International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18(3s) 2026 pp. 411-424
https://doi.org/10.70917/ijcisim-2026-2339

Article

Experimental Analysis and implementation of Encryption in IEEE 802.11 Using GNU Radios

Bhavini K. Kumawat[1,2], Rajesh M. Bodade[1,2], Gaurav Sharma[1]

[1] Military College of Telecommunication Engineering, Mhow, Madhya Pradesh, India
[2] Devi Ahilya Vishwa Vidyalaya, Indore, Madhya Pradesh, India
E-mail: bhavini.kumawat@gmail.com, rajeshbodade@gmail.com, gauravkh07@gmail.com
Corresponding Author: Bhavini K. Kumawat (E-mail: bhavini.kumawat@gmail.com)

Abstract: — Reliable and reproducible encryption mechanisms are essential in modern wireless communication systems, particularly when implementing IEEE 802.11 over software-defined radio (SDR) platforms. This paper presents an experimental framework that integrates Advanced Encryption Standard in Cipher Block Chaining mode (AES–CBC) with PKCS#7 padding into a GNU Radio based IEEE 802.11 transceiver using USRP B205mini-i hardware. The system operates over 2.412 GHz and 5.89 GHz bands and supports multiple modulation and coding schemes (MCS), including BPSK, QPSK, 16QAM and 64QAM, enabling a comparative study of link robustness under low and high order modulations. Packet Reception Ratio (PRR) is formally defined and employed as the primary reliability metric, with PRR measured as a function of transmit gain and modulation order under controlled indoor conditions. A complete reference implementation of AES–CBC is provided, together with validation scripts, known-answer tests and interoperability checks against OpenSSL to ensure cryptographic correctness. Experimental results show that low-order modulation, particularly BPSK with rate 1/2, achieves the most robust PRR performance, while higher order modulations exhibit sharper degradation with reduced link margin. The combined SDR–encryption testbed, along with shared code, test vectors and flowgraphs, offers a reproducible reference for future research on secure IEEE 802.11 implementations and can be extended to alternative cipher modes and adaptive link adaptation schemes.
Keywords: — IEEE 802.11 Wi-Fi, Encryption, Data integrity, SDR, GNU Radio.

1. INTRODUCTION

The increasing reliance on wireless communication and software-defined radio platforms has heightened the need for robust encryption and validation frameworks. AES–CBC remains a widely adopted block cipher mode due to its balance of security and efficiency. However, reproducibility and validation of cryptographic implementations are often overlooked in academic publications. This work addresses that gap by presenting a complete reference implementation, validation plan and reproducible test vectors. In addition, performance metrics such as Packet Reception Ratio (PRR) are introduced to quantify link reliability. By integrating mathematical definitions, algorithmic descriptions and reproducible scripts, the paper provides a comprehensive framework for both theoretical analysis and practical verification.

The novelty of the present work lies in the integration of cryptographic functionality, SDR-based IEEE 802.11 implementation and reproducible experimental validation within a single framework. Unlike prior studies that primarily focus on PHY/MAC realization, adaptive modulation, or SDR feasibility, the proposed work combines AES-CBC insertion, packet-level validation, and modulation-dependent PRR analysis over two operating bands using a practical GNU Radio testbed. The contribution is therefore not limited to encryption implementation alone, but extends to a reproducible secure communication framework that can be independently verified and extended for future SDR security research.

2. RELATED WORK AND BACKGROUND

Modern Wi-Fi systems use IEEE802.11 Protocol structure with different modulation configurations in hardware architecture and it is bounded to only one modulation technique. These types of configurations can lead to common behaviour in all environmental conditions which can be affected by any random extraneous factor. Hence the performance can be degraded in noisy environment leads to Low SNR, Low throughput, worse BER and many more conditions. The solution to this problem can be overcome by changing Modulation technique with different modulation coding scheme (MCS) either manually or automatically by sensing the situation to keep communication flawless. But, as the Protocol is structured in hardware, this solution is not possible with conventional Wi-Fi instruments. To overcome this issue, software-defined radio (SDR) platforms play a major role with its dynamic FPGA configuration controlled by Software. [1] Software defined radios are readily available with broad spectrum coverage up to GHz bands. This quality of SDR makes the instrument available to implement all modern communication technologies including cellular communication up to 5G, LTE, LoRA, RADAR and all IEEE wireless protocol including IEEE802.11. [2] Even though, a single hardware is well enough capable to implement all modern wireless communication technology just by software programming in FPGA. [3] Also, Implementation of IEEE protocol can be developed layer by layer for real time communication with other devices for real time evaluation and observations.[4]

Recently and in past years, worldwide researchers have been developing and evolving IEEE802.11 protocol on SDR platform using GNU radio including PHY and MAC layers. [5] Precise adaptive receiver to adopt and detect all modulation techniques has also been developed separately to meet the performance with GNU developed transmitter and real time Wi-Fi devices to gather the data.[6] Even, a single SDR is also now capable to behaves as Transmitter and receiver as well with high-resolution real-time video transmission for Drone and Radar communication setups. [7][8][9] Conventional Wireless instruments are not capable to integrate all these applications in one instrument yields need of SDR on high for modern applications. For software handling, GNU Radios is one of the most dominant software mediators between for real time application implementations.[10][11][12]

3. RESEARCH GAP IDENTIFICATION

As the integrity of the data is one of the most important parameters in wireless communication in user application, encryption is required seamlessly to prevent MITM attacks including spoofing, sniffing, malware and malicious frame injection even in encrypted scenario.[13]

Still these attacks are implemented on a particular frequency band with constant modulation techniques and latest Wi-fi Protected Access have some limitations a different approach is must to secure data in the air.[14]

4. RESEARCH METHODOLOGY

To improve security in Wi-fi, one more approach than Wireless Protected Access is that changing in modulation scheme with coding scheme. This provides an additional degree of freedom to match the user's throughput and coverage requirements. Following Modulation schemes have their own advantages and disadvantages. We are dividing users into two main categories: 1. Speed Requirement 2. Coverage area.

With this, we can understand that Lower order modulation techniques give large area coverage but speed is comparatively slow but good Signal to Noise Ratio and Vice-versa for Higher order Modulation techniques. These two major contexts covered most of the Sensus and hence required a dynamic and adaptive approach toward user application zone. The work is already approached and implemented by other researchers in past, but by changing the power and hardware requirements result may be vary and the work is done for unencrypted format. We introduce other hardware specification of SDR and will add AES encryption to protect data which is not currently covered.

5. PRACTICAL IMPLEMENTATION

We are using the IEEE 802.11p functional GNU Radio block developed by Bastian Bloessl [5,6] for vehicular network where data is up and down converted using Orthogonal Frequency Division Modulation Technology with SDR USRPN210 from Ettus Research. We are altering the hardware and f using following modulation techniques to evaluate the performance differences for low and high order modulation techniques with different code rate in constant conditions.

The following frequency bands are selected for analysis, reflecting the operational spectrum of contemporary IEEE 802.11 standards:

For performance evaluation, the transceiver was configured at two operating center frequencies, namely 2.412 GHz and 5.89 GHz, representing the 2.4 GHz WLAN band and the higher-frequency band used for comparative assessment in the proposed SDR framework. The experiments were carried out on the USRP B205mini-i platform in transceiver mode, as illustrated in Figure 1. The B205mini-i is a single channel, fullduplex SDR platform based on the Analog Devices AD9364 RF trans receiver, supporting operation from 70 MHz to 6 GHz with up to 56 MHz instantaneous bandwidth. Throughout the manuscript, all comparative performance results, PRR plots and confidence-interval figures correspond to measurements obtained only at 2.412 GHz and 5.89 GHz. Accordingly, these two frequencies should be considered the definitive operating points for all reported experiments. The device was interfaced with a host computer via USB 3.0 and controlled using the GNU Radio framework.

GNU Radios is used as software mediator for physical layer and MAC Layer interaction with Hardware SDR. Message Data Units were transmitted with the rate of 10 Packets/Second and getting converted into MAC Data by Various protocol data including adding Source, Destination and Base Station Addresses. MAC data is then up-converted and transmitted over the air through the PHY layer at center frequencies of 2.412 GHz and 5.89 GHz using low- and high-order modulation techniques for comparative performance analysis. The system bandwidth and symbol rate were configured to maintain consistent spectral occupancy across both frequency bands. Identical baseband parameters were preserved to ensure fair comparison.

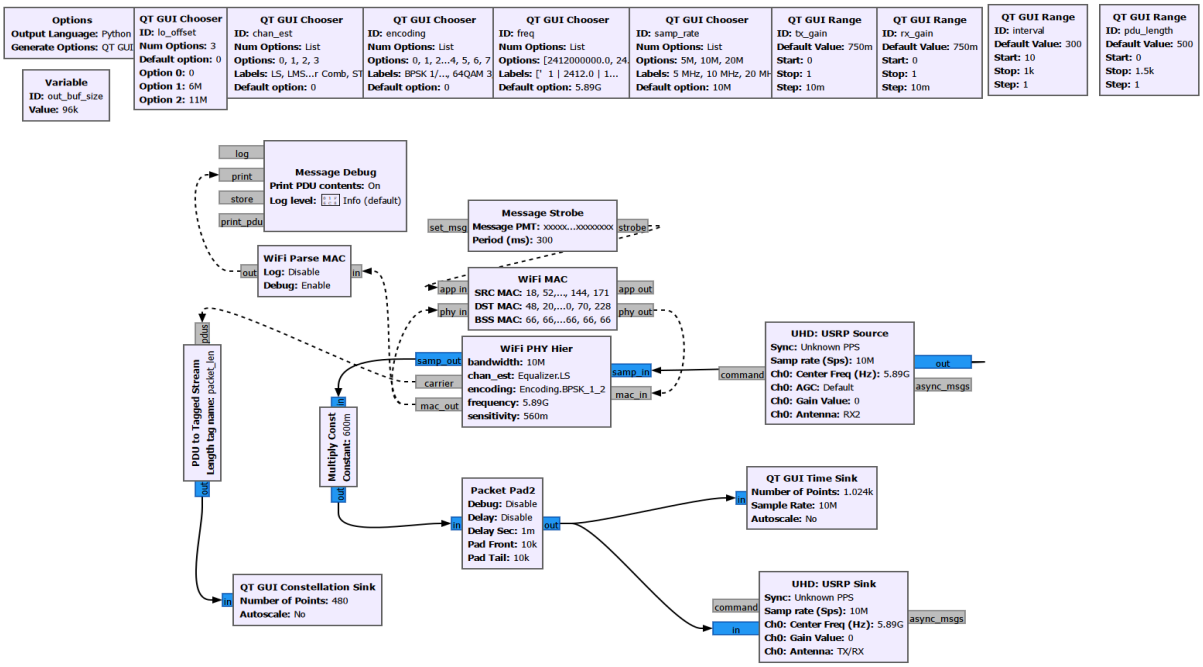

Figure 1. GNU Radio Flow Graph for IEEE802.11 trans-receiver.

This transceiver consists of separate transmitter and receiver blocks based on Orthogonal Frequency Division Multiplexing, as shown in Figure 1. It builds on the work of Fuxjäger , which presented one of the earliest IEEE 802.11 transmitter implementations on GNU Radio. WPC-updated-1.2-Experimental-Analysis-and-implementation-1.docx. The current work improves latency handling and enables reliable reception across adjacent frequency bands. After the management frame is added and the data is converted into a MAC frame, it is prepared for PHY-layer processing through FFT, OFDM subcarrier division, scrambling, pilot insertion, and cyclic prefix insertion.

WiFi PHY block is Hierarchal block which is dividing the frame into OFDM spectral division and up conversion modulation including the selection of modulation technique. Packet padding is done to as inter-leaver between two successive frames to avoid collision. It will be using for both transmission and reception. From “Sampin” , USRP Source can be connected for reception of the data and “Sampout” will be used to transmit the data. Figure 3 is showing the detailed flowgraph. WiFi MAC is converting feed data into MAC frame by adding Frame control and duration frame of 2 Bytes each, 6 Bytes each for Source, Destination and base Station address and Sequence control of 2 Bytes in the beginning and CRC32 check in the end of the data. Total 28Bytes are added in the data as supported bytes, Length of Data is variable.

A separate Receiver section having a reverse cycle designed for transmitter with additional blocks to examine and overcome delay and as depicted in Figure 4.

Received data can be verified using Wireshark as well Message Debug block of GNU Radio. Value of Delay can be varied to examine the performance and effect over the received signal in terms of Frequency and phase correlation. This Flowgraph can be divided into two major section: Frame Detection and Frame decoding. Correct detection of frame by comparing received frame with delayed frame. A separate Transmitter is also shown in Figure 2.

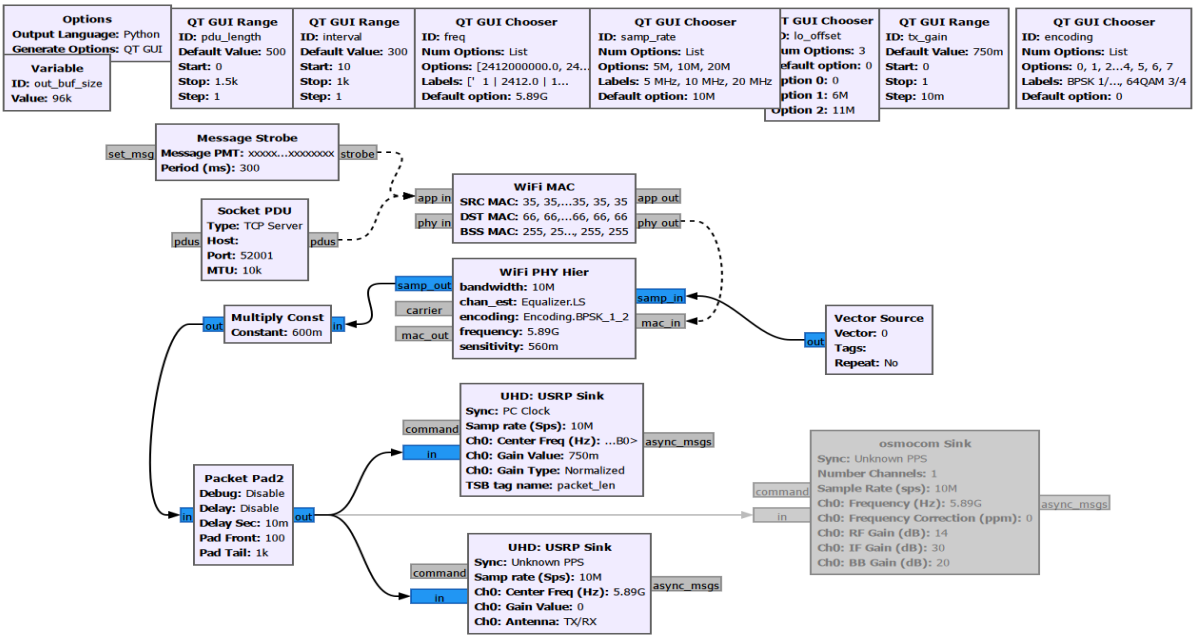

Fig. 2: GNU Radio Flow Graph for IEEE802.11 transmitter

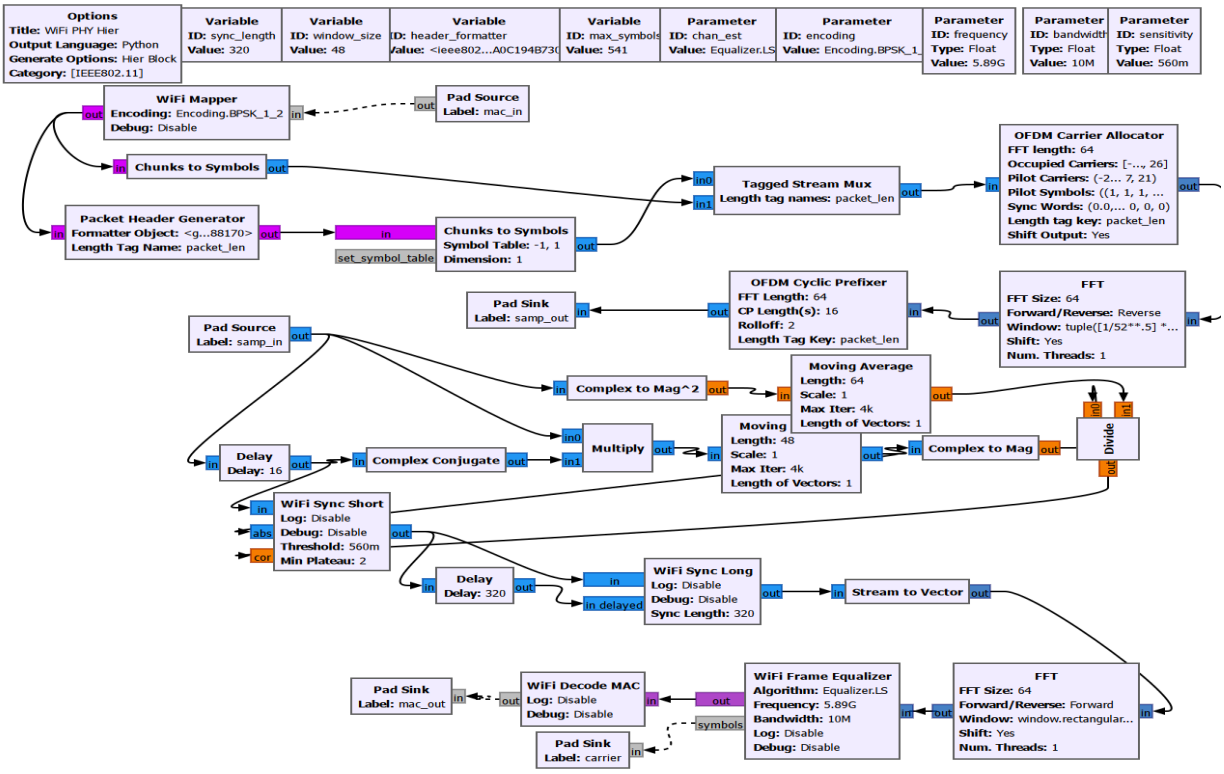

Fig. 3: GNU Radio Flow Graph for IEEE 802.11 PHY Hier Block

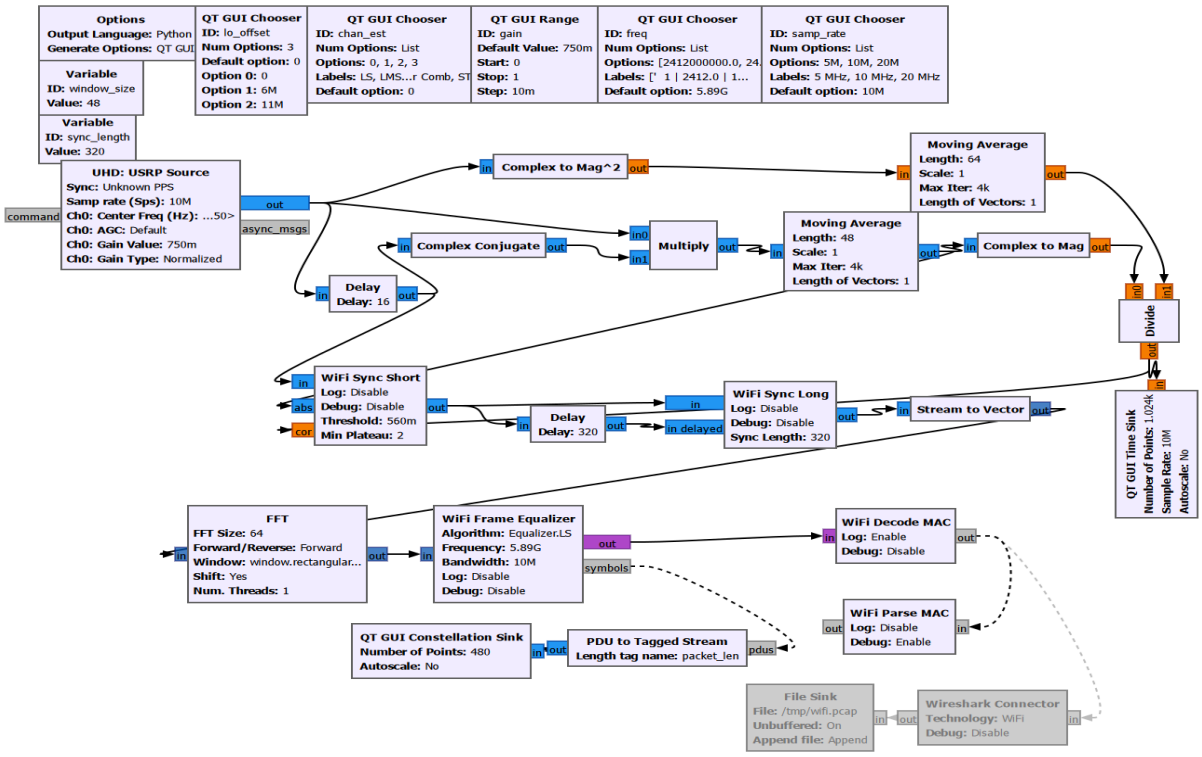

Fig. 4: GNU Radio Flow Graph for IEEE802.11 Receiver

First, low-order modulation techniques, namely BPSK and QPSK, were evaluated at code rates of 1/2 and 3/4. The same methodology was then applied to high-order modulation techniques, namely 16QAM and 64QAM, with code rates of 2/3 and 3/4. To evaluate the sensitivity of PRR (Packet Reception Rate) to signal strength, the transmit gain of the USRP B205mini-i was systematically varied from 0 dBm to 15 dBm in 1 dB increments. At each gain level, a fixed number of packets were transmitted, and the receiver computed the PRR.

5.1 Packet Reception Ratio (PRR)

Packet Reception Ratio (PRR) is a fundamental reliability metric in wireless communication systems. It quantifies the proportion of successfully received packets relative to the total number of transmitted packets. Formally, PRR is defined as:

$$\mathrm{PRR} = \frac{N_{\mathrm{success}}}{N_{\mathrm{total}}}$$

where Nsuccess denotes the number of packets correctly received and decoded and Ntotal represents the total number of packets transmitted during the observation period. PRR values range between 0 and 1 and are often expressed as a percentage:

$$\mathrm{PRR}(\%) = \frac{N_{\mathrm{success}}}{N_{\mathrm{total}}} \times 100$$

A higher PRR indicates greater link reliability and robustness against channel impairments, while a lower PRR reflects packet losses due to interference, fading, or congestion.

In experimental evaluations, PRR is typically measured alongside complementary metrics such as throughput, latency and Bit Error Rate (BER) to provide a comprehensive assessment of system performance.

At the receiver, GNU Radio flowgraphs performed carrier synchronization, timing recovery, equalization, demodulation and Viterbi decoding. Packet detection was achieved through preamble-based synchronization and

cyclic redundancy check (CRC) validation was used to determine successful packet reception. The measured PRR values were recorded for each transmission gain and modulation configuration. For statistical reliability, multiple trials were conducted at each gain setting and the reported PRR values represent averaged results. To ensure experimental consistency:

- The transmitter and receiver were positioned at a fixed separation distance.
- Antenna orientation and polarization were kept constant throughout the measurements.
- Experiments were conducted under similar environmental conditions to minimize uncontrolled channel variation.
- Identical hardware configurations were maintained when switching between 2.412 GHz and 5.89 GHz.

5.2 INTRODUCTION OF ENCRYPTION

We introduce AES-CBC (Advanced Encryption Standard – Cipher Block Chaining), a symmetric block cipher mode used to secure the transmitted data. Although IEEE 802.11 typically employs AES in CCM mode through WPA2/WPA3 for confidentiality and integrity, this work uses AES-CBC to develop a basic encryption prototype in GNU Radio, as shown in Figure 5. The embedded Python block was used for encryption implementation, and the resulting PRR measurements were analyzed against transmission gain for each MCS configuration. The comparison between 2.412 GHz and 5.89 GHz was used to study threshold shifts, transition slopes, and saturation behaviour under different modulation conditions.

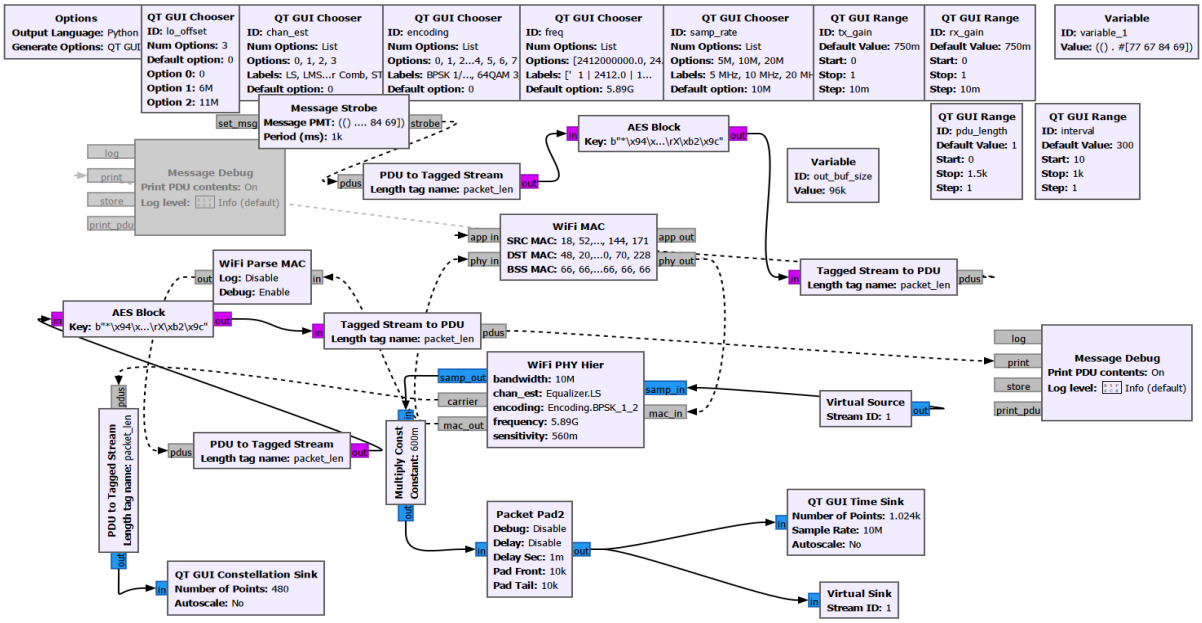

Fig. 5: GNU Radio Flow Graph for IEEE802.11 trans-receiver with AES-CBC Encryption

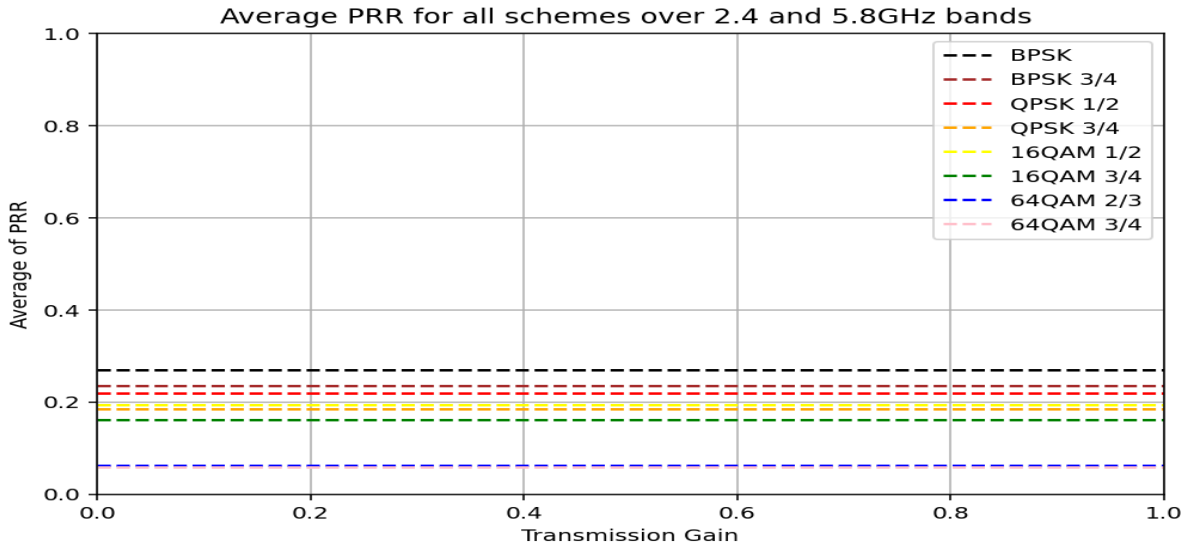

Fig. 6: Average Packet Reception Rate for all Modulation techniques over 2.412 and 5.89GHz Bands.

6. RESULT ANALYSIS AND OBSERVATIONS

With above practical setup, we will be examining performance of the trans receiver with a constant data packet over different centre frequencies and various modulation techniques. In this study, the comparative analysis is restricted to two fixed operating frequencies, 2.412 GHz and 5.89 GHz, and all subsequent PRR and confidence-interval results should be interpreted with respect to these two bands only. We evaluate and observe the process using PRR versus transmit gain curves for all modulation techniques. In addition, we interpret the results in terms of the effective Signal-to-Noise Ratio at the receiver.

6.1 Performance analysis for 5.89 GHz Band

For performance analysis, the ASCII message “MCTE123” (7 bytes) was used as the payload. After MAC-layer processing, 28 additional bytes were added for frame control, duration, source, destination, base-station addresses, sequence control, and CRC32, resulting in a 35-byte MAC frame. The destination address was set to [0x30, 0x14, 0x4a, 0xe6, 0x46, 0xe4], the source address to [0x12, 0x34, 0x56, 0x78, 0x90, 0xab], and the base-station address to [0x42, 0x42, 0x42, 0x42, 0x42, 0x42]. The CRC32/ISO-HDLC value was 0x5CBE06A8, and the final transmitted frame was 08 00 00 00 30 14 4a e6 46 e4 12 34 56 78 90 ab 42 42 42 42 42 42 30 02 4d 43 54 45 31 32 33 a8 06 be 5c. The frame was transmitted repeatedly at 1-second intervals, and PRR was measured over 10 packets at 1-meter distance using BPSK, QPSK, 16QAM, and 64QAM at different code rates. Figure 6 shows that PRR decreases as modulation order increases, with BPSK(1/2) providing the best overall performance. To improve the reliability of the reported results, the PRR values were obtained through repeated packet transmissions at each gain setting and then averaged for the corresponding modulation and coding scheme.

The gain sweep from 0 dBm to 15 dBm was maintained identically across both frequency bands to preserve fairness of comparison. This procedure ensures that the observed trends are representative of modulation-dependent receiver robustness rather than isolated trial behaviour. The confidence intervals shown in Figures 7–14 indicate that BPSK is the most suitable modulation scheme when coverage and link robustness are prioritized.

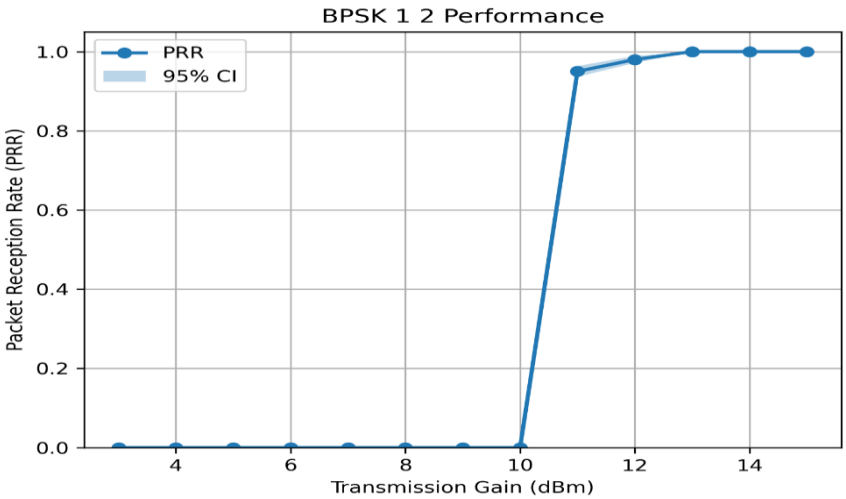

Fig. 7: Confidence Index for BPSK (1/2) over 2.412 and 5.89GHz Bands.

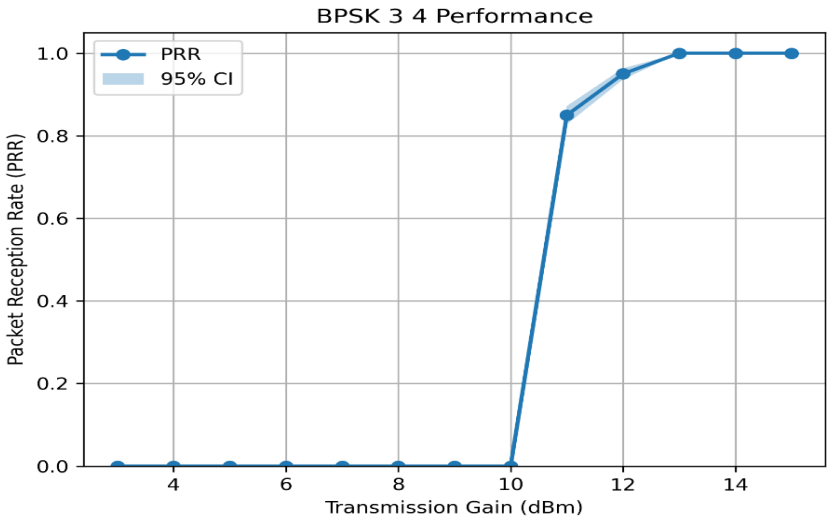

Fig. 8: Confidence Index for BPSK (3/4) over 2.412 and 5.89GHz Bands

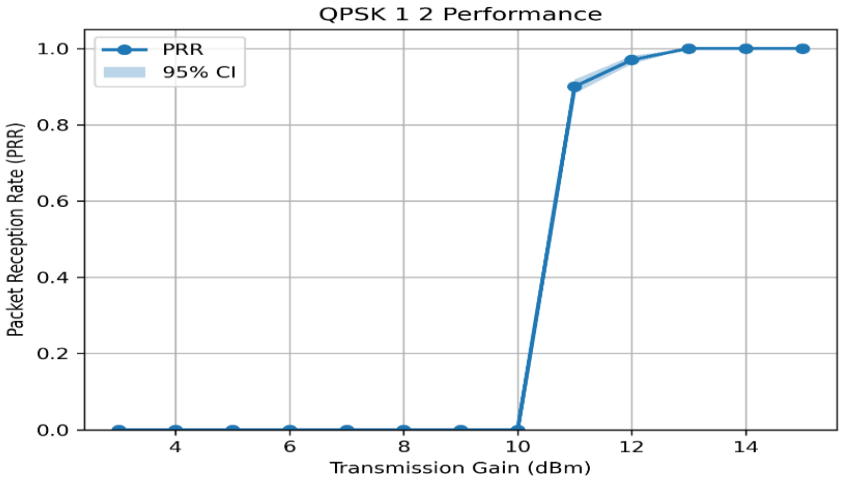

Fig. 9: Confidence Index for QPSK (1/2) over 2.412 and 5.89GHz Bands

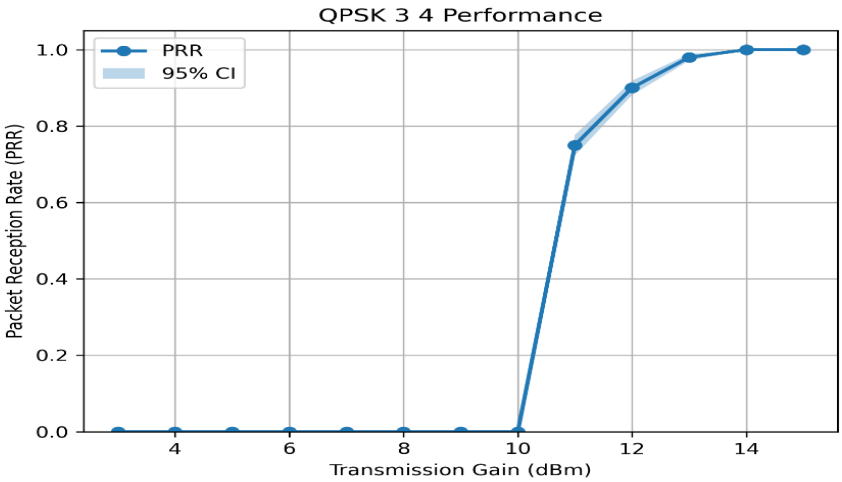

Fig. 10: Confidence Index for QPSK (3/4) over 2.412 and 5.89GHz Bands

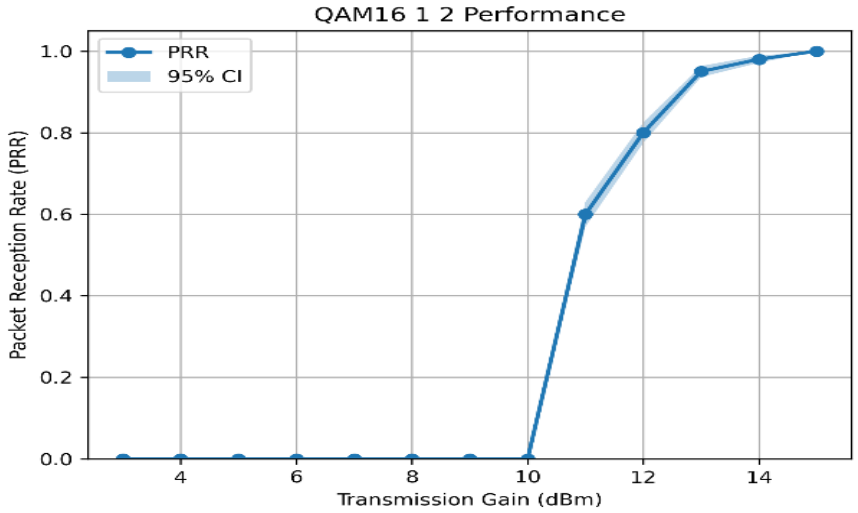

Fig. 11: Confidence Index for 16QAM (1/2) over 2.412 and 5.89GHz Bands

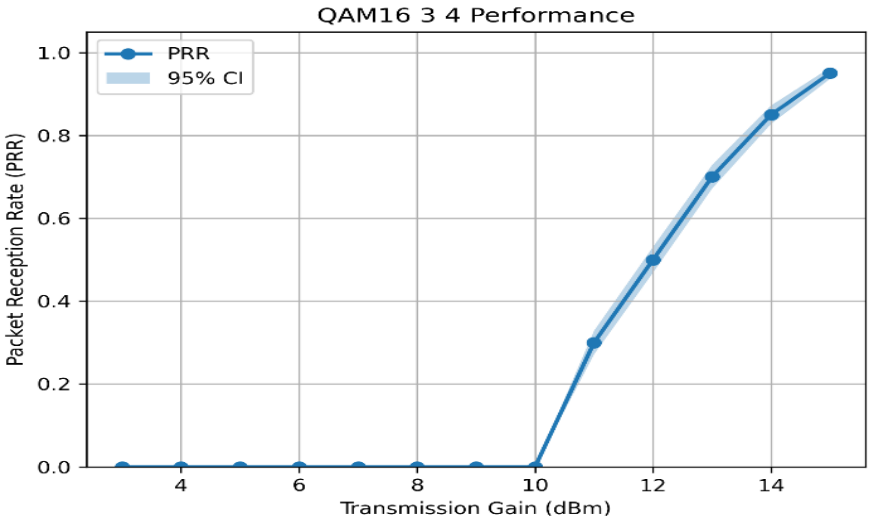

Fig. 12: Confidence Index for 16QAM (3/4) over 2.412 and 5.89GHz Bands

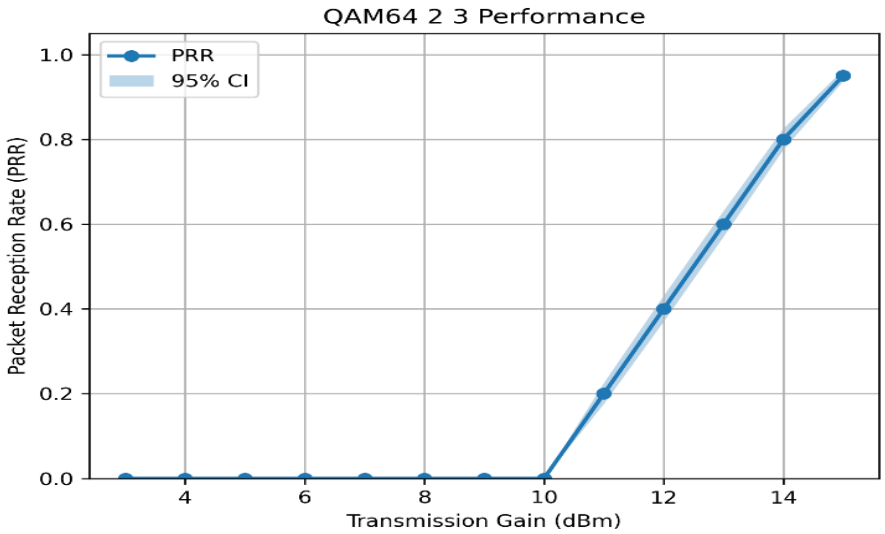

Fig. 13: Confidence Index for 64QAM (2/3) over 2.412 and 5.89GHz Bands

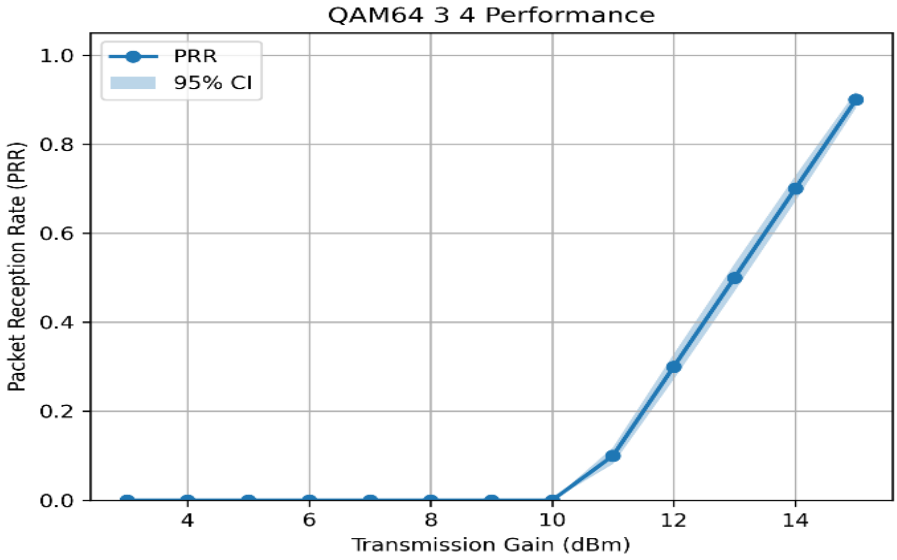

Fig. 14: Confidence Index for 64QAM (3/4) over 2.412 and 5.89GHz Bands

6.2 Encryption Implementation

We have implemented AES-CBC systematic encryption in Python embedded block of GNU radio and validated it using the following methods. Note: After Validation the Encrypted Output on GNU Radios is matching with the validation method but some characters are not printable and shown as ".". For "MCTE" word only last word 0x74 is showing as "N" and the first three are in non-printable categories but after cross verification we can conclude the correctness of the code.

6.3 Encryption Validation

Objective: Demonstrate functional correctness, determinism, and robustness of the AES–CBC implementation used in this work. Validation was performed using known-answer tests, PKCS#7 padding-boundary checks, and repeated encryption/decryption trials under identical settings.

The generated ciphertext was also cross-verified against a reference software implementation to confirm bit-level correctness and to ensure that the observed SDR behaviour resulted from the intended encrypted transmission process.

Let B = 16 bytes denote the AES block size. If the plaintext M is partitioned into blocks P1, P2, ..., Pn, then CBC encryption is performed as:

$$Ci = Ek(Pi \oplus Ci-1)$$

where C0 is the initialization vector (IV). The transmitted ciphertext is:

$$C = C0 \parallel C1 \parallel ... \parallel Cn$$

For decryption, the plaintext blocks are recovered as:

$$Pi = Dk(Ci) \oplus Ci-1$$

After decryption, the PKCS#7 padding bytes are removed to obtain the original plaintext. The validation process also included consistency checks for PKCS#7 padding insertion/removal and verification of ciphertext generation for fixed plaintext, key, and initialization vector combinations.

7. RESULT ANALYSIS

We implemented a reference AES–CBC encryption and decryption using the Python cryptography library. Additional validation was performed by checking correct PKCS#7 padding insertion and removal for short payloads and by verifying that decryption fails when incompatible parameters are applied. These checks are important because successful transmission alone does not guarantee cryptographic correctness at the block-processing level. Accordingly, the validation confirms both functional interoperability and internal consistency of the AES--CBC implementation used in the SDR chain. Plaintext is padded with PKCS#7 to 16 bytes, then encrypted block wise as $Ci = Ek(Pi \oplus Ci-1)$ with a 128-bit key. Decryption performs $Pi = Dk(Ci) \oplus Ci-1$ and removes PKCS#7 padding. The obtained results indicate that the dominant factor governing packet reception performance is the selected modulation and coding scheme rather than the presence of the encryption block itself. Lower-order modulation schemes retain higher robustness because they require lower effective SNR for correct symbol detection, whereas higher-order schemes show sharper PRR degradation as link margin decreases. From a practical perspective, this suggests that secure SDR-based IEEE 802.11 communication is most suitable under low- and medium-order modulation settings when reliable encrypted transmission is prioritized over peak spectral efficiency.

8. RESULT COMPARISON

We have compared the results in table 1 with previous researches for various parameters enlisted in Table 1. By the research study, we have observed absence of encryption in most the research paper yield makes it more practical testbench oriented than practical application. Table 1 highlights that the distinguishing contribution of the proposed study is the simultaneous inclusion of SDR implementation, embedded encryption, multiband evaluation, single-band analysis and GNU Radio realization in one experimentally validated framework. In contrast to prior works, the manuscript emphasizes reproducibility through validation scripts, test vectors and implementation-level integration, thereby strengthening its practical and scientific contribution. This comparison clarifies that the work advances beyond isolated PHY experimentation by introducing a secure and reproducible SDR communication prototype. All have used SDR, Multiband Analysis (MBA), Single-Band Analysis (SBA) and GNU as prominent tool but Encryption implementation is only introduced in our work with all mentioned parameters. Encryption can lead to more real time implementation for practical applications.

Table 1 Comparison of the proposed results with previously reported research works

Ref	SDR	Enc	MBA	SBA	GNU
[1] Bezerra et al. (2024)	Y	N	Y	Y	N
[2] Radu et al. (2020)	Y	N	N	Y	N
[4] Zheng et al. (2024)	Y	N	Y	Y	Y
[7] Polgar & Stef (2023)	Y	Y	Y	Y	Y

[9] Tang & Huang (2020)	Y	N	Y	Y	N
[10] Morman et al. (2022)	Y	Y	Y	Y	Y
[13] Kim et al. (2021)	Y	Y	Y	Y	N
This paper Proposed work	**Y**	**Y**	**Y**	**Y**	**Y**

9. CONCLUSION

This paper has presented a reproducible methodology for validating AES–CBC encryption in wireless communication systems. The experimental comparison was consistently performed over 2.412 GHz and 5.89 GHz operating bands, enabling uniform assessment of modulation-dependent PRR behaviour under encrypted SDR transmission conditions. The inclusion of reference Python code, OpenSSL verification commands and test vectors ensures transparency and facilitates independent replication of results. The definition and application of PRR further strengthen the evaluation of link reliability. By combining formal mathematical descriptions with practical artifacts, the work bridges the gap between theory and implementation. The present results also indicate that the integration of AES-CBC does not alter the overall modulation-performance ranking observed in the SDR link, where BPSK (1/2) remains the most robust configuration under reduced link margin. This observation is practically significant because it suggests that cryptographic integration can be achieved without changing the qualitative reliability behaviour of the wireless testbed. However, dedicated evaluation of encryption-induced latency, throughput reduction and processing overhead remains an important direction for future investigation.

Future research can extend this framework to other cipher modes, adaptive link adaptation algorithms and broader performance metrics, thereby advancing the reliability and security of modern communication systems.

ACKNOWLEDGMENTS

Some manuscript text and example code were prepared with assistance from AI tools, and the authors reviewed and verified all content. The reference implementation, test vectors, and verification scripts were produced and checked by the authors and are included in the supplementary material for reproducibility.

Funding Declaration: *The authors received no financial support for the research, authorship, and/or publication of this paper.*

Author Contributions

Bhavini K. Kumawat: *Conceptualization, SDR development, experimental implementation, data analysis, and original draft preparation.* ***Rajesh M. Bodade:*** *Supervision, methodology development, result validation, and manuscript review.* ***Gaurav Sharma:*** *Technical support, experimental validation, result interpretation, and manuscript review.*

Conflict of Interest

The authors declare that there is no conflict of interest regarding the publication of this work.

References

1. G.M.G. Bezerra, N.R. de Oliveira, T.N. Ferreira et al. “A comprehensive evaluation of software-defined radio performance in virtualized environments for radio access networks”, Annales des Télécommunications, 79(4), pp. 523–535, 2024. Available at: https://doi.org/10.1007/s12243-024-01044-2 (Accessed on: 21 June 2026).
2. F. Radu, A. Timofte, A. Balan, F. Sandu. “LTE Communications Using an SDR Platform”, in Proceedings of the 13th International Conference on Communications (COMM), Bucharest, Romania, pp. 393–396, 2020. Available at: https://doi.org/10.1109/COMM48946.2020.9141979 (Accessed on: 21 June 2026).
3. M. Dillinger, K. Madani, N. Alonistioti. Software Defined Radio: Architectures, Systems and Functions. John Wiley & Sons, Chichester, 2005.
4. Jingze Zheng, Chaojie Gu, Yuanchao Shu, Xiuzhen Guo, Shibo He, Zhiguo Shi, Jiming Chen, "SoftNB: A Fully Functional NB-IoT PHY for Various SDR Platforms", 2024 IEEE 32nd International Conference on Network Protocols (ICNP), pp.1-11, 2024.
5. B. Bloessl, M. Segata, C. Sommer and F. Dressler, "Towards an Open Source IEEE 802.11p stack: A full SDR-based transceiver in GNU Radio," 2013 IEEE Vehicular Networking Conference, Boston, MA, USA, 2013, pp. 143-149, doi: 10.1109/VNC.2013.6737601.

6. Bastian Bloessl, Michele Segata, Christoph Sommer and Falko Dressler. 2013. An IEEE 802.11a/g/p OFDM receiver for GNU radio. In Proceedings of the second workshop on Software radio implementation forum (SRIF '13). Association for Computing Machinery, New York, NY, USA, 9–16. https://doi.org/10.1145/2491246.2491248
7. Z. A. Polgar and M. Stef, "OFDM Transceiver with Adaptive Modulation Implemented in GNU Radio," 2023 46th International Conference on Telecommunications and Signal Processing (TSP), Prague, Czech Republic, 2023, pp. 37-42, doi: 10.1109/TSP59544.2023.10197787
8. F.L. Crespi, M. Maglioli, S. Benco and A. Perotti, "A real-time video broadcasting system based on the GNU Radio-USRP2 platform," 7th Karlsruhe Workshop on Software Radios, Karlsruhe, Germany, March 2012, pp. 42 - 46.
9. X.-W. Tang and X.-L. Huang, "A design of SDR-based pseudo-analog wireless video transmission system," Mobile Network Applications, vol. 25, pp. 2495–2505, Dec. 2020.
10. J. Morman, M. Lichtman and M. Muller, "The Future of GNU Radio: Heterogeneous Computing, Distributed Processing and Scheduler-asa- Plugin," MILCOM 2022 - 2022 IEEE Military Communications Conference (MILCOM), Rockville, MD, USA, 2022, pp. 180-185, doi: 10.1109/MILCOM55135.2022.10017973
11. Sumit Kumar, Garimella Ramamurthy: "Efficient Spectrum Sensing/- Monitoring Methods and Testbed Development for Cognitive Radio based WSN" 2014 Wireless Innovation Forum Conference on Communications Technologies and Software Defined Radio (SDR-WInnComm 2014)
12. Sumit Kumar, Chandan Pradhan, Kunal Sankhe, Garimella Ramamurthy: "Cognitive Base Station Design for Efficient Spectrum Utilization in Cellular Network" Eleventh International Conference on Wireless and Optical Communications Networks WOCN 2014
13. W. Kim, S. Kim and H. Lim, "Malicious Data Frame Injection Attack Without Seizing Association in IEEE 802.11 Wireless LANs," in IEEE Access, vol. 9, pp. 16649-16660, 2021, doi: 10.1109/ACCESS. 2021.3054130.
14. C.P. Kohlios, T. Hayajneh. "A Comprehensive Attack Flow Model and Security Analysis for Wi-Fi and WPA3", Electronics, 7(11), p. 284, 2018. Available at: https://doi.org/10.3390/electronics7110284 (Accessed on: 21 June 2026).
15. P. Fuxj\"ager, A. Costantini, D. Valerio, P. Castiglione, G. Zacheo, T. Zemen and F. Ricciato. IEEE 802.11p Transmission Using GNURadio. In 6th Karlsruhe Workshop on Software Radios (WSR), pages 1--4, Karlsruhe, Germany, March 2010.