JTE
HCMUTE
ISSN: 2615-9740

JOURNAL OF TECHNICAL EDUCATION SCIENCE
Ho Chi Minh City University of Technology and Education
Website: https://jte.edu.vn
Email: jte@hcmute.edu.vn

Development and Implementation of an LCMV Adaptive Beamforming Antenna Using USRP and GNU Radio

Khang Nguyen Viet Thai, Khanh Nguyen Tuan*, Thuyet Nguyen Vo That
Vietnamese-German University, Vietnam

*Corresponding author. *Email: khanh.nt@vgu.edu.vn*

ARTICLE INFO

Received: 09/07/2025
Revised: 15/08/2025
Accepted: 13/10/2025
Published: 28/11/2025

KEYWORDS

Adaptive Beamforming;
LCMV;
USRP;
GNU Radio;
Digital twin.

ABSTRACT

This present paper introduces the development and implementation of the Linearly Constrained Minimum Variance (LCMV) adaptive beamforming antenna system with two Universal Software Radio Peripheral (USRP) boards to receive and transmit the processed data through GNU's Not Unix (GNU) Radio Companion software. The collected data will then be displayed through an external dashbased Visual Studio Code Program in order to accurately measure the performance of the LCMV beamforming algorithm while being compared with the conventional antenna system (Delay-and-Sum). The comparison criteria of the performance metrics between the 2 systems in this paper are: White Noise Gain (WNG), Null Depth, Interference Suppression, and Half-Power Beam Width (HPBW). The Flowgraph, which represents the logic of the two antenna systems in GNU Radio Companion (GRC), is subjected to noise interference to further reflect their performance in real-world settings. The research also demonstrates a clear difference in the performance between the two antenna system models, establishing a foundation for developing digital twin-based communication systems that enable continuous monitoring, simulation, and optimization in future wireless networks.

Doi: https://doi.org/10.54644/jte.2025.1965

1. Introduction

As world technology has progressed rapidly over the last decade, the demand for an efficient communication system in a congested and noisy dynamic environment has grown significantly [1]. The traditional methods, such as the Delay-and-Sum (DAS) technique, have become unsuitable for practical application due to their limitations in a complicated signal environment. Hence, the advanced method of adaptive beamforming processing techniques with higher precision is utilized in order to reach the requirement threshold of complex communication systems, of which the Linearly Constrained Minimum Variance (LCMV) algorithm is a suitable candidate capable of solving the problem.

Through one of the foundation of adaptive beamforming techniques which are the Minimum Variance Distortionless Response (MVDR), the theoretical foundation of the Linearly Constrained Minimum Variance were introduced when the research tried to address the issues of robustness [2], such as uncertainties in the covariance matrix estimation, while improved the practicality of the MVDR antenna processing method, from here, the LCMV algorithm has been continuously developed and improved over the years. More recently, advancements such as digital twin-assisted beamforming have emerged, enabling real-time interaction between physical environments and their virtual replicas to further optimize beam patterns and adapt to dynamic wireless conditions [3].

Hence, with all of the necessary qualities that are already present in the LCMV adaptive beamforming technique, this research aims to utilize the beamforming algorithm and implement the advanced signal processing method in a real-time beamforming system, which also leverages the developed interactive visualization tools to accurately measure the performance metrics of the real-time system. The research is realized by using the GNU Radio Companion Software by integrating the signal processing process

JTE
HCMUTE
ISSN: 2615-9740

JOURNAL OF TECHNICAL EDUCATION SCIENCE
Ho Chi Minh City University of Technology and Education
Website: https://jte.edu.vn
Email: jte@hcmute.edu.vn

in the flowgraph and creating the logic for both antenna system models through the embedded python blocks where the computed snapshot data is then sent to the visualization tools which provides an intuitive, real-time display of the radiation pattern with key performance metrics [4] as followed: DOA, Null Depth, HPBW, Interference Suppression, and WNG.

The remainder of the paper is organized as follows: Section 2 describes the methodology, including the GNU Radio Flowgraph, the mathematical foundations, and the hardware setup of both antenna systems. Section 3 presents the experimental results, timing and snapshot statistics, and the performance comparisons using time-domain, constellation, and radiation patterns plots. Section 4 discusses the limitations, interprets the results, and proposes improvements. Finally, Section 5 concludes the paper and outlines the future work, including digital-twin integration.

2. Methodology

In this section, the details regarding the methodology used to develop a real-time adaptive antenna system, where USRP B210 devices serve as the critical interface between the real-world environment and the digital processing domain in GNU Radio. The USRP devices capture analog RF signals that are digitized and then segmented into two distinct data streams, where both are processed by a dedicated Python-based algorithm implementing the LCMV and DAS beamforming methods. Furthermore, Post-processing is achieved by monitoring the resulting data through the built-in functions in GNU Radio, or through a custom external dashboard developed using Visual Studio Code and Dash for advanced performance evaluations.

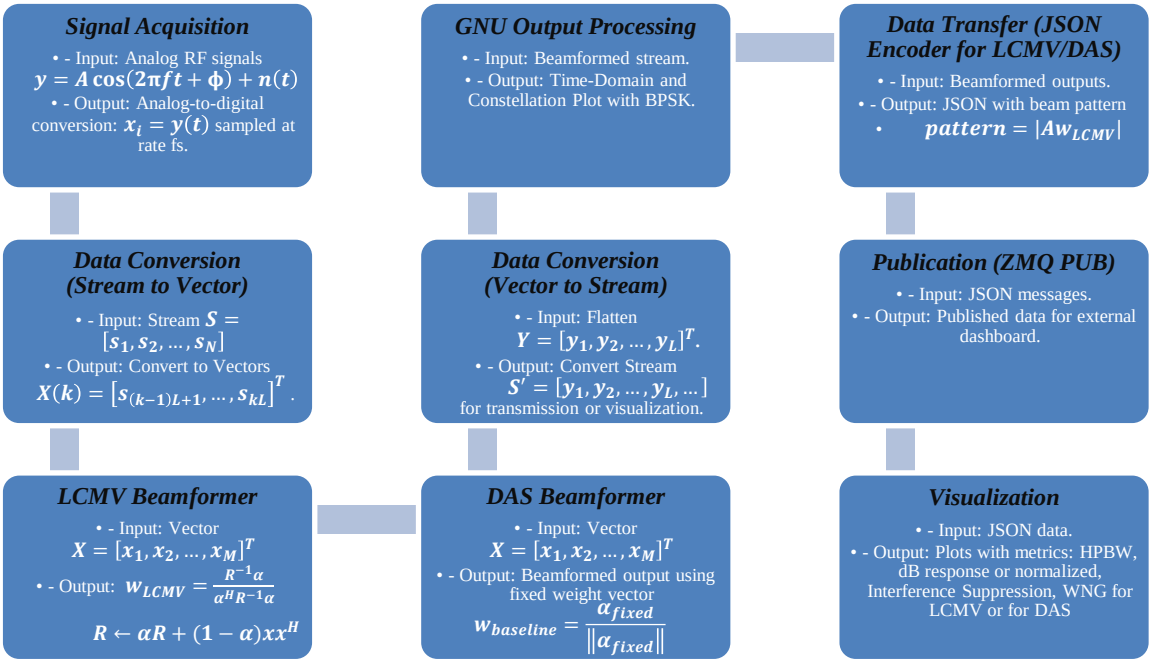

Figure 1. *Signal Processing and Visualization Pipeline.*

As shown in Figure 1, the Block diagram illustrates the overall system architecture, which highlights the priorities of the research: Implementing the LCMV beamforming algorithm with Python and establishing a robust communication channel between GNU Radio and the dashboard [5], which is essential for real-time performance monitoring. This framework not only ensures accurate interference suppression and detailed performance metric analysis (including white noise gain and beamwidth) but also provides a foundation for extending these adaptive signal processing techniques to other applications, such as robot arm control [6]. The following sections will further explain the adaptive beamforming formulas and iterative calculation that form the core of the LCMV approach used in this project.

JTE
HCMUTE
ISSN: 2615-9740

JOURNAL OF TECHNICAL EDUCATION SCIENCE
Ho Chi Minh City University of Technology and Education
Website: https://jte.edu.vn
Email: jte@hcmute.edu.vn

2.1. GNU Radio Schematic

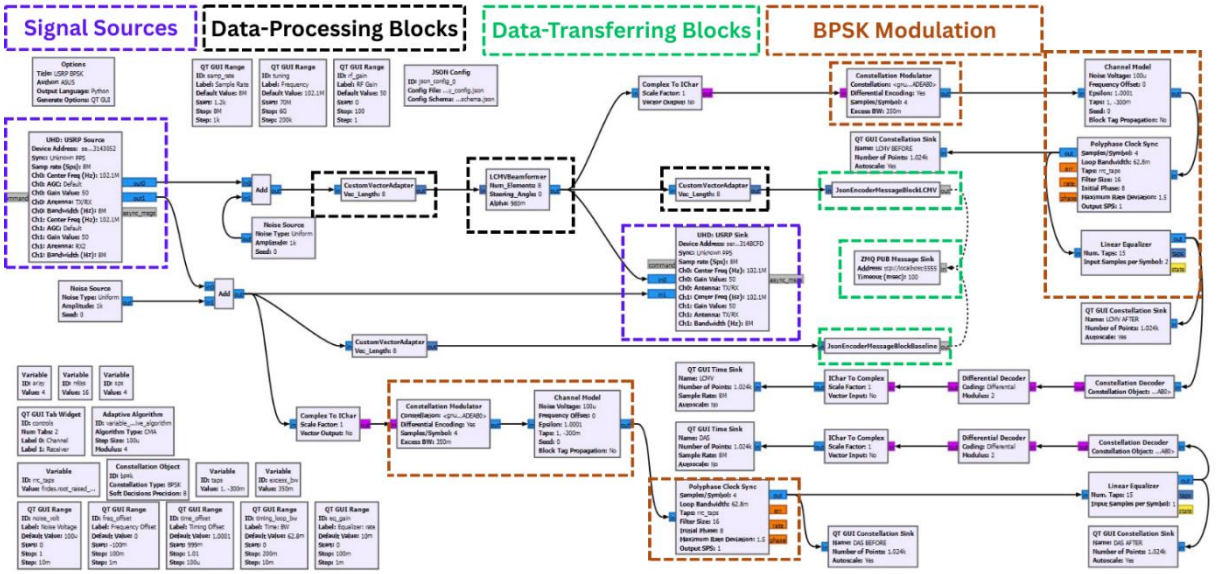

Figure 2. *The GNU Radio Flowgraph.*

A flowgraph option in GNU Radio Companion serves as the architectural blueprint of a software-defined radio, which represents a directed network of signal-processing blocks that collectively define the system's data flow and logic. As shown in Figure 2 (and summarized in Figure 1), the project design is organized into three functional categories: Signal Sources, Data-Processing Blocks, and Data-Transferring Blocks.

Signal Sources include both generic and hardware-specific elements. Pure Source Blocks generate the baseband signals and inject (set to 1000) is a simulation-specific that defines the standard deviation of the noise signal, while aligning with the noise models in digital communications [7], [8] to emulate challenging interference conditions which this high noise level yields more snapshots for robust LCMV beamforming. In parallel, the USRP Source Block interfaces directly with the USRP hardware devices connected with the antenna, digitizing the incoming RF signal into complex I/Q samples for downstream processing. On the other hand, the output standard Sink Blocks (time domain and constellation graph) which is modulated by Binary Phase-Shift Keying (BPSK) through Channel Model and Polyphase Clock Sync option [9] to visualize accurate data within GRC application, whereas the USRP Sink reconverts beamformed digital data into analog RF signal, amplifies it, and retransmits the signal via the antenna connected to the hardware devices.

Data-Processing Blocks occupy the core function of the flowgraph while performing three primary tasks in this research, which are:

1. Data Conversion: Custom Stream-to-Vector and Vector-to-Stream blocks reshape the continuous I/Q samples into fixed length vector for efficient matrix operations.
2. Beamforming Algorithm: An embedded Python program implements the LCMV (and, for comparison, DAS) algorithm, updating covariance matrices with exponential averaging and computing adaptive weight vectors.
3. Data Merging: An Add block that combines parallel outputs and ensures the LCMV and DAS paths can be directly compared without creating additional schematics.

Lastly, the Data Transferring Blocks serialize both antenna-system outputs into JavaScript Object Notation (JSON) messages, which are published through Zero Message Queue (ZMQ). These messages feed an external Dash-based web application, where the mentioned criteria of the performance metrics are computed and rendered as Polar [10] and Cartesian Plots.

2.2. Practical Implementation and Theoretical Framework of the Project

2.2.1. Practical Application

As outlined above, the flowgraph serves as the software-defined radio's architecture, which translates the theoretical signal-processing concepts into practical and executable modules. Furthermore, each custom block has been modified to integrate the project algorithms that adapt to varying RF conditions, perform real-time sample synthesis efficiently, and relay the results to the research external dashboard.

Table 1 demonstrates each block's interfaces and primary functions within the project framework.

Table 1. *Functional Overview of the Key Blocks in the design schematic.*

Block Name	Inputs	Outputs	Core Functionality
USRP Source	RF channel, gain	Stream of complex64	Digitizes antenna I/Q into a sample stream
Stream to Vector	complex64 stream	(complex64, N) vectors	Buffers N samples, outputs fixed-length vectors
LCMV Beamformer	(complex64, N) vectors	(complex64, N) vectors	Updates covariance, computes weight vector, applies beamforming
Vector to Stream	(complex64, N) vectors	complex64 stream	Flattens vectors back to the sample stream
JSON Encoder (LCMV/DAS)	(complex64, N)	PMT-wrapped JSON	Computes instantaneous beam pattern, packages angles & values
ZMQ PUB	JSON messages	—	Publishes to TCP socket for dashboard

As seen in the table, the signal processing technique is implemented in the GRC flowgraph, which is composed of a series of interconnected blocks, where each serves a distinct role in transforming, analyzing and transmitting antenna signal data. After the data has been vectorized, it is processed by the LCMV Beamformer, a custom block which performs adaptive beamforming by updating the covariance matrix, computing the optimal weight vector, and applying it to steer the antenna beam toward the desired direction of arrival (DoA) while suppressing interference [11]. The computed output vectors are then converted back into continuous data for real-time visualization or further serial processing stages.

Additionally, to enable the mentioned external monitoring and control of the signal system, the processed data is then passed through the JSON encoder blocks, which extract the key parameters such as the instantaneous beam pattern and format them into JSON messages wrapped in PMT (Polymorphic Types) for standardized communication [12]. Finally, the ZeroMQ (ZMQ) PUB block publishes these encoded messages over a TCP socket, facilitating integration with an external dashboard or interface for interactive visualization and system feedback since ZMQ offers a flexible and efficient way to send messages across the process, system, and network with its high-performance message library [13].

2.2.2. Theoretical Foundations of the Algorithm

In this section, the core principles and formulas of the proposed system algorithm are explained in detail to clarify how they are implemented and integrated within the project. The use of these mathematical foundations is essential to ensure that the system performs adaptive beamforming accurately and efficiently.

First of all, the algorithm in the LCMV embedded Python block is designed to achieve adaptive signal processing by optimizing the antenna array performance while under constrained conditions. Its main goal is to reduce the array's total output power while maintaining a distortionless response in the desired direction by applying a linear constraint on the beamformer weight. Mathematically, the LCMV beamformer computes the optimal weight vector as formulated below:

$$w_{LCMV} = \frac{R^{-1}\alpha}{\alpha^H R^{-1}\alpha} \quad (1)$$

Where:

- w_{LCMV} is the beamforming weight vector.
- R is the sample covariance matrix of the received signal.
- R^{-1} is the inverse (or pseudo-inverse) of the coordinate matrix.
- α is the steering vector for the desired direction.
- α^H is the Hermitian transpose (conjugate transpose) of the steering noise.

This approach enables the LCMV algorithm to adaptively suppress interference while maintaining fidelity to the desired signal, making it particularly effective for real-world applications where environmental uncertainties are prevalent [2]. Next, the subsequent step of the algorithm involves the utilization of the steering vectors, which encodes the anticipated phase shifts for the target direction [11] and is derived from the following mathematical expression.

$$\alpha_n = exp(-j\pi n sin(\theta)) \quad (2)$$

Where:

- α_n is the nth element of the steering vector.
- j is the imaginary unit.
- π is the constant pi.
- n is the index of the antenna, and
- θ is the steering angle in radians.

The steering vector expression is subsequently implemented in Python and incorporated into the LCMV beamforming block within GRC.

In the final stage of the LCMV algorithm, the incoming data vectors are processed to generate beamformed outputs for each snapshot. Eq. (3) below details this covariance update mechanism, which is implemented in the algorithm Work method to execute the adaptive LCMV beamforming algorithm system, which generates the desired output at the targeted location and angles.

$$R \leftarrow \alpha R + (1-\alpha)xx^H \quad (3)$$

where:

- R is the covariance matrix of the received signal.
- α is the forgetting factor (with $0 \leq x \leq 1$), which determines the weight given to previous snapshots.
- x is the received signal snapshot (a vector), and
- x^H denotes the Hermitian (conjugate transpose) of x.

With the combination of these formulas, the LCMV beamforming algorithm embedded Python blocks in the flowgraph are created, which enable the beamformer to perform as intended while sending the output data to the external visualization tools in order to determine the performance metrics of the system. Although the primary LCMV beamforming block handles the core adaptive processing, the LCMV JSON encoder algorithm still re-computes the beam pattern with Eq. (4) to package up-to-date metrics into a self-contained JSON message. This ensures real-time monitoring of beamforming performance without disrupting the data flow.

$$pattern = |Aw_{LCMV}| \quad (4)$$

where:

- A is the steering matrix, a precomputed array of steering vectors. While w_{LCMV} is the adaptive weight vectors of the LCMV beamformer.

On the other hand, the DAS algorithm is integrated in the DAS JSON encoder block by implementing Eq. (4) for beam pattern calculation, while for the DAS algorithm, the weight vector is derived by normalizing the fixed steering vector, which imposes predetermined phase shifts on the incoming signals, which are expressed through Eq. (5) below.

$$w_{baseline} = \frac{\alpha_{fixed}}{\|\alpha_{fixed}\|} \tag{5}$$

where:

- $w_{baseline}$ represent the fixed weight vector for the DAS beamformer.
- $\frac{\alpha_{fixed}}{\|\alpha_{fixed}\|}$ is the fixed steering vector for the desired steering angle and its Euclidean norm, which ensures the weight is normalized for unit gain in the steering direction.

Building on the previous formula in the flowgraph, this section explores the formulation used in developing the Visualization Tools, starting from the HPBW function, which is expressed as the difference between the two interpolated angles, as the Eq. (6):

$$HPBW = \theta_{right} - \theta_{left} \tag{6}$$

where:

- θ_{right} is the Interpolated angle on the right side of the main lobe.
- Whereas θ_{left} Representing the interpolated angle on the left side of the main lobe.

The visualization tools implement two modes for converting beam pattern response to decibels, when the setting is set to use raw data, the pattern is directly converted from its raw amplitude values into dB using Eq. (7), while in the other setting, the beam pattern is computed through Eq. (8). By presenting both raw-metric and normalized-power views in the visualization tools, the project offers complementary insight, such as the raw-based dB scale highlights absolute signal levels and noise floor characteristics, while the normalized conversion emphasizes relative beam shape and comparative metrics (such as Null Depth) in a manner that is easier to interpret across different scenarios.

$$dB_{response} = 10log_{10}(max(response, \epsilon)) \tag{7}$$

$$dB_{response} = 10log_{10}(max(\frac{response}{max(response)}, \epsilon)) \tag{8}$$

where:

- $dB_{response}$ is the decibel representation of the beam pattern response.
- $response$ is the raw amplitude or power value of the beam pattern at a specific angle θ.
- ϵ is a small positive constant to ensure numerical stability.
- $\frac{response}{max(response)}$ is the normalized response, which highlights relative shape and sidelobe characteristics.

The other performance metrics, which are Interference Suppression and WNG calculation, are implemented through their respective formula below, while the Null Depth is determined as the 10th percentile of the dB response over the sidelobe region.

$$Inference\ Suppression(dB) = -\rho_{90}(\{dB(\theta): \theta\ \epsilon\ S\}) \tag{9}$$

where:

- $dB(\theta)$ is the beam pattern response in dB at angle θ.
- S is the set of angles in the sidelobe region: $S = \{\theta || \theta - DOA| > \frac{HPBW}{2}\}$
- ρ_{90} denotes the 90th percentile value of the dB response within S.

$$White\ Noise\ Gain\ (dB) = -\frac{1}{|S|}\sum_{\theta \epsilon\ S} dB\ (\theta) \tag{10}$$

where:

- $|S|$ is the number of angle samples in the sidelobe region.
- $\sum_{\theta \epsilon\ S}\ is$ The summation over all angles θ within the sidelobe region S.
- $dB(\theta)$ is the power level in decibels at angle θ , defined as $10log_{10}(P(\theta))$, where $P(\theta)$ Is the normalized power of the beam pattern at the angle θ (relative to the Peak power).

However, the White noise gain result for the DAS system is computed separately with its own formula, which reflects the theoretical results where the gain of such an array is guaranteed through fixed weights and does not adapt to the signal flow. The formula below is the expression of the implementation.

$$White\ Noise\ Gain\ (dB) = 10log_{10}(N) \tag{11}$$

where:

- N is the number of elements in the uniform linear array

2.2.3. Hardware Configuration of the Experimental Setup

Alongside the flowgraph and the external Dash application. The experiment setup incorporates 2 USRP B210 SDR boards, each fitted with a dual-band vertical antenna on its RF ports. Compared to other SDR solutions, USRP devices offer robust real-time transmission and receive capabilities, an open-source software ecosystem, and seamless integration with tools such as GNU Radio Companion [14]. Each USRP is connected to the host computer via USB and is identified by its serial address in the Source and Sink blocks within the GRC flowgraph to designate one board as the transmitter and the other as the receiver [15]. On the other hand, the setup configuration only used 1 antenna for each USRP device due to the limitation in available hardware devices; however, the project made up for the lack of options by recording the signal from the single antenna and emulating it as a spatial snapshot for simulation [16], which makes the system capable of achieving desired results. Figure 3 demonstrates the physical setup of the test environment.

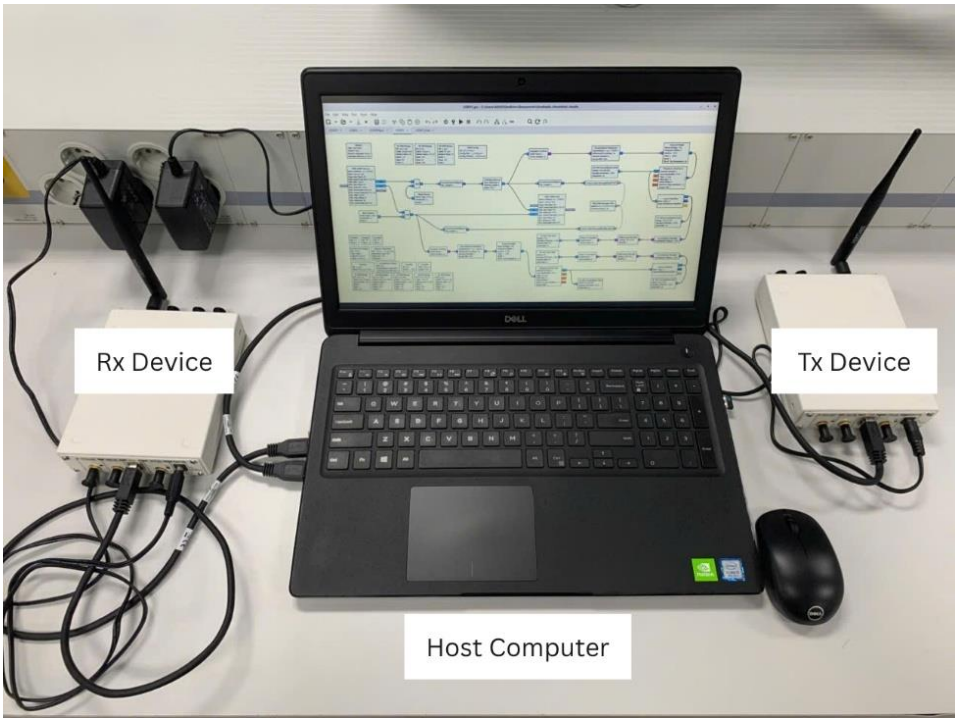

Figure 3. *Physical Setup of the Test Environment.*

JTE
HCMUTE
ISSN: 2615-9740

JOURNAL OF TECHNICAL EDUCATION SCIENCE
Ho Chi Minh City University of Technology and Education
Website: https://jte.edu.vn
Email: jte@hcmute.edu.vn

3. Results

In this section, the project presents and compares the beamformed outputs produced by the LCMV adaptive algorithm and the conventional DAS method, obtained in GNU Radio Companion visualization tools and the external dashboard developed through Visual Studio Code [17]. The evaluation begins with a analysis of the time-domain traces and constellation plots to assess the performance metrics. This is followed by a comparison using the dashboard, which displays the instantaneous radiation patterns and evaluates key performance metrics. The primary criteria for comparing the two systems are WNG, Null Depth, Interference Suppression, and HPBW. In addition to these beamforming performance metrics, the computational efficiency of the implementation was also measured. According to our configuration, the average beamforming time per snapshot is 1.034ms over a total of 19,912 snapshots.

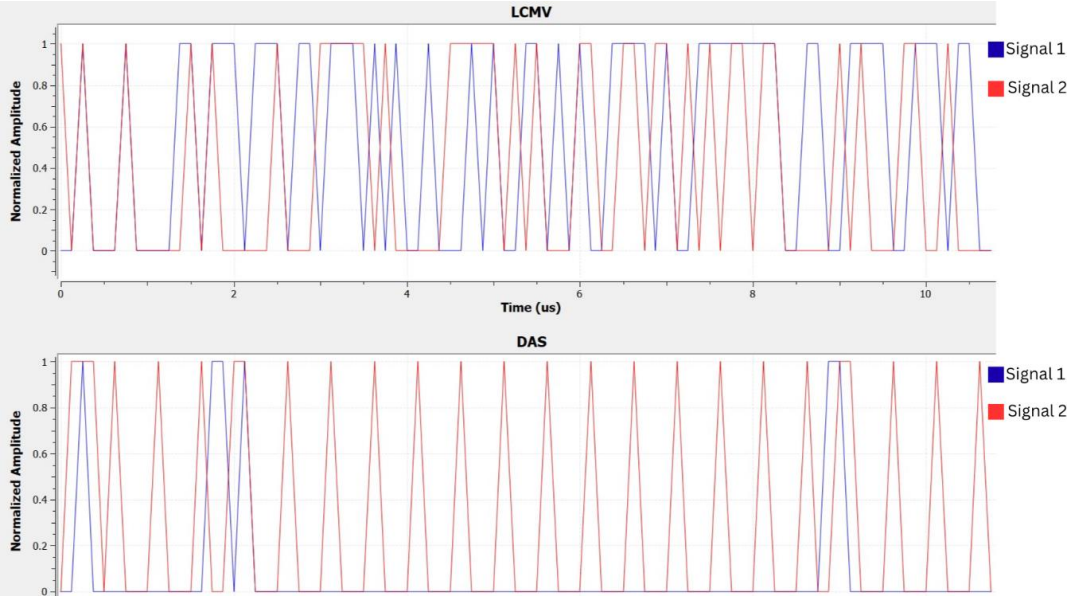

Figure 4. *Time-Domain Signal Visualization.*

Figure 4 shows the Time Sink outputs from the GRC flowgraph, illustrating the in-phase (real) and quadrature (imaginary) components for both the LCMV and DAS systems over successive update intervals [18], and demonstrates the normalized amplitude. These traces reveal the system stability and signal complexity: where the LCMV response is notably steadier [19], with the plot shows clean, well-defined digital signal, where both signal data is represented, and switching between high and low levels as the data is being preserved with a high signal-to-noise ratio (SINR) while the opposite happen with the DAS systems which is the results of the data being corrupted by interference.

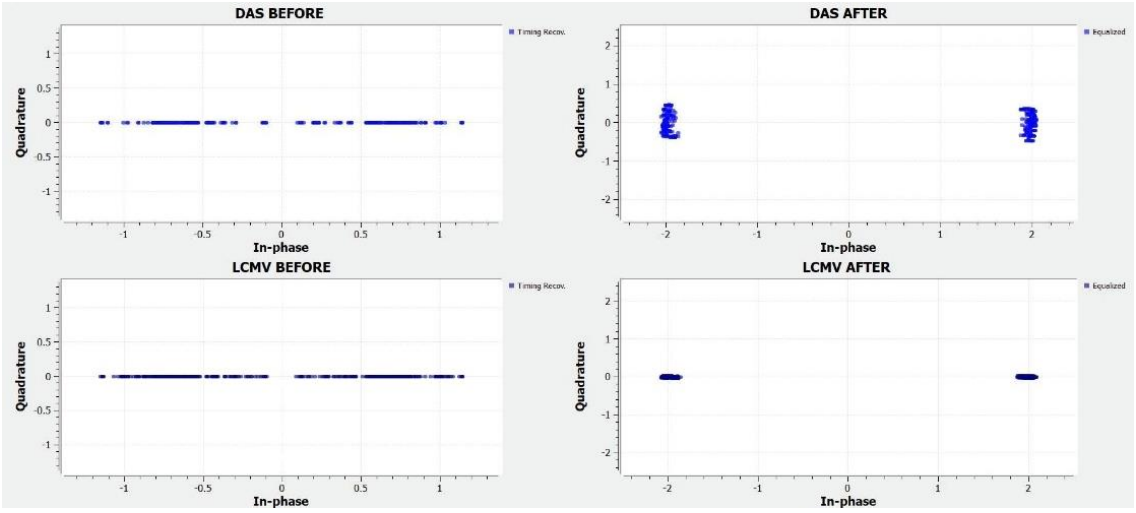

Figure 5. *Constellation Plot of Received Signal.*

JTE
HCMUTE
ISSN: 2615-9740

JOURNAL OF TECHNICAL EDUCATION SCIENCE
Ho Chi Minh City University of Technology and Education
Website: https://jte.edu.vn
Email: jte@hcmute.edu.vn

Figure 5 presents the constellation outputs from the GRC flowgraph sub-branches, which run in addition with timing synchronization and channel equalization for BPSK Modulation. Constellation plots visualize modulation fidelity by showing amplitude and phase variation in the complex signal [20]. In Figure 5, the LCMV points form a tighter cluster compared to the more dispersed DAS scatter in both before and after graphs, indicating that LCMV has effectively suppressed noise and interference [21]. This improved clustering corresponds to a higher SINR and lower error vector magnitude, directly translating into better signal quality and reduced bit-error rates (BER) [22].

Figure 6 and Figure 7 display the instantaneous radiation patterns for both DAS and LCMV at the selected steering angle, using either normalized or raw data to evaluate real-time performance. These polar plots reveal each array peak response, computed inherently via covariance updates and steering-vector alignment rather than external sensor [23]. When normalized, both systems align closely with the intended −75° direction, though the calculated DoA shows a minor −74° offset and the AS main lobe exhibits ripple, indicating incomplete noise suppressing.

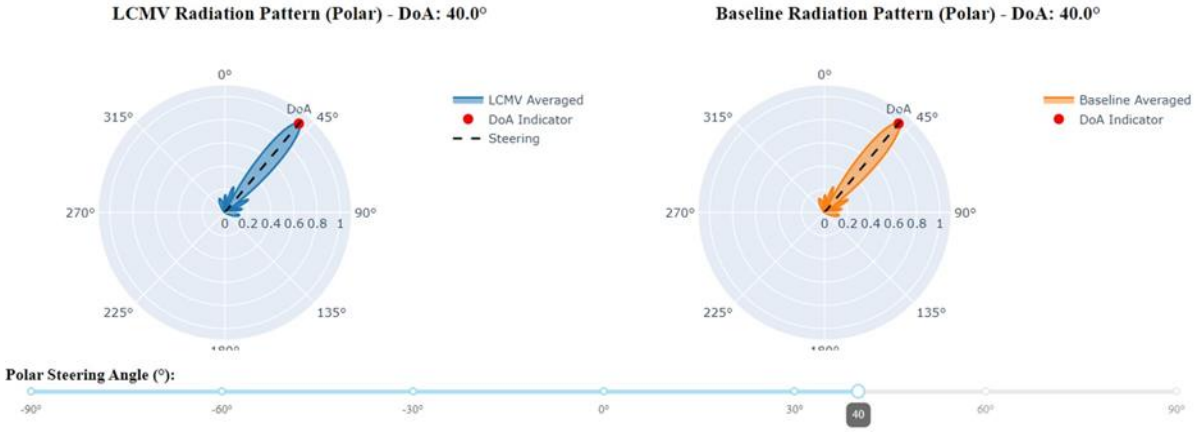

Figure 6. *Radiation Diagram Using Normalized Signal Strength.*

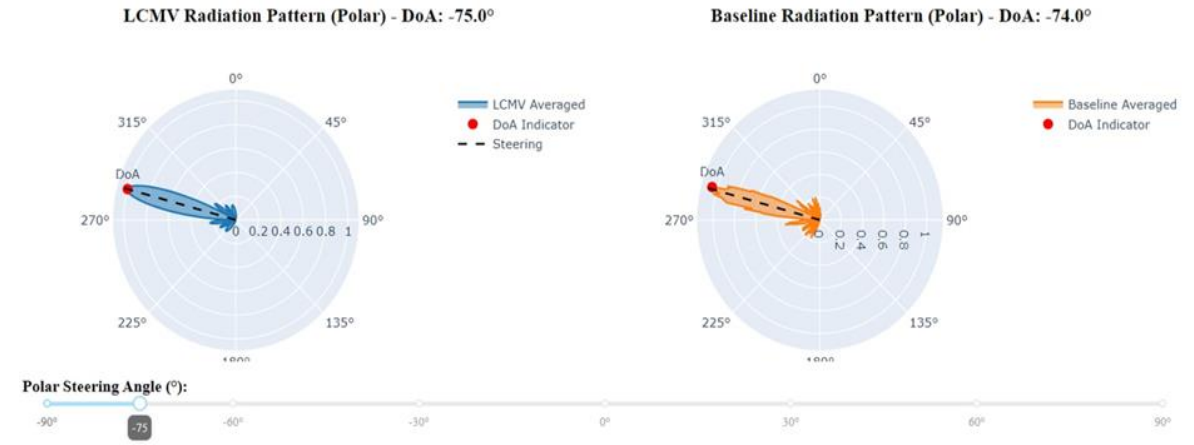

Figure 7. *Unnormalized Radiation Pattern Distribution.*

The Cartesian (azimuth) plot illustrates the beamforming response across horizontal angles, highlighting the main lobes, nulls, and sidelobes, with performance metrics displayed alongside for deeper insight. Figures 8 and 9 compare the normalized power response and raw data captured through the USRP and processed in GNU Radio. Under normalization, both LCMV and DAS share almost identical main lobes, null depth, and side-lobe levels, which yield the same HPVW. However, the LCMV system achieves higher interference suppression (8.48 dB vs 5.65 dB) and WNG (14.09 dB vs 9.04dB). On the other hand, when using raw data, the peaks of these metrics remain unchanged, but DAS exhibits a deeper null (-30dB), whereas LCMV delivers a slightly narrower HPBW of 17.4°, which indicates improved directivity [24]. Overall, the LCMV system outperformed the DAS system in WNG

JTE
HCMUTE
ISSN: 2615-9740

JOURNAL OF TECHNICAL EDUCATION SCIENCE
Ho Chi Minh City University of Technology and Education
Website: https://jte.edu.vn
Email: jte@hcmute.edu.vn

and interference rejection by trading some null depth for a narrower beamwidth, thereby enhancing gain and SNR [25].

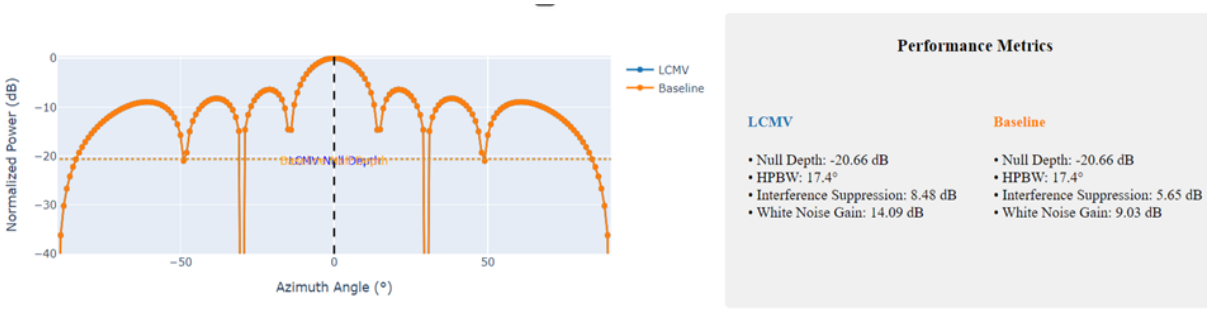

Figure 8. *Normalized Azimuth Pattern with Performance Metrics.*

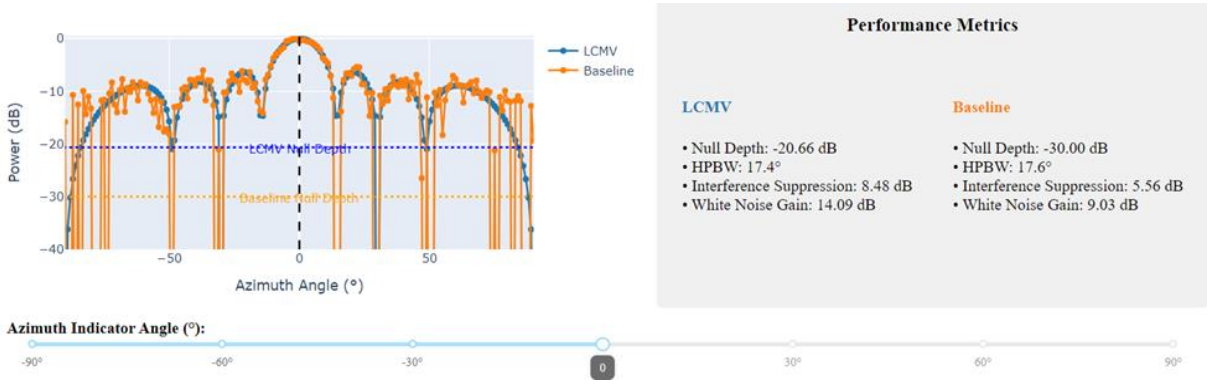

Figure 9. *Cartesian Representation and Evaluation Metrics Using Raw Input Data*.

Table 2 below summarizes the project key achievements by listing the parameters derived from simulation setting or real-time measurements, while Table 3 summarizes the project results. Both of these tables provide the necessary data that will be further analyzed in the discussion section

Table 2. *Implementation Outcomes and Theoretical Comparison*

Parameter	Data Status	Limitation and Theory Comparison
Radiation Pattern	Real-time Data	The measure beamformed output aligns closely with theoretical predictions
White Noise Gain	Simulation Data	The LCMV method yields higher values than the DAS benchmark; however, the gain could be further increased for reasons similar to the Interference Metrics section.
Null Depth	Real-time Data	The returned measurements fall within the expected results range
Interference Suppression	Simulation Data	Since the external visualization tools use constant parameter settings and scaling adjustments that reflect the theoretical data instead of practical results
HPBW	Real-time Data	Narrower beamwidth compared to the DAS system, which matches the mathematical data
DoA	Real-time Data	The DoA results demonstrate that the beam steering operates through simulation, not dynamically.

Table 3. *Performance Metrics Comparison and Conclusions*

Performance Metrics	LCMV	DAS	Conclusion
Time-Domain Signal Stability	Steadier traces with clean, well-defined digital signals (high SINR preserved)	More corrupted by interference; higher amplitude variations	LCMV maintains better signal integrity under noise (Figure 4).
Constellation Clustering	Tighter clusters (lower error vector magnitude, reduced BER)	More dispersed scatter	LCMV suppresses noise and interference more effectively post-BPSK modulation (Figure 5).
Null Depth	-20.66 dB	-30 dB or -20.66 dB in the normalized result	DAS often achieves deeper nulls, but LCMV trades this for a narrower beamwidth and better interference suppression
Half-Power Beam Width	17.4°	17.4°to 17.6°	LCMV achieves a narrower beam for improved directivity (Eq. 6); it enhances SNR and gains in real-time settings.
Interference Suppression	8.48 (dB)	5.65 (dB)	LCMV offers around 2.83 dB better suppression via adaptive covariance updates (Eq. 9);
WNG	14.09 (dB)	9.03 (dB)	LCMV provides approximately 5 dB higher gain with adaptive weighting; aligns with theoretical expectations
Sidelobe Level (dB)	Lower average sidelobes (better suppression in both mode)	Higher sidelobes, more susceptible to interference	LCMV minimizes sidelobe energy adaptively, while reducing unwanted signal pickup compared to DAS.
DoA	Accurate	Fixed Steering	LCMV steering through simulation adapts to maintain DoA fidelity.

4. Discussion

The visual comparison demonstrates that the LCMV yields a smoother time-domain trace, a more compacted constellation cluster, and a more precise radiation pattern and azimuth angle than the DAS method. Furthermore, by characterizing the LCMV beamforming with real-time USRP data, this project offers valuable practical insights into the algorithm capabilities within modern communications systems.

Nevertheless, three main limitations of the project were identified. First, the USRP transmitter is currently not fully utilizing the transmitted RF samples to further refine and optimize the data based on post-beamforming RF samples [26]. Secondly, certain variables, such as the steering angle, number of array elements, and limitation of spatial diversity, are designed with fixed parameters when packaging JSON messages for the external visualization tool and for the flowgraph. This prevents the visualization from reflecting the accurate dynamic behavior of the beamforming algorithms under varying conditions. Finally, the JSON data stream is unable to convey detailed noise environment characteristics from the mixture of data sent by the flowgraph, which makes the calculations have to rely heavily on scaling factors in order to make up for the potential missing data received through the JSON data format.

Further enhancements for the project can be made by implementing pilot-signal feedback for true closed-loop covariance and steering updates [27] by integrating motion or external sensors to adjust array parameters in real time [28] and adopting a hybrid serialization scheme to transfer detailed environmental and performance data to the dashboard [29].

5. Conclusions

In conclusion, this project has successfully implemented an LCMV adaptive beamforming system with real-time signal processing using USRP B210, supported by both the GRC application and

Dashboard Visualization. Furthermore, the LCMV showed enhanced interference mitigation and higher performance metrics while trading off some null depth for a narrower beamwidth. During the implementation, the project encountered limitations in different areas: integrating complex signal processing within the GNU Radio flowgraph, signal tracking, and synchronizing transmission on the USRP devices. These limitations highlight specific avenues for future development. Moving forward, integrating a Digital Twin framework—which enables a real-time, virtual representation of the beamforming system—could help simulate, monitor, and iteratively optimize system performance, ultimately leading to more robust and intelligent adaptive communication systems.

Acknowledgments

This research is funded by the Vietnam Ministry of Education and Training under grant number B2024-VGU-03.

Conflict of Interest

The authors declare no conflict of interest.

Data Availability Statement

The data that support the findings of this study are available from the corresponding author upon reasonable request.

REFERENCES

[1] A. Sârbu, R. Papa, A. Digulescu, and C. Ioana, "A Software-Defined Radio Platform for Teaching Beamforming Principles," *Applied Sciences 2024, Vol. 14, Page 10386*, vol. 14, no. 22, p. 10386, Nov. 2024, doi: 10.3390/APP142210386.
[2] Y. J. Gu, Z. G. Shi, K. S. Chen , and Y. Li, "Robust Adaptive Beamforming for Steering Vector Uncertainties Based on Equivalent DOAs Method," *Progress in Electromagnetics Research*, vol. 79, pp. 277–290, 2008, doi: 10.2528/PIER07102202.
[3] B. Salehi, U. Demir, D. Roy, S. Pradhan, J. Dy, S. Ioannidis, and K. R. Chowdhury, "Multiverse at the Edge: Interacting Real World and Digital Twins for Wireless Beamforming," *IEEE/ACM Transactions on Networking*, vol. 32, no. 4, pp. 3092–3110, 2024, doi: 10.1109/TNET.2024.3377114.
[4] C. A. Balanis, *Antenna Theory: Analysis and Design*, 3rd ed. Hoboken, NJ: Wiley-Interscience, 2005.
[5] Stephen. Few, *Information dashboard design : the effective visual communication of data*. O'Reilly, 2006.
[6] J. Novoa, R. Mahu, A. Díaz, J. Wuth, R. Stern, and N. Becerra Yoma, "Weighted delay-and-sum beamforming guided by visual tracking for human-robot interaction," *arXiv preprint arXiv:1906.07298*, Jun. 2019, doi: 10.48550/arXiv.1906.07298.
[7] J. G. Proakis and M. Salehi, *Digital Communications*, 5th ed., McGraw-Hill, 2008.
[8] GNU Radio, "Noise Source," GNU Radio Wiki, [Online]. Available: https://wiki.gnuradio.org/index.php/Noise_Source. [Accessed: Oct. 2, 2025].
[9] F. M. Gardner, "Interpolation in digital modems—Part I: Fundamentals," IEEE Transactions on Communications, vol. 41, no. 3, pp. 501–507, Mar. 1993, doi: 10.1109/26.221081.
[10] E. R. Tufte, *The Visual Display of Quantitative Information*, 2nd ed. Cheshire, CT: Graphics Press, 2001.
[11] T. Murai and Y. Kagawa, "Electrical impedance computed tomography based on a finite element model," *Proc. 7th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, 1985, doi: 10.1109/tbme.1985.325526.
[12] D. Crockford, "The application/json media type for JavaScript Object Notation (JSON)," *IETF RFC 4627*, Jul. 2006. [Online]. Available: https://www.rfc-editor.org/rfc/rfc4627
[13] E. Vishesh Narendra Pamadi, P. Punit Goel, R. Supervisor, and P. Arpit Jain, "Issue 2 www.jetir.org (ISSN-2349-5162)," JETIR, 2020. [Online]. Available: www.jetir.org.
[14] E. Blossom, "GNU Radio: Tools for exploring the radio frequency spectrum," *Linux Journal*, [Online]. Available: https://www.linuxjournal.com/article/7319. Accessed: Feb. 22, 2025.
[15] Ettus Research, "USRP B210 USB Software Defined Radio (SDR) - Ettus Research, a National Instruments Brand," Accessed: Feb. 22, 2025. [Online]. Available: https://www.ettus.com/all-products/ub210-kit/
[16] A. A. Abdallah and Z. M. Kassas, "Multipath mitigation via synthetic aperture beamforming for indoor and deep urban navigation," IEEE Transactions on Vehicular Technology, vol. 70, no. 9, pp. 8838–8853, Sep. 2021, doi: 10.1109/TVT.2021.3094807.
[17] Microsoft, "Visual Studio Code – Code Editing. Redefined." Accessed: Feb. 22, 2025. [Online]. Available: https://code.visualstudio.com/
[18] J. G. Proakis and Masoud. Salehi, *Digital communications*. McGraw-Hill, 2008.
[19] M. I. Skolnik, *Introduction to Radar Systems*, 3rd ed. New York, NY: McGraw-Hill, 2001.
[20] David. Tse and Pramod. Viswanath, *Fundamentals of wireless communication*. Cambridge University Press, 2008.
[21] S. Haykin and M. Moher, *An Introduction to Analog and Digital Communications*, 2nd ed. Hoboken, NJ: Wiley, 2007.
[22] A. Goldsmith, *Wireless Communications*. Cambridge, U.K.: Cambridge Univ. Press, 2020.
[23] L. C. Godara, "Application of antenna arrays to mobile communications, Part II: Beam-forming and direction-of-arrival considerations," *Proc. IEEE*, vol. 85, no. 8, pp. 1195–1245, Aug. 1997.
[24] G. Hardesty, "Antenna beamwidth," Accessed: Feb. 22, 2025. [Online]. Available: https://www.data-alliance.net/blog/antenna-beamwidth/
[25] C. Pan and J. Chen, "A framework of directional-gain beamforming and a white-noise-gain-controlled solution," *IEEE/ACM Trans. Audio Speech Lang. Process*, 2022, doi: 10.1109/TASLP.2022.3202127.
[26] T. F. Collins, R. Getz, D. Pu, and A. M. Wyglinski, *Software-Defined Radio for Engineers*. Analog Devices Perpetual eBook License – Artech House Copyrighted Material, 2018.

[27] S. A. Hojjatoleslami and J. Ecittler, "Detection of clusters of microcalcification using a k-nearest neighbour classifier," *Proc. 18th Int. Conf. IEEE Engineering in Medicine and Biology Society (EMBC)*, 1996, doi: 10.1049/ic:19960493.
[28] Zetifi Company, "Smart Antennas for Enhanced Connectivity - Zetifi." Accessed: Feb. 23, 2025. [Online]. Available: https://zetifi.com/products/smartantennas/
[29] T. Schenk, *RF Imperfections in High-rate Wireless Systems: Impact and Digital Compensation*, Dordrecht, Netherlands: Springer, 2008, doi: 10.1007/978-1-4020-6903-1.

Khang Nguyen Viet Thai received his Bachelor's degree in Electrical Engineering and Information Technology (EEIT) from the Vietnamese-German University, in collaboration with Frankfurt University of Applied Sciences, in 2025. He is currently pursuing a Master's degree in Communications and Signal Processing at Technische Universität Ilmenau, Germany. His research interests include embedded systems, microwave circuits and modules, and semiconductor technology.
Email: 16142@student.vgu.edu.vn. ORCID: https://orcid.org/0009-0002-8499-7858

Khanh Nguyen Tuan received the B.S. and M.S. degrees in electronic telecommunication from Can Tho University, Can Tho, Vietnam, in 2010 and Ho Chi Minh University of Technology, Ho Chi Minh City, Vietnam, in 2014, respectively. He got the PhD degree in Electronic and Computer Engineering at National Taiwan University of Science and Technology in 2023. He is currently pursuing a postdoctoral degree at the Vietnamese-German University. His research interests include semiconductors, radio-frequency biomedical sensors, noncontact vital-sign radar sensors, and microwave circuits and modules. He was a visiting scientist at the Integrated Electronic Systems (IES) Lab at TU Darmstadt in 2025.
Email: khanh.nt@vgu.edu.vn. ORCID: https://orcid.org/0000-0002-5162-4417

Thuyet Nguyen Vo That received a B.E. in Electronics and Telecommunications from Vietnam National University HCMC - University of Technology in 2009. He got an M.Sc. in Mechatronics and Sensor Systems Technology from the Vietnamese-German University in 2016. He is currently a lab engineer in electronics, digital systems, and signal processing at the Vietnamese-German University. His research interests include RTL design in IP development, FPGA based prototyping and applications.
Email: thuyet.nvt@vgu.edu.vn. ORCID: https://orcid.org/0009-0003-9368-640X