	0.0136	P
Serial test	0.9936	P
Linear Complexity Test	0.591409	P
Longest run Test	0.0000*	F
Approximate Entropy Test	0.0165	P
Cumulative sum Test	0.9993	P
Random Excursions Test	0.0500	P

4. Implementation of the proposed Transmitter and receiver Architecture

For real-time IR sensor data transmission wirelessly, it contains an FPGA board and an Arduino UNO board with a Bluetooth interface on the transmitter side. The XOR gate (IC 74HCT86) on the receiver side of the Arduino board that interfaces with Bluetooth has one input from the TRNG output and receives encrypted data from the transmitter. Then the data is displayed using the serial monitor of the Arduino IDE (Integrated development environment) and the Arduino Bluetooth controller mobile application. EX-OR'ing carries out the decryption algorithm of the received encrypted data using TRNG bitstreams. The proposed data transmitter and receiver architectures are represented in Figures 5 and 6, respectively, illustrating the proposed system's fundamental components. The FPGA analyses the obtained data and serves as the base platform for data transport. The HC05 integrated Bluetooth Serial Communication Module was used on both the sending and receiving endpoints to establish a Bluetooth connection. The construction commences with installing a Bluetooth interface across two FPGA devices, accompanied by developing the TRNG algorithm encryption and the IR sensor data processing. On the transmitter side, IR sensor data was encrypted using the VHDL programming language, while the TRNG architecture was employed on the receiving side.

4.1. Details of Transmitter architecture

The suggested Transmitter architecture is depicted in Fig. 5. XORing the jitter signal generated by two ring oscillators with the 400MHz ID out signal (DCO output) from the ADPLL and the feedback loop generated by Q1 of DFF1 [10].

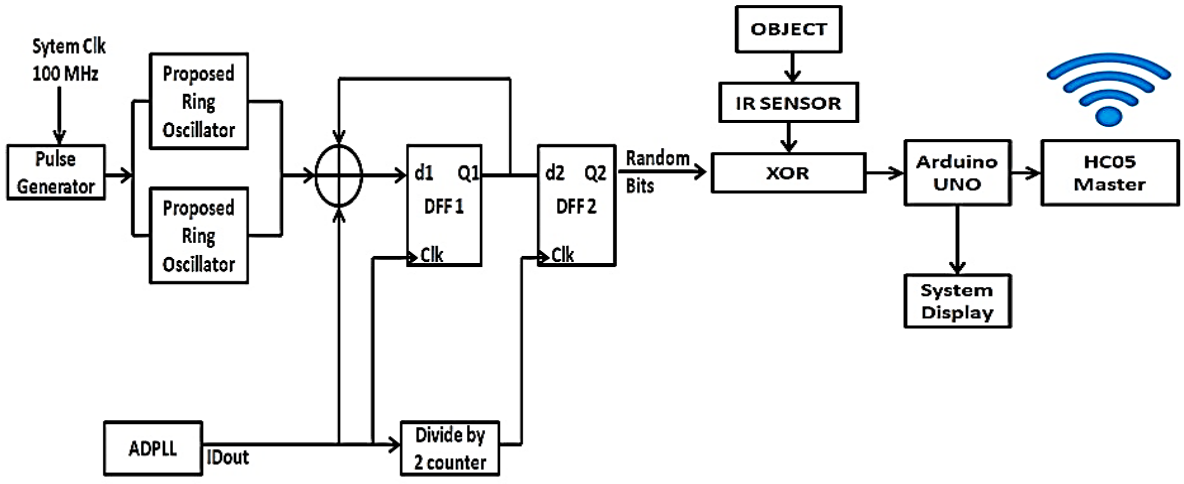


Fig. 5 Transmitter architecture for the proposed wireless communication system

The output of DFF1's Q1 is then passed into DFF2's d2 together with the (Clock) CLK signal created by the counter. Finally, Q2 of DFF2 outputs the created random sequences. The ADPLL circuit architecture [30] utilized a center frequency (fo) of 50 MHz using a modulus factor of k equal to 4. fo is the center frequency, N is identical to eight, and M

is equal to sixteen. In this case, M is a parameter with a usual value of 8,16,32... The ID-clock signal is the DCO CLK pulse equal to 2Nfo [29]. The parameters are displayed in Table 2. After XORing the random bitstream produced by the TRNG with the entropy of the IR sensor data [16], an encrypted data stream is generated that is interfaced with the Arduino UNO for wireless data transfer via the HC05 Bluetooth. As a result, information or data transmitted wirelessly using TRNG encrypted architecture is highly secure.

Table 2. ADPLL Designing Criteria in the Suggested TRNG Framework [10]

ADPLL Design used in TRNG	
Parameter	Design
K	4
M	16
N	8
Centre Frequency (fo)	50MHz

4.2. Details of Receiver Architecture

The objective of the receiver architecture is to receive data transmitted from the sending side successfully. The receiver architecture decrypts the data using a decryption algorithm. This section discusses the hardware setup required to carry out the implemented task and the details of each hardware component required to assemble the entire system in its initial phase. After wirelessly receiving encrypted data in the HC05 secondary module, decryption is performed using an XOR gate (IC 74HCT86) in conjunction with an ADPLL-based TRNG architecture to recover the original data. The receiver architecture, as shown in Fig. 6, consists of four major components: a Bluetooth module HC-05 (slave mode), an Arduino UNO (ATmega 328), and an XOR gate (IC 74HCT86), and a TRNG architecture based on ADPLL for decryption. The receiver successfully receives the data transmitted by the transmitter, and the received data is XORed with TRNG to recover the original data.

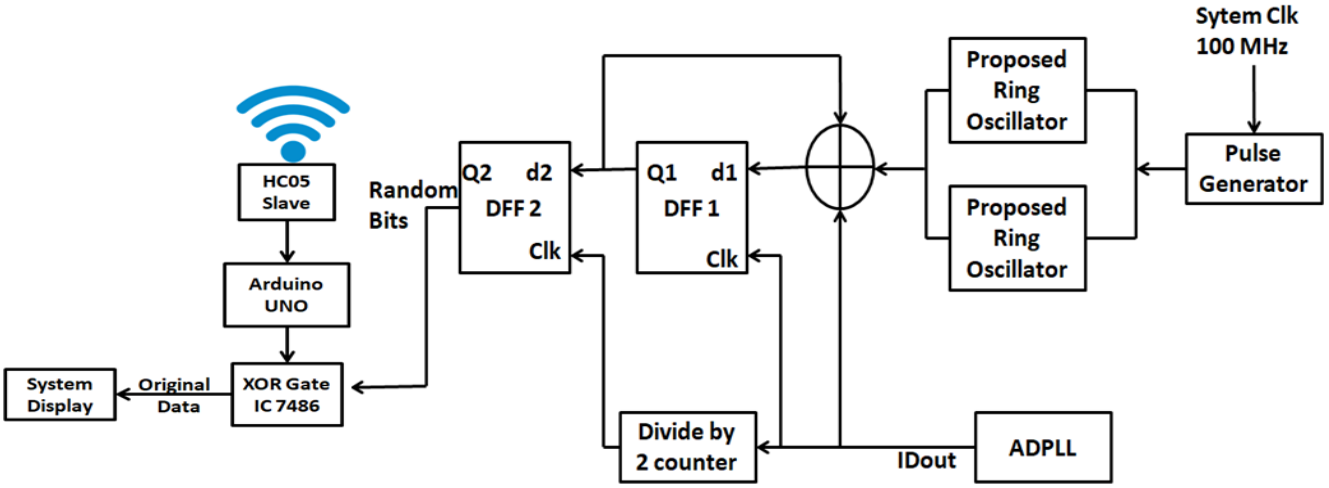


Fig. 6 Receiver architecture for the proposed wireless communication system

5. AT command mode for HC-05 module configuration

The HC-05 Unit is a widely used and widely available Bluetooth module that can be used to enable wireless communication in embedded projects. The module can act as both a master and an enslaved person. Serial communication is used to link the module to a microcontroller. It's essentially a radio transceiver that operates at 3 Mbps in the unlicensed 2.4ghz frequency, commonly recognized as ISM spectral region [35]. To configure the device in Attention command (AT) Mode, as shown in Table 3, the module's key pin must be connected to the microcontroller. It should be switched to low logic first and then to high logic to enter the AT mode. When the main pin is switched to HIGH logic, the module's serial connection baud rate is set to 38400 bits per second even before it is ready to enter the order response work state. The module's preferred baud rate for a startup is 9600 bits per second, determined when the key pin is set HIGH following the module's powering on [36].

Table 3. Commands for HC

Command	Description	Comment
AT	Checking communication	OK
AT+PSWD=XXXX	Set Password	OK
AT+NAME=XXXX	Configure the Bluetooth Name	OK
AT+UART=Baud rate, stop bit, the parity bit	Change Baud rate	OK
AT+VERSION?	Bluetooth response of version number	+Version: XX OK
AT+ORGL	Obtain configuration information from the producer	Parameters: device type, module model, serial parameter, passkey, etc.
AT+ROLE=Param1	Set Role of HC05(1=Master / 0=Slave)	OK
AT+ADDR	Bluetooth Address	OK

6. Experimental results for the proposed TRNG-based wireless communication system

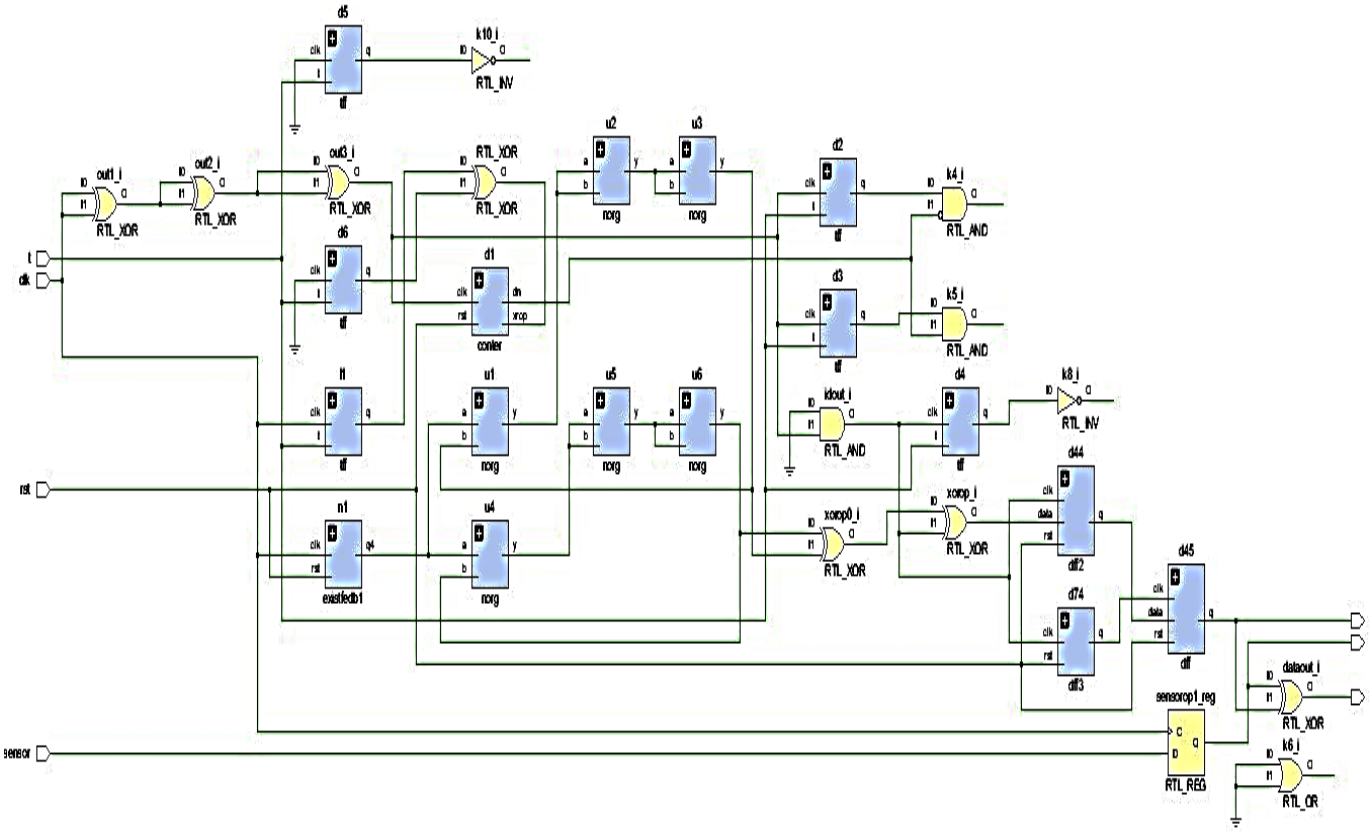


Fig. 7 RTL Schematic diagram for the proposed TRNG used in Transmitter architecture

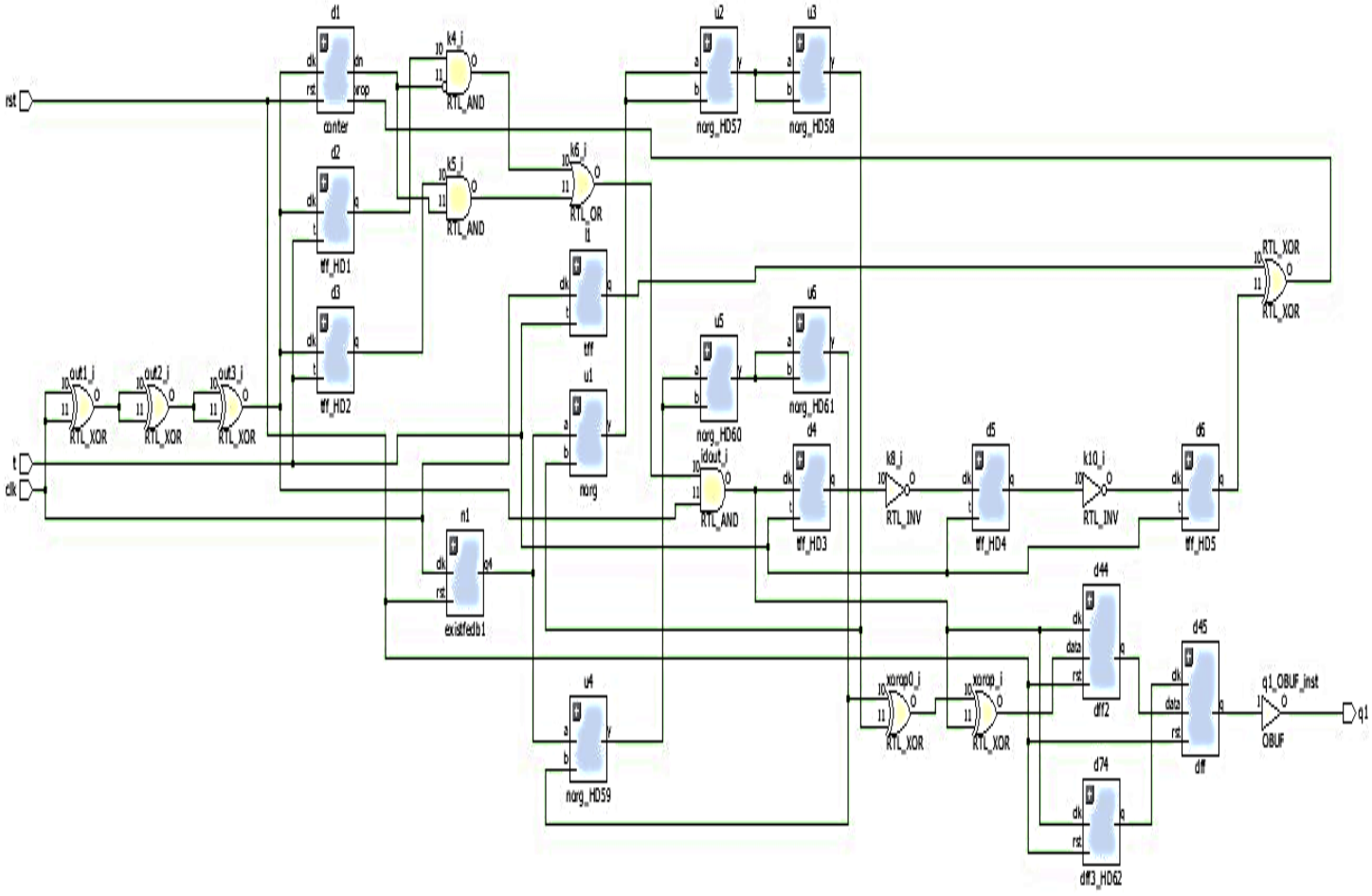


Fig. 8 RTL Schematic diagram for the proposed TRNG used in Receiver architecture

Table 4. Pin details for implemented Transmitter architecture used in the proposed wireless communication system.

Symbol	Details	Mode	FPGA Pin	Arduino UNO Pin	HC05 master	IR Pin
CLK	System Clock	Input	W5			
t	T-FF	Input	V17			
rst	Reset	Input	V16			
q1	Output random bit	Output	A14			
IR-sensor	Sensor input	Input	K17			OUT
xorout	Encrypted data	Output	M18	8		
Tx	UART Transmitter output	Output		RX	TXD	
Rx	UART Receiver input	Input		TX	RXD	
Vcc	Power Supply		USB port	5VDC	Vcc	Vcc

Table 5. Pin details for implemented Receiver architecture used in the proposed wireless communication system.

Symbol	Details	mode	FPGA Pin	Arduino UNO Pin	HC05 master	IR Pin
CLK	System Clock	Input	W5			
t	T-FF	Input	V17			
rst	Reset	Input	V16			
q3	Output random bit	Output	A14			
Tx	UART Transmitter output	Output		RX	TXD	
Rx	UART Receiver input	Input		TX	RXD	
Vcc	Power Supply		USB port	5VDC	Vcc	Vcc

6.1. Encrypted and decrypted output of the proposed Architecture

Using the serial monitor of the Arduino IDE and the Mobile Bluetooth controller software, the outcome of the TRNG encrypted with IR sensor data transfer wirelessly via the HC05 module is shown in Fig. 9, 10, and 11.

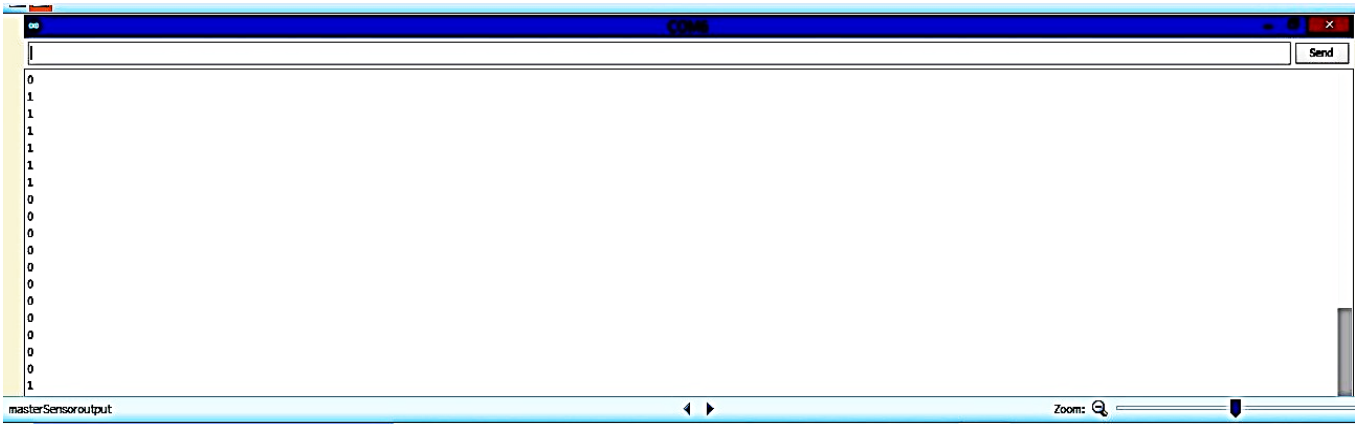


Fig. 9 Transmitted IR sensor data using Master HC05 Bluetooth displayed on Arduino IDE

The output of transmitter architecture is displayed on the serial monitor of Arduino IDE. Here the IR sensor data coming from the object is encrypted using ADPLL-based TRNG and transmitted wirelessly using the HC05 master Bluetooth module.

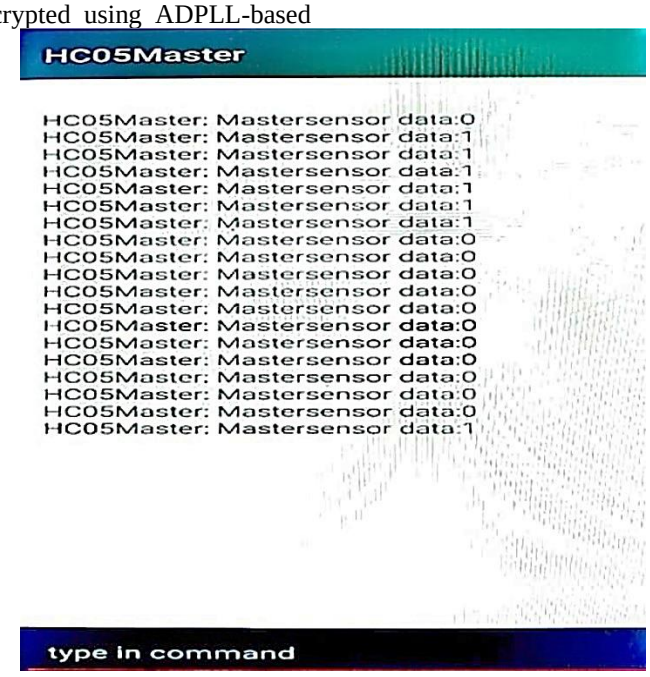


Fig. 10 Transmitter IR sensor data displayed using Arduino Bluetooth Control app

The output of the transmitter displayed of Arduino Bluetooth controller application used in ADPLL based TRNG wireless communication system

Fig. 11 Received IR sensor data using Slave HC05Bluettoth displayed on Arduino IDE

The proposed workstation demonstrates the wireless transmission of real-time encrypted IR sensor data utilizing ADPLL-based TRNG architecture. The complete experimental setup for the proposed TRNG-based wireless communication system is shown in Fig.12. Here breadboard is used to realize the description of data using the XOR gate. Fig 12 shows that the IR sensor detects no objects, producing a logic high output binary value passed as one input to the transmitter side XOR gate circuit and another input from the TRNG output.

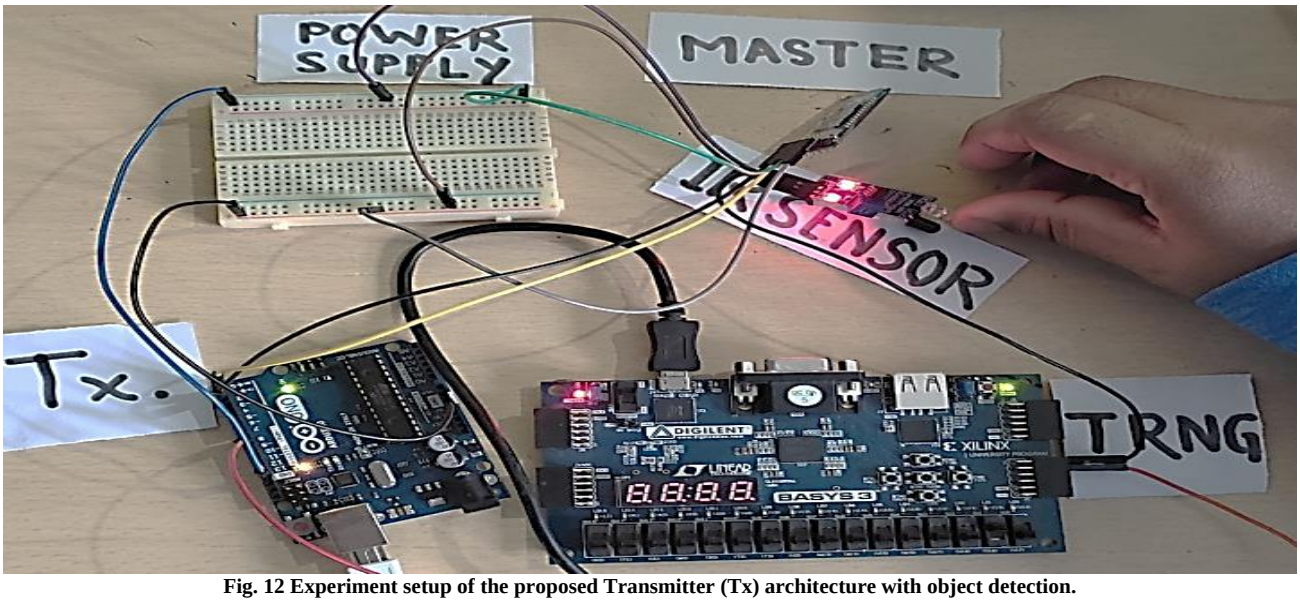


Fig. 12 Experiment setup of the proposed Transmitter (Tx) architecture with object detection.

Two Arduino boards are used for master and agent configuration. Using a master-slave configuration, IR sensor data is transferred securely from the transmitter side to the receiver side wirelessly. Whereas Fig 13 depicts the receiver side of the experiment, where decryption is carried out utilizing an XOR gate (IC 74HCT86) in combination with an ADPLL-based TRNG architecture to obtain the original information.

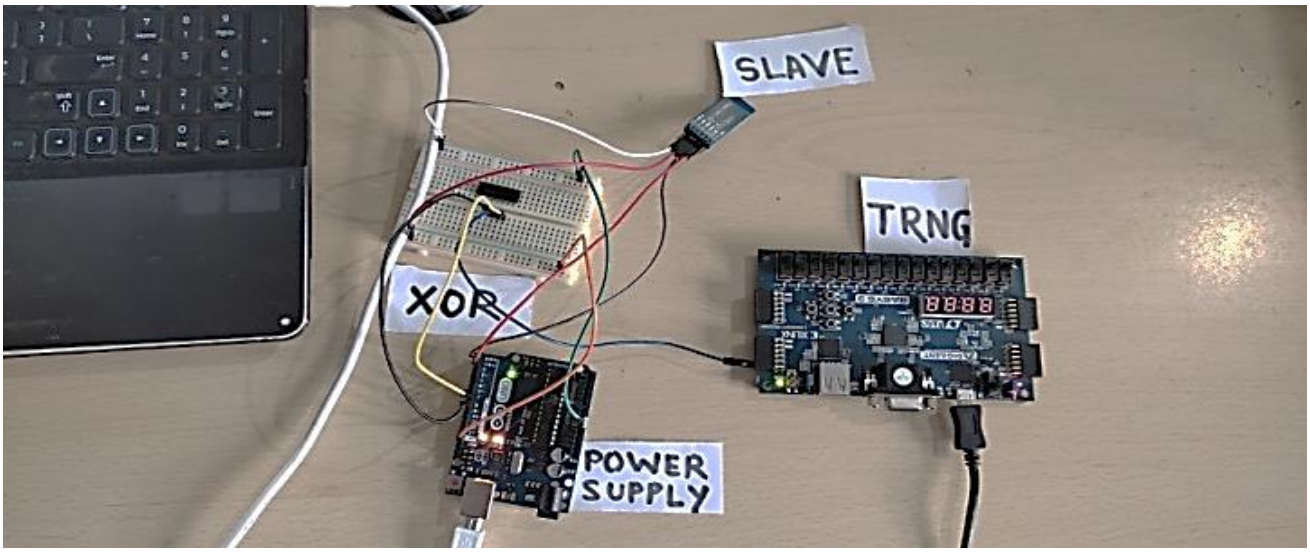


Fig. 13 Experiment setup of the proposed Receiver (Rx) architecture-based wireless communication

6.2. Synthesis results for the proposed architecture

A series of statistical tests are used to evaluate the quality of the proposed design's random number generator. The synthesis report indicates that the designed transmitter architecture has a data path delay of 4.939 ns, and the number of LUTs utilized is 1. In contrast, the receiver architecture has a data route delay of 1.469ns and a resource consumption area of only 2 LUTs. Table 6 contains the synthesis outcomes for the proposed wireless architecture using TRNG.

Table 6. Synthesis Outcomes of the Built TRNG Architecture for Wireless Design

Parameter	Proposed design	
	Transmitter (Tx)	Receiver (Rx)
Total On-Chip Power (W)	0.074 W	0.076 W [10]
Data path Delay (ns)	4.939 ns	1.469 ns[10]
Area (Number of 4 input LUTs)	1	2 [10]

7. Comparison with the Existing design

Table 7 analyses the efficiency of several TRNGs, whereas Table 8 compares numerous TRNG topologies from the synthesized report. The approach we recommended makes greater use of the hardware resources available. Furthermore, Tx architecture consumes 7.4 mW, and Rx architecture consumes 7.6 mW, much less than other literature. Table 8 shows that although using fewer hardware resources and consuming less power, the architecture's overall output bitrate is unaffected, with a maximum of 202.47 Mbps for the Transmission system and 680.73 Mbps for Reciever systems.

Table 7. Comparative analysis of various designs

Citation	Entropy	Device	Hardware resource	Post-processing
Proposed Tx for implemented application	Metastability/Jitter	Xilinx Artrix-7 FPGA	1 LUT	Not done
Proposed Rx for implemented application	Metastability/Jitter	Xilinx Artrix-7 FPGA	2 LUTs	Not done
[7]	Metastability/Jitter	Xilinx XCKU040	1 PLL 5 Primitives	Done

			5 Slices	
[23]	ADPLL jitter/Metastability	Xilinx XC7A35T-CPG236-1	1 LUT	Done
[37]	Metastability	Altera Cyclone III	511 LUTs	Done
[38]	Ring oscillator	Xilinx XC5VLX50T	147 LUTs	Done
[39]	RS-Latch	Xilinx XC4VFX20	580 Slices	Not done
[40]	Chaotic ring oscillator	Xilinx XC6SLX16	44 LUTs	Done
[41]	PLL jitter	Altera Stratix	120 LE	Done
[42]	Metastability	Virtex-5	n/a LUT 64 Latches	Not done
[43]	Free running ring oscillator	Xilinx Spartan-6 FPGA	10 LUTs and 5 FFs	Not done
[10]	Metastability/Jitter	Xilinx XC7A35T	2 LUTs and 1 Slice	Not done

Table 8. Assessing the synthesis outcomes of several TRNG architectures.

Citation	Area (millimeter^2)	Power	Speed	Post processing	Testing	Platform
Proposed Tx for implemented application	Not Applicable	7.4 mW	202.47 Mbit/s	Not done	NIST SP800-22	Field Programmable Gate Arrays
Proposed Rx for implemented application	Not Applicable	7.6 mW	680.73 Mbit/s	Not done	NIST SP800-22	Field Programmable Gate Arrays
[23]	Not Applicable	7.2 mW	200 Mbit/s	XOR	NIST SP800-22	Field Programmable Gate Arrays
[44]	0.02	0.8 mW	25 Mbit/s	Not Done	FIPS 140-2	0.09 μm CMOS
[45]	Not Applicable	Not Applicable	4.59 Mbit/s	XOR	FIPS 140-1 and NIST SP800-22	Field Programmable Gate Arrays
[46]	0.037689	1.32 mW	50 Mbit/s	Von Neumann	NIST SP800-22 and TestU01	0.18 μm CMOS
[47]	Not Applicable	Not Applicable	447.83 Mbit/s	XOR and32-bit addition	NIST SP 800-22	CPU
[48]	Not Applicable	125 mW	1.5 Mbit/s	Not Done	NIST SP800-22	Field Programmable Gate Arrays
[49]	Not Applicable	Not Applicable	2.02 Mbit/s	6 bit LFSR	FIPS 140-1	0.18 μm CMOS
[50]	93.1	1.0967 mW	127 Mbit/s	XOR	NIST	0.45 μm CMOS
[10]	Not Applicable	7.6 mW	680.7 Mbit/s	Not Done	NIST SP800-22	Field Programmable Gate Arrays
[51]	0.057	26.1 mW	300 Mbit/s	XOR	NIST	0.35 μm CMOS

8. Discussion

[7] talks about a TRNG that uses PLL and several FPGA primitives. Jitter from PLLs and FF metastability from FFs is the main sources of entropy. According to the designer, the proposed design used less electricity and occupied nearly minimal storage while providing a throughput of 100Mbps. PLL, on the other hand, used more energy and occupied a larger area than ADPLL. Hence ADPLL-based designs are significantly preferable when constructing a dependable TRNG. According to the study, ring oscillators are utilized to produce jitters. An extra supply of unpredictability, ASR, is employed to enhance the output sequences' productivity, considerably improving the TRNG's unpredictability. In [40], a chaos RO-based TRNG with a throughput of 125 Mbps can be constructed as simple yet robust. Compared to the suggested TRNG designs, [43] employs more LUTs and produces a minimum throughput. [41] A typical digital 0.018 m n-well Complementary Metal Oxide Semiconductor (CMOS) technology was used to construct a prototype model with a throughput of 10 Mbps. PLL has been employed in TRNG for a long period; current research has demonstrated that a digitized form of ADPLL with discrete-time ring oscillators can also be used in TRNG for secure real-time wireless transfer of data using short-range Bluetooth technology. As shown in Table 8, TRNG is based on a typical chaotic oscillator as the generator of unpredictability, similar to Chua's circuits [49], Jerk circuitry [46], and no-equilibrium chaotic network [45], taking into consideration chaos' lower frequency and narrowband. Analyzing the overall output sequence of all TRNGs dependent on discrete-time chaos reveals that [47] is substantially quicker than various chaotic output bits due to its computation. Compared to the TRNG space of the same CMOS technology in [51], [50] has the largest area. The authors developed circuits with an eight-stage pipeline ADC and a two-stage pipeline ADC, which explains why this is the case. ADC is often employed to create chaotic sequences with randomized values; it is faster but requires a lot of storage space.

9. Conclusion and Future work

This research establishes a realistic implementation of TRNG-based IR-sensor data transmissions capable of securing real-time data transmissions via HC05 wireless communication networks. We have used FPGA, Arduino UNO, an IR sensor, HC05 Bluetooth modules, and an XOR gate (IC 74HCT86) to assemble a wireless communication system. More specifically, the proposed wireless transfer technique for establishing secure data encryption strengthens its resistance to several attacks, including probabilistic and key extraction attacks. It protects real-time applications from latent security vulnerabilities. A more secure and dependable TRNG is produced by harnessing the seeds of randomness from the All digital phased locked loop, ring oscillator, and flip-flop, which is employed in the encryption and decryption method of our proposed wireless system. The output of the transferred data is captured using the Arduino IDE Serial Monitor along with the Arduino Bluetooth Control mobile application. The design and implementation were done using Vivado v.2015 on a Xilinx FPGA. Upcoming studies will examine if this approach applies to various TRNG architectures with wifi-based applications that require considerable engineering complexity and how this entropy source behaves when subjected to active manipulation attacks. With the findings of this study, the wireless application potential of ADPLL-based TRNG as encryption algorithms is optimistic, making it the most efficient and secure alternative for a wide variety of activities like cyber defense, finance, and the Internet of Things (IoT)

Acknowledgment

The author wishes to extend his deepest gratitude to Dr. Manoj Kumar for his continuous support and counsel throughout the writing process.

References

[1] Eastlake, D., Crocker, S., Schiller, J, “Randomness Recommendations for Security - RFC 1750,” 1994. Available At: Http://Www.Faqs.Org.

[2] Goldberg, I., Wagner, D,” Randomness and The Netscape Browser,” *Dr. Dobb's Journal*, 1996.

[3] Buchmann, J., Dahmen, E., & Szydlo, M, “ Hash-Based Digitalsignature Schemes,” *Post-Quantum Cryptography*, Pp.35–93, 2009. Https://Doi.Org/10.1007/978-3-540-88702-7_3

[4] Loza, S., & Matuszewski, L, ” A True Random Number Generator Using Ring Oscillators and SHA-256 As Post-Processing, *2014 International Conference on Signals and Electronic Systems (ICSES)*, 2014. Https://Doi.Org/10.1109/Icses.2014.6948739

[5] L. Zhou, F. Tan, and F. Yu, "A Robust Synchronization-Based Chaotic Secure Communication Scheme with Double-Layered and Multiple Hybrid Networks," *IEEE Systems Journal*, Pp. 1–12, 2019.

[6] T Yang, “A Survey of Chaotic Secure Communication Systems,” *Int. J. Comput. Cogn*, vol.**2**, no.2, Pp.81–130, 2004.

[7] Guido Di Patrizio Stanchieri, Andrea De Marcellis, Elia Palange, Marco Faccio, “A True Random Number Generator Architecture Based on A Reduced Number of FPGA Primitives,” *AEU - International Journal of Electronics and Communications*, vol.105, Pp.15-23,2019. ISSN 1434-8411.

[8] Crocetti, L., Di Matteo, S., Nannipieri, P., Fanucci, L., & Saponara, S, “ Design and Test of an Integrated Random Number Generator with All-Digital Entropy Source,” *Entropy*, vol.24, no.2, Pp.139, 2022.Https://Doi.Org/10.3390/E24020139

[9] Tobin, P., Tobin, L., Mckeever, M., & Blackledge, J, " On The Development of A One-Time Pad Generator for Personalizing Cloud Security," *Conference Papers*, 2017.Https://Doi.Org/10.21427/D7VF9V

[10] Huirem Bharat Meitei & Manoj Kumar, "FPGA Implementation of True Random Number Generator Architecture Using All-Digital Phase-Locked Loop," *IETE Journal of Research*, 2021.DOI: 10.1080/03772063.2021.1963333

[11] Bluetooth Committee, Specifications of The Bluetooth System (Core), 1999.

[12] Radhapuram, S., Yoshihara, T., & Matsuoka, T, "Design and Emulation of All-Digital Phase-Locked Loop on FPGA," *Electronics*, vol.*8*, no.11, Pp.1307, 2019. Https://Doi.Org/10.3390/Electronics8111307

[13] MD, R., & SM, T, *"FPGA* Secured Wireless Communication Using Aes," *Ijera.Com*, 2013. Https://Www.Ijera.Com/Papers/Vol3_Issue4/NP3424042407.Pdf.

[14] Rai, S., & Thakare, A, " Implementation of Reliablewireless Real-Time Automation System Based on Android Mobile Phone and FPGA," *Ijsr.Net*, 2013.Https://Www.Ijsr.Net/Archive/V4i1/SUB15189

[15] J. G. Pandey, T. Goel and A. Karmakar, "Hardware Architectures for PRESENT Block Cipher and Their FPGA Implementations," *IET Circuits, Devices & Systems*, vol. 13, no. 7, Pp. 958–969, 2019.

[16] Taha, H., Sazish, A., Ahmad, A., Sharif, M., & Amira, A, "Efficient FPGA Implementation of A Wireless Communication System Using Bluetooth Connectivity," *Ieeexplore.Ieee.Org*, 2010. Retrieved 22 January 2022, From Https://Ieeexplore.Ieee.Org/Document/5537610/.

[17] A. Cherkaoui, V. Fischer, L. Fesquet, and A. Aubert, "A Very High Speed True Random Number Generator with Entropy Assessment," *in Proceedings of International Workshop on Cryptographic Hardware and Embedded Systems* (CHES'13), *Ser. Lecture Notes in Computer Science*, vol. 8086. Springer-Verlag, Pp. 179–196, 2013.

[18] Sadoudi, S., Tanougast, C., Azzaz, M., & Dandache, A," Design and FPGA Implementation of A Wireless Hyperchaotic Communication System for Secure Real-Time Image Transmission," *EURASIP Journal on Image and Video Processing*, vol.2013, no.1, 2013. Https://Doi.Org/10.1186/1687-5281-2013-43

[19] Wallace, K., Moran, K., Novak, E., Zhou, G., & Sun, K, "Sensor-Based Random Number Generation for Mobile and Iot Devices," *IEEE Internet of Things Journal*, vol.*3*, no.6, Pp.1189-1201, 2016. Https://Doi.Org/10.1109/Jiot.2016.2572638

[20] Koyuncu, İ., & Turan Özcerit, A, " The Design and Realization of A New High-Speed FPGA-Based Chaotic True Random Number Generator," *Computers & Electrical Engineering*, vol.*58*, Pp.203-214, 2017.Https://Doi.Org/10.1016/J.Compeleceng.2016.07.005

[21] Vasyltsov, I., Hambardzumyan, E., Kim, Y.-S., & Karpinskyy, B. (N.D.)," Fast Digital TRNG Based on Metastable Ring Oscillator," *Cryptographic Hardware and Embedded Systems – CHES 2008*, Pp.164–180, 2008. Https://Doi.Org/10.1007/978-3-540-85053-3_11

[22] B. Sunar, W. Martin, and D. Stinson, "A Provably Secure True Random Number Generator with Built-in Tolerance To Active Attacks". in: *IEEE Transactions on Computers*, vol.56, no.1, Pp. 109–119, 2007. ISSN: 0018-9340. Doi: 10.1109/TC.2007. 250627.

[23] Meitei, H., Kumar, M," FPGA Implantations of TRNG Architecture Using ADPLL Based on FIR Filter As A Loop Filter," *SN Appl. Sci*, vol.4, no.96, 2022. Https://Doi.Org/10.1007/S42452-022-04981-6

[24] Buchmann, J., Dahmen, E., & Szydlo, M, " Hash-Based Digital Signature Schemes," *Post-Quantum Cryptography*, Pp.35–93, 2009. Https://Doi.O Rg/10.1007/978-3-540-88702-7_3

[25] Usermanual.Wiki, "HC Sries Product Manual 201104 Hc-05-User-Instructions-Bluetooth," 2022. <Https://Usermanual.Wiki/Pdf/Hchc05userinstructionsbluetooth201.191890306/Html

[26] IR Sensor: Circuit, Types, Working Principle & Its Applications. Watelectronics.Com. Https://Www.Watelectronics.Com/Ir-Sensor/.

[27] 7486 Technical Data. Futurlec.Com. Https://Www.Futurlec.Com/74/IC7486.Shtml.

[28] Internet Resource, "AMD Random Number Generator Library." <Https://Developer.Amd.Com/Wordpress/Media/2013/12/AMD-Random-Number-Generator- User-Guide.Pdf> [Accessed September 2018].

[29] Chaudhary, A. K., & Kumar, M, " Design and Implementation of ADPLL for Digital Communication Applications," *IEEE Xplore*, 2017. Https://Doi.Org/10.1109/I2CT.2017.8226159

[30] Lata, K., & Kumar, M, " ADPLL Design and Implementation on FPGA," *IEEE Xplore*, 2013. Https://Doi.Org/10.1109/ISSP.2013.6526917

[31] Sandeep Vallabhaneni and Sanjay Attri," Design of an All-Digital Pllcore on FPGA," *Sci /Engr. SF, AISG/AISD, IISU, ISRO India.*

[32] Lata, Kusum, and Manoj Kumar, "ALL Digital Phase-Locked Loop (ADPLL): A Survey," *International Journal of Future Computer and Communication*, Pp. 551–554, 2013. 10.7763/Ijfcc.2013.V2.225.

[33] A Rukhin, J Soto, J Nechvatal, M Smid, E Barker, S Leigh, M Levenson, M Vangel, D Banks, A Heckert, J Dray, S Vo, "A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications," *Technical Report, NIST Spec. Publication 800-22 Revision 1a*, NIST, Gaithersburg, 2010

[34] Bassham, L., Rukhin, A., Soto, J., Nechvatal, J., Smid, M., Barker, E., Leigh, S., Levenson, M., Vangel, M., Banks, D., Heckert, N., & Dray, J, "A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications," *Csrc.Nist.Gov*,2010.

[35] R. Shorey and BA. Miller, "The Bluetooth Technology: Merits and Limitations, in Personal Wireless Communications*," 2000 IEEE International Conference on*, Pp. 80–84, 2000.

[36] *Bluetooth Module HC-05 Sensors & in Modules*. Electronicwings.Com Https://Www.Electronicwings.Com/Sensors Modules/Bluetooth-Module- Hc

[37] Jun B, Kocher P (1999) Intel random number generator. in: rambus; cryptography research, inc, white paper prepared for intel corporation, https:// www.r ambus. com/ intel- random- numbergenerator/

[38] Jessa M, Matuszewski L (2011) Enhancing the randomness of a combined true random number generator based on the ring oscillator sampling method. IEEE. https:// doi. org/ 10. 1109/ ReCon Fig. 2011. 35

[39] Hata H, Ichikawa S (2012) FPGA implementation of metastability-based true random number generator. IEICE Trans Inf Syst 95(2):426–436. https:// doi. org/ 10. 1587/ trans inf. e95.d. 426

[40] Yang Y, et al (2017) A reliable true random number generator based on novel chaotic ring oscillator. in: 2017 IEEE international symposium on circuits and systems (ISCAS) www. seman ticscholar. org/ paper/A- relia ble- true- random-number- gener atorbased-on- Yang- Jia/c3e65 e27fd 09968 93497 7d250 f0ead 2c13e 60b35, https:// doi. org/ 10. 1109/ISCAS. 2017. 80508 43

[41] Fischer V, Drutarovsky M, Šimka M, Bochard N (2004) High performance true random number generator in altera stratix FPLDs. Field Progr Log Appl. https:// doi. org/ 10. 1007/ 978-3- 540- 30117-2_ 57

[42] Ben-Romdhane M, Graba T, Danger J-L (2013) Stochastic model of a metastability-based true random number generator. Trust Comput. https:// doi. org/ 10. 1007/ 978-3- 642- 38908-5_7

[43] Yang B, Rožic V, Grujic M, Mentens N, Verbauwhede I (2018) ESTRNG:a high-throughput, low-area true random number generator based on edge sampling. IACR Trans Cryptogr Hardw Embed Syst. https:// doi. org/ 10. 13154/ tches. v2018. i3. 267- 292

[44] X. Zhang and C. Wang, "A novel multi-attractor period multi scroll chaotic integrated circuit based on CMOS wide adjustable CCCII," *IEEE Access*, vol. 7, pp. 16336–16350, 2019.

[45] A. Akgul, H. Calgan, I. Koyuncu, I. Pehlivan, and A. Istanbullu, "Chaos-based engineering applications with a 3D chaotic the system without equilibrium points," *Nonlinear Dynamics*, vol. 84, no. 2, pp. 481–495, 2016.

[46] C. Wannaboon, M. Tachibana, and W. San-Um, "A 0.18- CMOS high-data-rate true random bit generator through modulation of chaotic jerk circuit signals," *Chaos: an Interdisciplinary Journal of Nonlinear Science*, vol. 28, no. 6, p. 063126, 2018.

[47] J. S. Teh, A. Samsudin, M. Al-Mazrooie, and A. Akhavan, "GPUs and chaos: a new true random number generator," *Nonlinear Dynamics*, vol. 82, no. 4, pp. 1913–1922, 2015.

[48] I. Cicek, A. E. Pusane, and G. Dundar, "A new dual entropy core true random number generator," *Analog Integrated Circuits and Signal Processing*, vol. 81, no. 1, pp. 61–70, 2014.

[49] H. Moqadasi and M. B. Ghaznavi-Ghoushchi, "A new Chua's circuit with monolithic Chua's diode and its use for efficient true random number generation in CMOS 180 nm," *Analog Integrated Circuits and Signal Processing*, vol. 82, no. 3, pp. 719–731, 2015.

[50] S. Ergun and S. Ozoguz, "Truly random number generators based on non-autonomous continuous-time chaos," *International Journal of Circuit theory and Applications*, vol. 38, no. 1, pp. 1–24, 2010.

[51] M. Park, J. C. Rodgers, and D. P. Lathrop, "True random number generation using CMOS Boolean chaotic oscillator," *Microelectronic Journal*, vol. 46, no. 12, pp. 1364–1370, 2015.

[52] Rai, S., & Thakare, A. (2015). Implementation of ReliableWireless Real-Time Automation System Based on Android Mobile Phone and FPGA. Retrieved May 4, 2022, from https://www.ijsr.net/archive/v4i1/SUB15189.pdf.