Efficient Doppler Frequency Simulator for Multifrequency

Sukjae Yoon, *Member, IEEE*, Kyoduk Ku, *Member, IEEE*, and Hoyoung Yoo, *Member, IEEE*

***Abstract*—This article introduces an innovative interpolation-based radar simulation system (IRSS) designed to simulate Doppler frequencies across multiple frequencies with minimal hardware complexity. Traditional radar simulation systems, such as Analog Radar System Simulators (ARSSs) and Digital Radar System Simulators (DRSSs), face challenges when supporting multifrequency simulations due to the need for parallel processing of individual Doppler frequencies. The proposed IRSS exploits linear interpolation and the superposition property, enabling a single interpolation process to handle multiple frequency components efficiently. The IRSS structure was implemented using a field programmable gate array (FPGA)-based universal software radio peripheral (USRP), and its performance was evaluated through experimental testing. The results demonstrated that the IRSS accurately generated Doppler frequencies for both single-frequency and multifrequency signals, maintaining consistency with theoretical predictions. The system effectively simulated Doppler shifts for various target speeds while preserving hardware simplicity, unlike traditional simulators that require increased resources proportional to the number of frequencies. This research highlights the advantages of using linear interpolation to reduce hardware complexity and improve scalability in radar simulators. Consequently, the proposed IRSS provides a cost-effective and efficient solution for modern radar systems that demand multifrequency capabilities, making it well-suited for applications in complex environments such as autonomous vehicles, military operations, and aviation.**

***Index Terms*—Doppler frequency, linear interpolation, radar system, universal software radio peripheral (USRP).**

I. Introduction

RADAR systems are technologies that detect and track the position and speed of objects using electromagnetic waves, and they are applied in various fields such as military, aviation, meteorology, and maritime traffic control [1]. In the defense industry, they are used for detecting aircraft and missiles [2], [3], in air traffic control, for tracking aircraft [4], in meteorology, for predicting precipitation and storms [5], [6], and in maritime applications, for preventing ship collisions [7]. Recently, radar systems have also been applied to autonomous vehicles, smart homes, and industry [8], [9], and their importance is expanding across various industries. Radar systems measure the position of an object through time delay and its speed using the Doppler effect. The change in Doppler frequency occurs according to the relative speed of the moving object, allowing for the calculation of the velocity of the object.

However, these technologies are significantly affected by various environmental conditions, making verification in real environments challenging and expensive [10], [11], [12]. Particularly in high-risk environments such as the military and aviation sectors, recreating actual scenarios for testing is not only limited but also constrained by time and resources [13], [14]. To address these issues, the need for radar system simulators has emerged. Simulators provide a testing environment that can virtually reproduce various environments and scenarios, allowing for the realistic simulation of real-world conditions [15]. Recently, radar system simulators have become essential for ensuring reliability and stability before actual implementation. They enable precise performance evaluation of radar systems under different environmental variables and conditions [11], [12], and they allow the simulation of various functions, including velocity measurement using Doppler frequencies, from the design stage, enabling early detection and improvement of potential issues [16], [17], [18], [19], [20].

In this article, we propose a method to simulate Doppler frequency using DSP algorithms. Specifically, the proposed simulator applies linear interpolation to the frequency samples fed into the system, adjusting the amplitude of each sample to modify the output signal frequency. The simulated radar signal is determined by the input frequency and Doppler frequency, which vary according to the relative speed between the radar and the target. We can present a method to vary the Doppler frequency by changing the linear interpolation ratio. Additionally, the need for multifrequency has become increasingly emphasized to optimize radar system performance and enhance reliability [21], [22], [23], [24], [25] in complex environments such as urban multipath environments [9], ground clutter scenarios [12], [26], and intentional jamming

Received 27 November 2024; revised 16 June 2025; accepted 14 August 2025. Date of publication 4 September 2025; date of current version 16 September 2025. This work was supported in part by the National Research Foundation of Korea (NRF) grant funded by Korean Government Ministry of Science and ICT (MSIT) under Grant 2022R1A5A8026986; in part by the Institute of Information and Communications Technology Planning and Evaluation (IITP) grant funded by Korean Government (MSIT) under Grant 2022-0-01170; in part by the National Research and Development Program through the National Research Foundation of Korea (NRF) funded by MSIT under Grant 2020M3H2A1078119; and in part by MSIT, South Korea, through the Information Technology Research Center (ITRC) Support Program Supervised by IITP under Grant IITP-2024-RS-2024-00436406. The Associate Editor coordinating the review process was Dr. Gang Yu. *(Corresponding author: Hoyoung Yoo.)*

Sukjae Yoon is with Duksan Navcours, Daejeon 34014, South Korea, and also with the Department of Electrical Engineering, Chungnam National University, Daejeon 34134, South Korea (e-mail: craz0309@gmail.com).

Kyoduk Ku and Hoyoung Yoo are with the Department of Electrical Engineering, Chungnam National University, Daejeon 34134, South Korea (e-mail: gdku.cas@gmail.com; hyyoo@cnu.ac.kr).

Digital Object Identifier 10.1109/TIM.2025.3606061

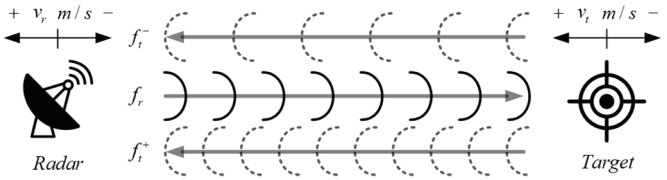

Fig. 1. Basic radar system based on the Doppler effect.

interference [10], [23]. The previous Doppler simulators target only single frequencies, resulting in an increase in complexity proportional to the number of supported frequencies when simulating multiple signals. This article proposes a simulator that supports multifrequency using the superposition property of signals and a single interpolation process. This structure minimizes the increase in complexity, allowing multifrequency simulation through a single hardware platform. The structure of this article is as follows. Section II covers the background on Doppler frequency and previous simulators necessary to understand this article. Section III explains the proposed interpolation-based simulator structure. Section IV presents the implementation of the proposed simulator using a field programmable gate array (FPGA)-based universal software radio peripheral (USRP) and discusses the experimental results. Finally, Section V concludes this article.

II. Background

The Doppler effect refers to the phenomenon where the frequency of a signal transmitted by a radar changes when it is reflected off a target and returns to the radar, due to the relative movement between the radar and the target [28]. When the target approaches the radar, the wavelength of the reflected signal shortens, increasing the frequency of the received signal. Conversely, when the target moves away from the radar, the wavelength of the reflected signal lengthens, decreasing the frequency. Figure 1 illustrates the Doppler effect in a basic radar system based on the radar and the target [28]. In Fig. 1, the frequency of the signal transmitted by the radar is denoted as f_r, and the frequency of the signal reflected by the target is denoted as f_t. When the target moves closer to the radar, it is represented as f_t^+, and when it moves away, it is represented as f_t^-. According to [28], the frequency f_t reflected by the target consists of the component f_r transmitted by the radar and the Doppler frequency component f_d caused by the relative movement of the target. More precisely, the frequency f_t, reflected by the target, is determined by the transmitted f_r and the relative speed v of the target is shown below

$$f_t = f_r + f_d = \left(\frac{c+v}{c-v}\right) \times f_r \tag{1}$$

where c is the speed of light, 3×10^8 m/s. In radar systems, the Doppler frequency is calculated based on the relative speed $v = v_t - v_r$ between the radar and the target. Typically, the radar is stationary, so v is usually equal to v_t. Furthermore, by rearranging the equation, it can be expressed as

$$f_t = \left(\frac{c-v+2v}{c-v}\right) \times f_r = \left(1+\frac{2v}{c-v}\right) \times f_r = f_r + \left(\frac{2v}{c-v}\right) \times f_r. \tag{2}$$

TABLE I
Example of Doppler Frequency [Hz]

v \ f_r	800 MHz	900 MHz	1 GHz	1.1 GHz	1.2 GHz
± **100** [*m/s*]	± 534	± 600	± 667	± 734	± 801
± **250** [*m/s*]	± 1,334	± 1,501	± 1,668	± 1,835	± 2,001
± **500** [*m/s*]	± 2,669	± 3,002	± 3,336	± 3,669	± 4,003
± **750** [*m/s*]	± 4,003	± 4,503	± 5,003	± 5,504	± 6,004
± **1,000** [*m/s*]	± 5,337	± 6,004	± 6,671	± 7,338	± 8,006

Using (1) and (2), the Doppler frequency can be expressed as

$$f_d = \frac{2v}{c-v} f_r. \tag{3}$$

Additionally, if $c \gg v$, (3) can be simplified as

$$f_d = \frac{2v}{c} f_r. \tag{4}$$

As a result, a radar simulator is a system that generates the frequency f_t to the transmitted frequency f_r by adding the Doppler frequency f_d based on (1) through (4). For example, Table I shows the various Doppler frequencies generated based on the frequency f_r and the relative speed v. The f_r ranges from 800 MHz to 1.2 GHz, and the Doppler frequencies are shown for relative speeds ranging from ±100 m/s to ±1 km/s. For instance, a 1 GHz radar carrier frequency with a target moving at 500 m/s induces a Doppler frequency of 3336 Hz, resulting in a reflected signal of 1000 003 336 Hz.

Building a radar simulation system to model Doppler effect based on target movement is significantly influenced by various environmental conditions, making it challenging and costly to verify in real-world settings [10], [11], [12], [16], [18], [19], [20], [27], [28], [29], [30], [31], [32], [33], [34]. Especially in high-risk areas like the military and aviation sectors, recreating real-life scenarios for testing is not only limited but also extremely dangerous. To address these challenges, radar system simulators have been actively developed, including analog radar system simulator (ARSS) [29], [30] and digital radar system simulator (DRSS) [27], [31], [32], [33], [34], [35]. ARSS was one of the first hardware-based radar system simulators [29], [30]. As shown in Fig. 2(a), the block diagram of the ARSS [29], [30] includes components like the Doppler calculator, Doppler generator, local oscillator (LO), and mixer. Initially, the Doppler calculator uses (4) to compute f_d by taking f_r and v as inputs. The Doppler generator then takes the calculated Doppler frequency f_d and produces a signal $\tilde{f}_d$ corresponding to this frequency. Here, f represents the value of the frequency, while $\tilde{f}$ denotes the wave signal associated with that frequency. $\tilde{f}_r$ can either be provided as an external input or generated by the LO. Finally, the mixer produces the $\tilde{f}_t$ signal, simulating the reflection from the target. Based on (1)–(4), the ARSS [29], [30] can create various Doppler frequencies based on the given radar frequency f_r and the relative speed v. Although ARSS [29], [30] successfully generated various Doppler frequencies, it encountered difficulties in producing high-frequency signals due to the bandwidth limitations of analog IC chips. To overcome this,

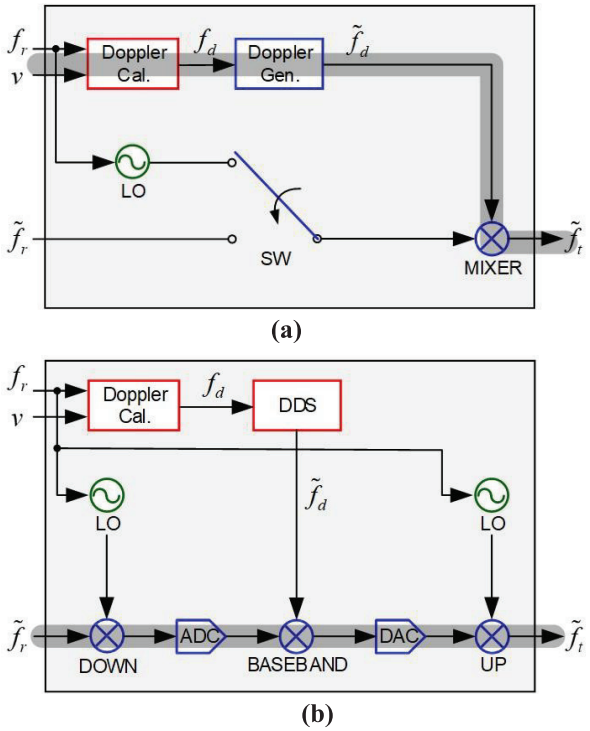

Fig. 2. Previous Serial ARSS [29], [30] and DRSS [27], [31], [32], [33], [34], [35] for single frequency: (a) serial ARSS and (b) serial DRSS.

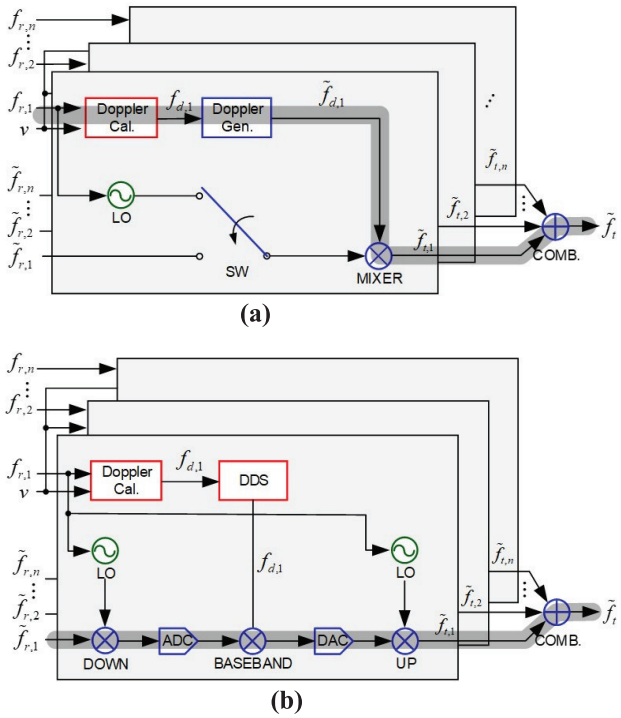

Fig. 3. Previous Parallel ARSS [29], [30] and DRSS [27], [31], [32], [33], [34], [35] for multifrequency: (a) parallel ARSS and (b) parallel DRSS.

DRSS was developed [27], [31], [32], [33], [34], [35], which downconverts RF signals to IF or baseband signals, enabling Doppler frequency simulation at the digital level. Fig. 2(b) shows the block diagram of a previous DRSS [27], [31], [32], [33], [34], [35]. Like ARSS [29], [30], it calculates the Doppler frequency f_d using f_r and speed v through a Doppler calculator. However, unlike ARSS [29], [30], DRSS [27], [31], [32], [33], [34], [35] downconverts the $\tilde{f}_r$ signal waveform, mixes it with $\tilde{f}_d$, and then performs upconversion. Additionally, the DRSS [27], [31], [32], [33], [34], [35] uses a Direct Digital Synthesizer (DDS) for Doppler generation, whereas ARSS [29], [30] directly generates the analog $\tilde{f}_d$. As a result, the DRSS [27], [31], [32], [33], [34], [35] generates a $\tilde{f}_t$ signal at the digital level that simulates reflections from a target based on the given relative speed v. Compared to ARSS [29], [30], which directly generates high-frequency bands, DRSS [27], [31], [32], [33], [34], [35] processes baseband or IF signals, allowing it to handle frequency ranges that ARSS [29], [30] cannot manage. To facilitate system-level hardware analysis further, the block diagrams employ color-coded lines to distinguish between component domains: red lines represent digital components, blue lines correspond to analog components, and green lines indicate RF components. In addition, the longest signal path is emphasized using bold and shaded lines to highlight potential latency-critical routes. This convention is applied uniformly across all diagrams.

Recently, the need for multifrequency capability has become more prominent to optimize radar system performance and enhance reliability [16], [18], [19], [26], [36], [37] in complex environments such as urban multipath environments [9], ground clutter scenarios [12], [26], and intentional jamming interference [10], [23]. Utilizing multiple frequencies simultaneously overcomes the limitations of single-frequency systems, allowing for more precise detection and tracking [16], [37]. For example, techniques such as frequency hopping [26], [27], [28], which improve radar performance by varying the transmission frequency over short intervals, or frequency-modulated continuous wave (FMCW) [16], [21], [37], [38], which transmit signals by continuously modulating the frequency over time, demonstrate the advantages of multifrequency approaches. Additionally, multifrequency systems play a crucial role in improving reliability in cluttered or interference-prone environments [21]. Using different frequencies, it is possible to cancel out or avoid interference and noise that may occur in specific frequency bands [21], [36], thereby enhancing the overall signal processing performance of the system. However, the previous ARSS [29], [30] and DRSS [27], [31], [32], [33], [34], [35] structures used analog-based ICs or digital baseband signal processing to simulate single-frequency Doppler shifts. To handle multiple frequencies, it is necessary to individually generate the Doppler shift for each frequency according to (1)–(4). Therefore, in the previous structures, simulating multiple Doppler frequencies requires parallel expansion of the single Doppler frequency simulation structure, resulting in additional hardware resources proportional to the number of frequencies supported. Fig. 3 illustrates a system based on ARSS [29], [30] and DRSS [27], [31], [32], [33], [34], [35] designed to simulate multiple frequency signals. As shown in Fig. 3, each independent Doppler frequency $f_{d,i}$ $(1 \le i \le n)$ is generated and combined to produce the target frequency $f_{t,i}$ $(1 \le i \le n)$ for each respective frequency. The previous f in Fig. 2 is extended to

TABLE II
SUMMARY OF VARIOUS INTERPOLATION METHODS

Method	Equation	(#ADD/SUB)*	(#MUL/DIV)*
Linear	$A(t)=A_k+(A_{k+1}-A_k)(t-t_k)/(t_{k+1}-t_k)$	3	1
Polynomial (Lagrange)	$A(t)=\sum_{j=0}^{n} A_j L_j(t),\ L_j(t)=\prod_{0\le m\le n, m\ne j}(t-t_m)/(t_j-t_m)$	$n(n+1)$	n^2
Spline	$A_k(t)=a_k+b_k(t-t_k)+c_k(t-t_k)^2+d_k(t-t_k)^3$	$2n+3$	$2n+3$

*Approximate numbers

f_i ($1 \le i \le n$) in Fig. 3, requiring an increase in complexity by n times and additional combiner hardware resources. Finally, these frequencies are combined to produce the final f_t.

III. PROPOSED METHOD

As multifrequency applications expand in radar systems, the previous radar simulators limited to single frequencies face increased hardware complexity with each additional frequency. To address this, this article applies an interpolation technique to simulate Doppler frequencies and proposes a structure that utilizes signal superposition properties, which enables a single interpolation to simulate Doppler frequencies across multiple frequencies.

A. Linear Interpolation

Interpolation is a widely used technique for estimating intermediate values between sampled data points. Typical methods include linear interpolation [41], polynomial interpolation [39], and spline interpolation [42]. Linear interpolation offers simplicity and low computational cost by connecting adjacent points with straight lines, making it suitable for real-time applications. Polynomial and spline interpolations improve accuracy by fitting higher order or piecewise polynomials, but they incur significantly higher hardware complexity, especially as the polynomial degree increases. Table II summarizes the equations and arithmetic operation counts for each method. Even at a low degree of $n = 3$, polynomial and spline methods require substantially more operations than linear interpolation. Given the need for low-latency and resource-efficient implementation, the proposed Interpolation-based Radar Simulation System (IRSS) adopts linear interpolation to simplify the architecture and enable real-time processing. Furthermore, linear interpolation techniques in signal processing are generally classified into amplitude, frequency, and phase interpolation. Amplitude interpolation directly estimates signal amplitudes and is both simple and efficient. Frequency interpolation smooths instantaneous frequency variations, while phase interpolation maintains phase continuity. The proposed system simulates radar signals by adjusting amplitudes according to target velocity. To meet real-time constraints with minimal hardware overhead, amplitude interpolation is employed. While it may introduce phase discontinuities and minor amplitude distortion, these effects are acceptable in our simulation context. Compared to frequency or phase interpolation, the adopted method offers a favorable balance between accuracy and efficiency.

Algorithm 1 Proposed Interpolation

Input: SampleS_i, before interpolation
Output: SampleS_o, after interpolation

```
    // initialization
1:  i = 0; i' = 0; d = 0; w = α;
    // main loop
2:  for (i = 0; i < L; i++; i'++)
3:    d = S_i[i']−S_i[i − 1]
      // discrepancy between input samples
4:    S_o[i] = S_i[i'] +d · w
      // linear interpolation
5:    if (|w| < 1)
        // weight update
6:      w = w + α;
7:    else
        // weight reset & index adjust
8:      w = α;
9:      i'= i' +sign(α)
10:   end if
11: end for
```

Fig. 4 illustrates the digital signal processing steps for the interpolation method proposed in this article. In Fig. 4, the black samples represent the input samples $S_i[i]$ ($0 \le i \le L$), while the blue samples represent the output samples $S_o[i]$ obtained through linear interpolation. In this article, the interpolation is achieved by adding a compensation value to adjacent input samples. The compensation value is calculated by multiplying the difference d between input samples by a weight w, which itself is determined by accumulating the interpolation ratio α. If the absolute value of weight w exceeds 1, indicating that the interpolation magnitude surpasses the sample period, the input index i is determined by the sign of α. If α is positive, the algorithm skips the next sample; if α is negative, the current sample is reused once more. Subsequently, w is reset to α. When the interpolated values based on the ratio α are applied to all samples of the sampled signal, the signal's period is effectively changed in proportion to α. Fig. 4(a) shows a case with a positive α, where the frequency increases via linear interpolation, while

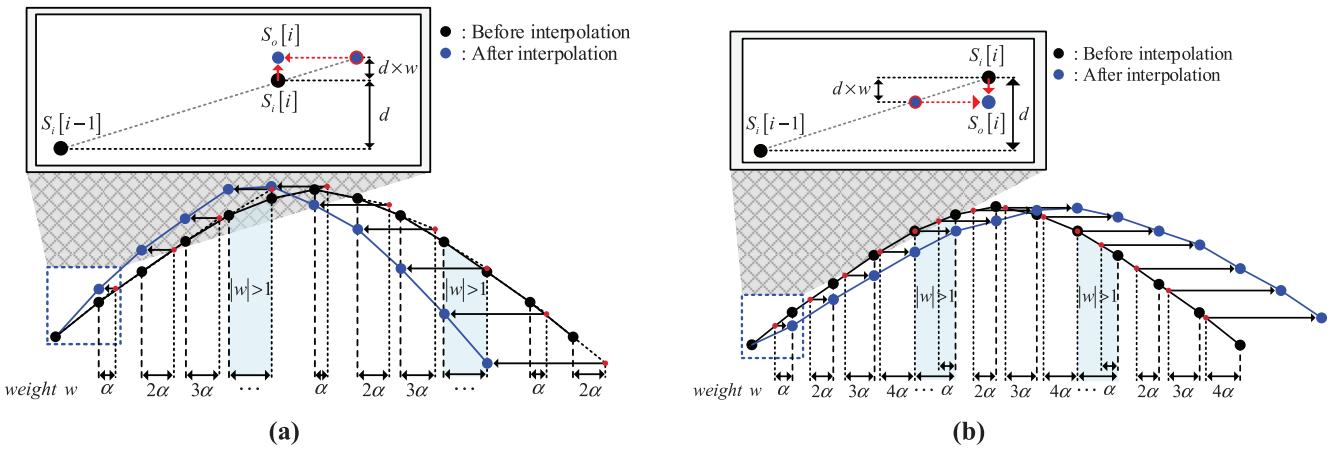

Fig. 4. Concept of linear interpolation: (a) positive α and (b) negative α.

Fig. 4(b) shows a case with a negative α, indicating a decrease in frequency. The linear interpolation process described above can be summarized in Algorithm 1. As shown in Line 3, the discrepancy for a given input sample $S_i[i]$ $(0 \le i \le L)$ is calculated. Then, as described in Line 4, the compensation value is added to perform linear interpolation, resulting in single subtraction, addition, and multiplication. Note that the proposed method calculates the compensation value by multiplying the discrepancy and the weight. Ordinarily, the weight is computed by accumulating the interpolation ratio α as described in Line 6. If the absolute value of the weight exceeds 1, it is considered to have surpassed the sample, the weight is reset, and the index is adjusted as described in Lines 8 and 9. This process is applied to all input samples to calculate the output samples S_o.

According to Fig. 4 and Algorithm 1, frequency variation can be achieved by interpolating the input signal samples. More precisely, when the input signal sample $S_i[i]$ has a frequency f_i, and the interpolated output signal sample $S_o[i]$ has a frequency f_o scaled by α, f_o and f_i satisfy the following relation as:

$$f_o = (1 + \alpha) f_i. \tag{5}$$

To provide examples, Fig. 5(a) demonstrates the results when Algorithm 1 is applied to an input frequency f_i of 1 Hz, with the set to 1/4, 2/4, and 3/4, and a sampling frequency designated as 20 Hz. According to (5), the frequency changes to 1.25, 1.5, and 1.75 Hz, respectively. Similarly, Fig. 5(b) shows the results for $\alpha = -1/4, -2/4$, and $-3/4$ with the same input, where the frequency changes accordingly to 0.75, 0.5, and 0.25 Hz, as calculated by (5). In conclusion, by adjusting the interpolation ratio α in (5), the desired frequency of a given signal can be achieved through linear interpolation. Applying this approach in a radar simulator system enables the generation of signals with Doppler frequencies by tuning α. By substituting $2v/c$ for α in (1) –(4) and using f_r as f_i, f_o can be computed as f_t.

Furthermore, this interpolation property preserves the characteristics of (5) even for multifrequency signals with mixed components. In other words, frequency modification can be applied to signals expressed as linear combinations through

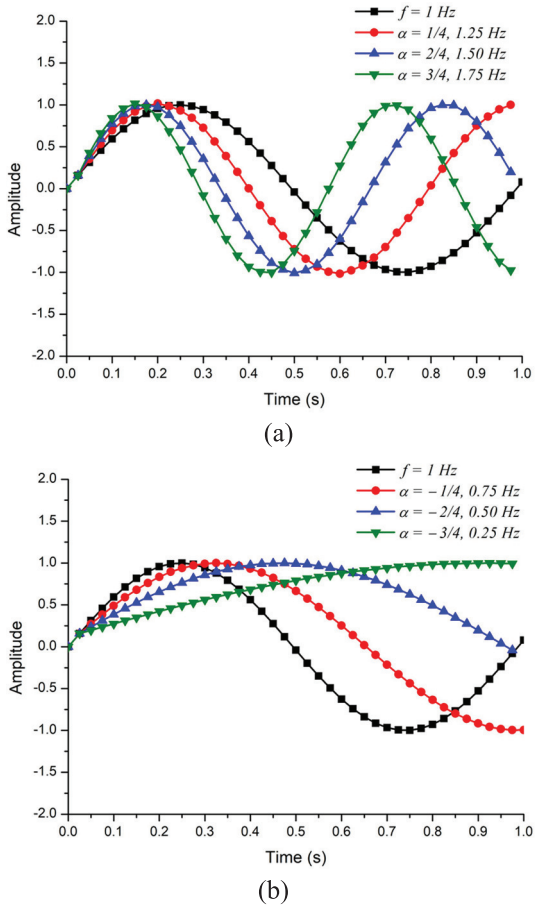

Fig. 5. Example of single frequency: (a) positive α and (b) negative α.

interpolation, while maintaining the superposition property. Fig. 6 demonstrates that linear interpolation satisfies the superposition property. In detail, Fig. 6(a) and (b) show signals with $f_{r,1} = 1$ Hz and $f_{r,2} = 1.5$ Hz, respectively, while Fig. 6(c) displays the combined signal $f_r = f_{r,1} + f_{r,2}$. Fig. 6(d), (e), and (f) show the results of applying interpolation with $\alpha = 1/4$ to $f_{r,1}$, $f_{r,2}$ and f_r.

According to Fig. 6, the interpolation result for the linearly combined signal in Fig. 6(c) is the same as that shown in

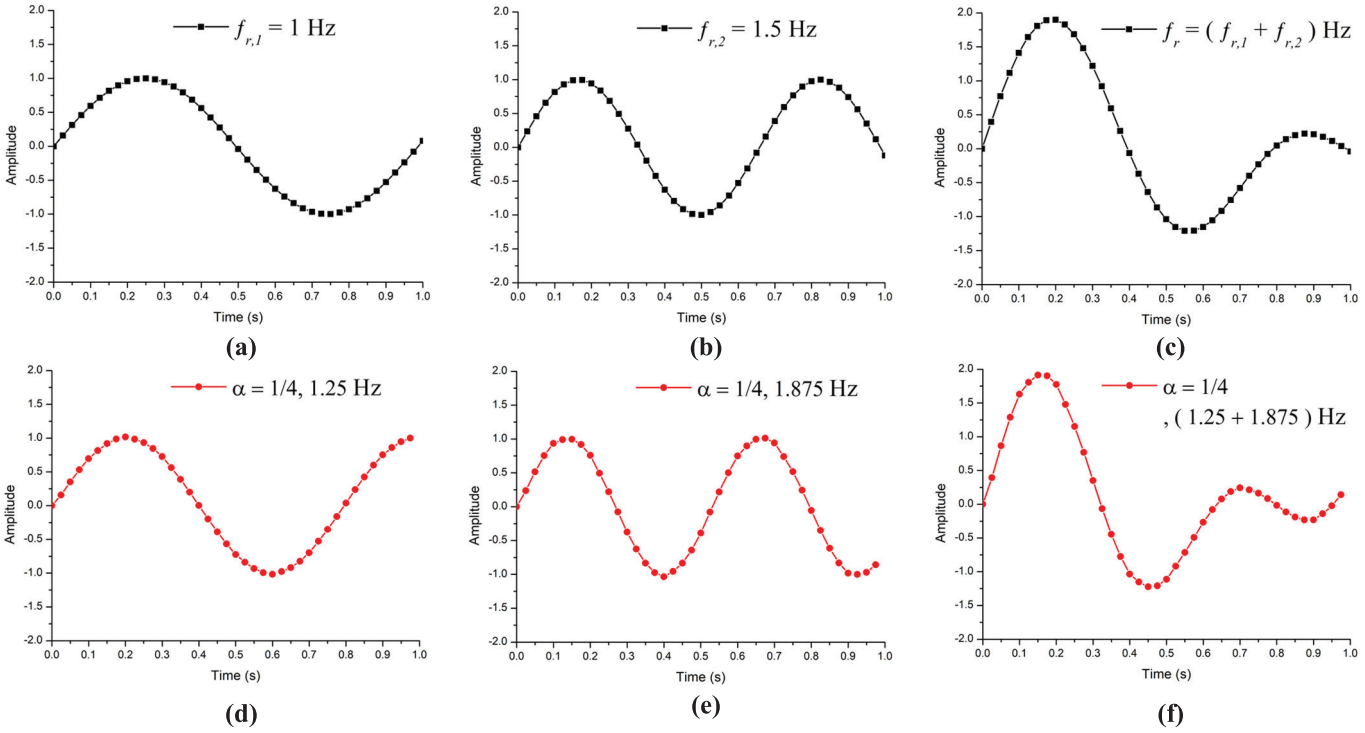

Fig. 6. Example of multifrequency. (a) $f_{r,1}$, (b) $f_{r,2}$, (c) $f_r = f_{r,1} + f_{r,2}$, (d) $\alpha f_{r,1}$, (e) $\alpha f_{r,2}$, and (f) $\alpha f_r = \alpha f_{r,1} + \alpha f_{r,2}$.

Fig. 6(f), and Fig. 6(f) matches the result obtained by combining Fig. 6(d) and (e). It is important to note that Fig. 6(f) can be obtained by a single interpolation on the combined signal without the need to interpolate each signal separately and then combine them. Consequently, this linear interpolation property allows the simulation of Doppler frequencies by performing a single interpolation on the combined signal, eliminating the need to generate individual Doppler frequencies for each frequency, as required in conventional ARSS [29], [30] and DRSS [27], [31], [32], [33], [34], [35] systems.

B. Interpolation-Based Radar Simulation System

The proposed IRSS exploits two key signal processing properties. One is the ability to change signal frequency using linear interpolation, and the other is the property that interpolation on a combined signal produces the same result as applying interpolation to each signal separately and then combining them. The overall steps of the proposed IRSS are summarized as follows.

Step 1: Downconvert by f_c.

Step 2: Linear interpolation by α.

Step 3: Upconvert by $(1 + \alpha) f_c$.

Fig. 7 illustrates the overall operation of the proposed IRSS in the frequency domain. In the first step, the radar-transmitted signal containing multifrequency components $f_r = f_{r,1} + f_{r,2} + \cdots + f_{r,n}$ is downconverted to the baseband. For simplicity, it is assumed that the radar-transmitted signal has dual-frequency components as $f_r = f_{r,1} + f_{r,2}$. While individual frequencies can be downconverted using their respective frequencies in single-frequency cases, for a signal with multifrequency components, an appropriate center frequency f_c is chosen for downconversion. Note that we denote the difference between ith frequency component and the center frequency as $f_{\Delta,i} = f_{r,i} - f_c$. In the second step, linear interpolation is applied to the downconverted $f\Delta$, i signal in the baseband by a factor of α. When linear interpolation is performed by α times as per Algorithm 1, the output frequency can include Doppler components for $f\Delta$, i added proportionally by α, as shown in (5). In the third step, the signal in the baseband is upconverted by $(1 + \alpha) f_c$. As a result, the final output signal f_t is obtained. In previous DRSS systems, upconversion and downconversion are performed using the same f_c frequency. However, in the proposed structure, upconversion considers the Doppler frequency component by incorporating a factor of $(1 + \alpha)$. After performing the three steps with the combined signal f_r and its individual frequency components $f_{r,i}$, the proposed IRSS generates the final output as

$$\begin{aligned}\sum_{i=1}^{n}\left(f_{t,i}\right) &= \sum_{i=1}^{n}\left((1+\alpha)\, f_{\Delta,i} + (1+\alpha)\, f_c\right)\\ &= \sum_{i=1}^{n}\left((1+\alpha)\left(f_{\Delta,i} + f_c\right)\right)\\ &= \sum_{i=1}^{n}\left((1+\alpha)\, f_{r,i}\right). \end{aligned} \tag{6}$$

It is important to note that the proposed IRSS generates signals with each frequency component combined with its Doppler frequency using a single interpolation.

Finally, the proposed structure is illustrated in Fig. 8. It takes a signal $\tilde{f_r}$ containing multiple frequencies as input, along with the target speed v and center frequency f_c. The signal is first downconverted by f_c using a LO to convert it

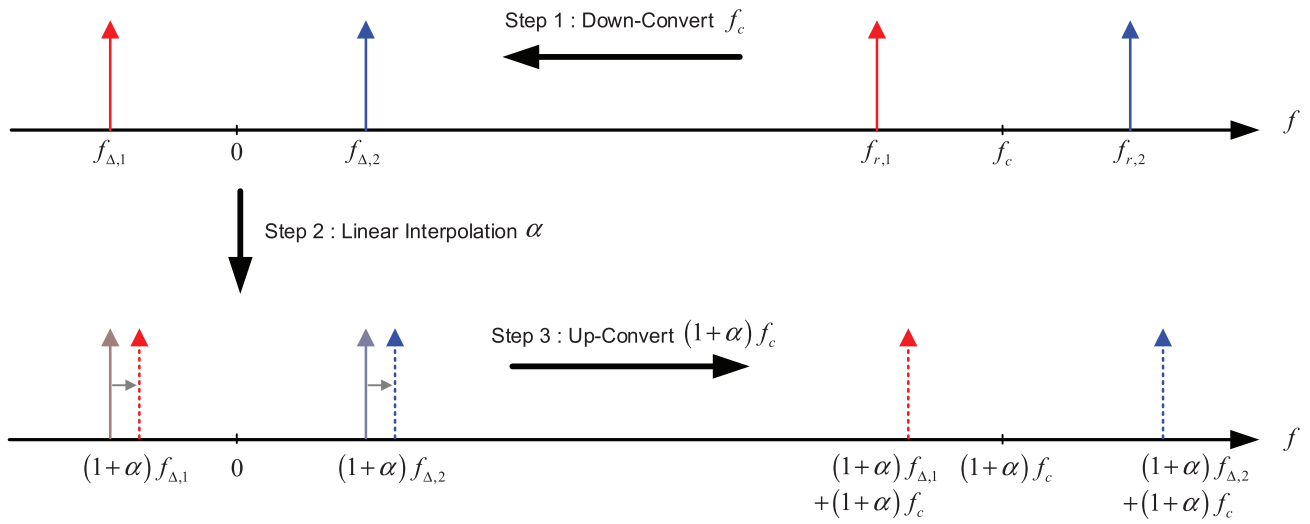

Fig. 7. Frequency model of the proposed system.

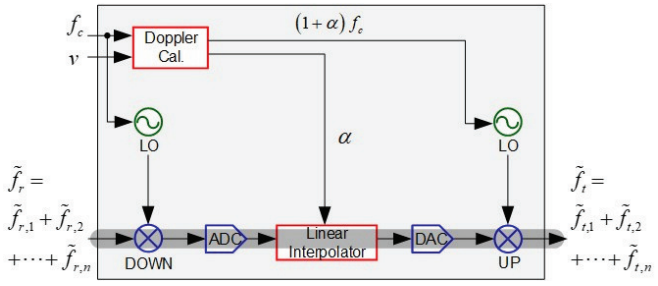

Fig. 8. Proposed interpolation-based radar system simulator.

TABLE III
FPGA IMPLEMENTATION RESULTS OF THE PROPOSED IRSS

Resource	Total	Used	Utilization (%)
#LUT	254,200	49,264	19.4
#Register	508,400	78,722	15.5
#Block RAM	795	125	15.7
#DSP	1,540	276	17.9
Power (W)	-	7,468	-
Throughput (MB/s)	-	800	-

to a baseband signal. Next, linear interpolation is performed according to Algorithm 1, dependent on the target speed v. In other words, the α value is calculated as $2v/c$, and it can be adjusted for different speeds of targets, allowing simulations for various scenarios. Finally, the signal is upconverted by $(1+\alpha)f_c$ to calculate the final f_t, which includes all frequency components from 1 to n, as shown in (6). The proposed system enables the simulation of Doppler variations for a given multifrequency signal according to the desired scenario. Notably, compared to the previous ARSS [29], [30] in Fig. 3(a) and DRSS [27], [31], [32], [33], [34], [35] in Fig. 3(b) by utilizing the upper position property, the system reduces hardware complexity by simulating the signal with a single interpolation, without needing to apply interpolation to each individual signal component.

IV. EXPERIMENTAL RESULTS

To verify the proposed IRSS, a demo environment was set up for testing, as shown in Fig. 9. The signal generator produced an RF signal $\tilde{f}_r$ containing various signal components, which was then fed into the USRP. Given the RF signal, the USRP converted the received RF signal to the IF band using the set frequency f_c, then sampled it with the analog-to-digital converter (ADC). The linear interpolator calculated the α ratio based on the set speed v and performed interpolation. The signal was then converted to an analog signal by the digital-to-analog converter (DAC) and upconverted by $(1+\alpha)f_c$ to generate the final output signal $\tilde{f}_t$. Finally, the generated signal was sent to a spectrum analyzer to verify whether it matched the doppler frequency corresponding to speed v for the input signal $\tilde{f}_r$ from the signal generator. More precisely, the proposed IRSS was implemented on National Instruments USRP 2954-R. Typically, a USRP is composed of an RF section and an IF section. For the RF section, the amplifier (AMP), LO, ADC, DAC, and filter were adjusted to operate in our test environment, and the IF section was designed in the FPGA to perform linear interpolation. In addition, Table III summarizes the resource utilization of the Xilinx Kintex-7 XC7K410T FPGA, detailing the usage of LUTs, registers, block RAMs, DSP slices, and the overall throughput and power consumption. The experimental results demonstrate that the proposed IRSS occupies less than 20% of the available hardware resources, with total power consumption remaining below 7.5 W. The ADC and DAC handle I/Q data with 16-bit resolution, and a measured throughput of 800 MB/s is achieved based on a 200 MS/s sampling rate. These results confirm the suitability of the proposed structure for lightweight, real-time radar simulation.

Fig. 10 shows the results of measuring the USRP output with a spectrum analyzer when a single-frequency signal of 1 GHz was generated by the signal generator and the

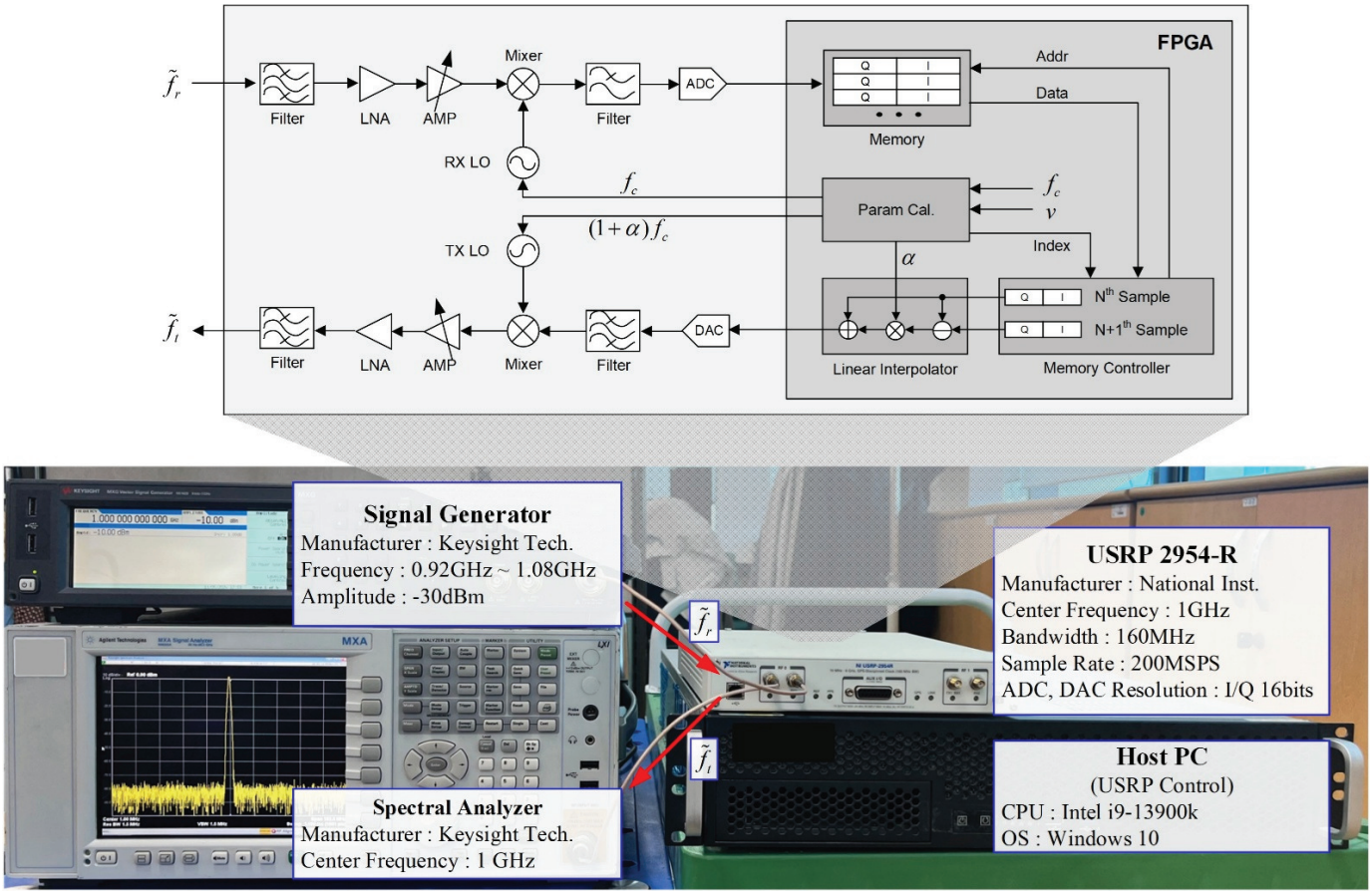

Fig. 9. Prototype system based on the proposed IRSS.

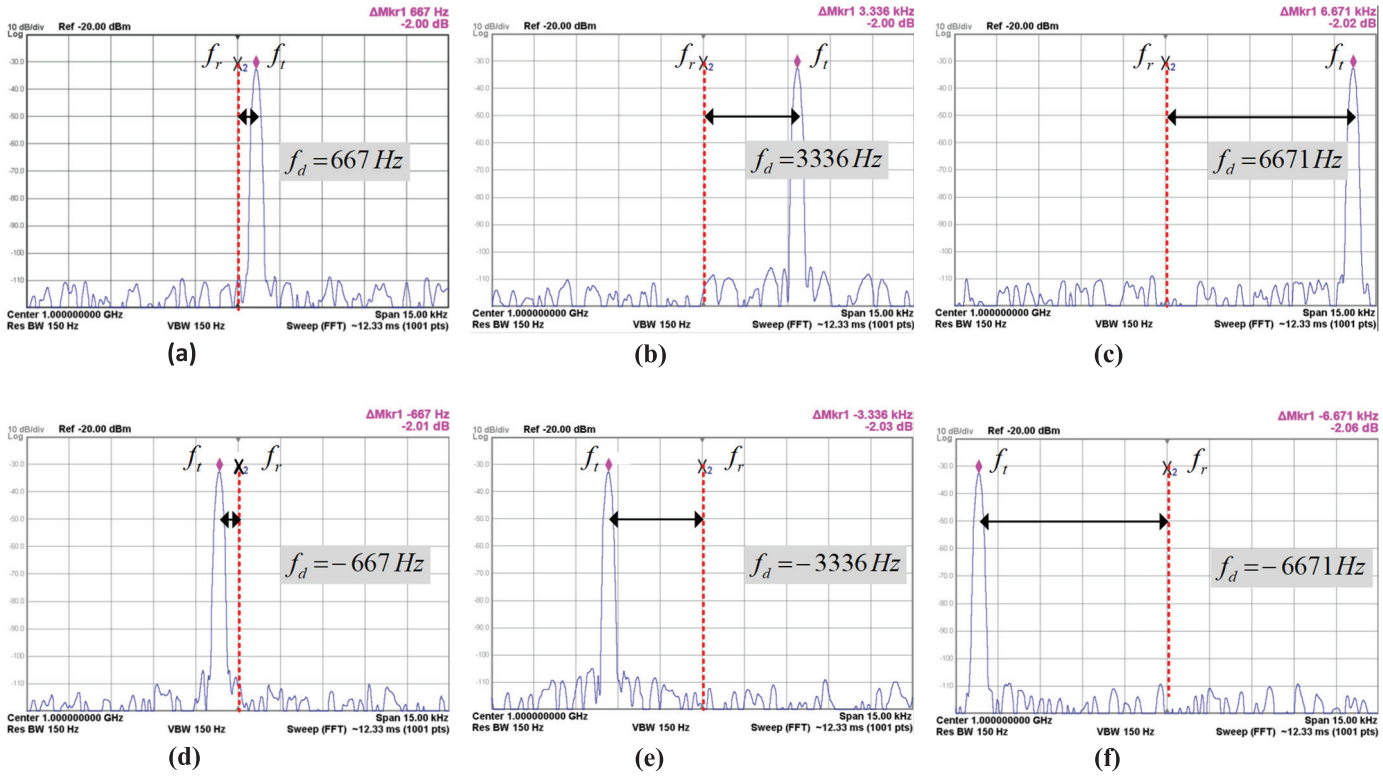

Fig. 10. Single frequency results: (a) +100 m/s, (b) +500 m/s, (c) +1000 m/s, (d) −100 m/s, (e) −500 m/s, and (f) −1000 m/s.

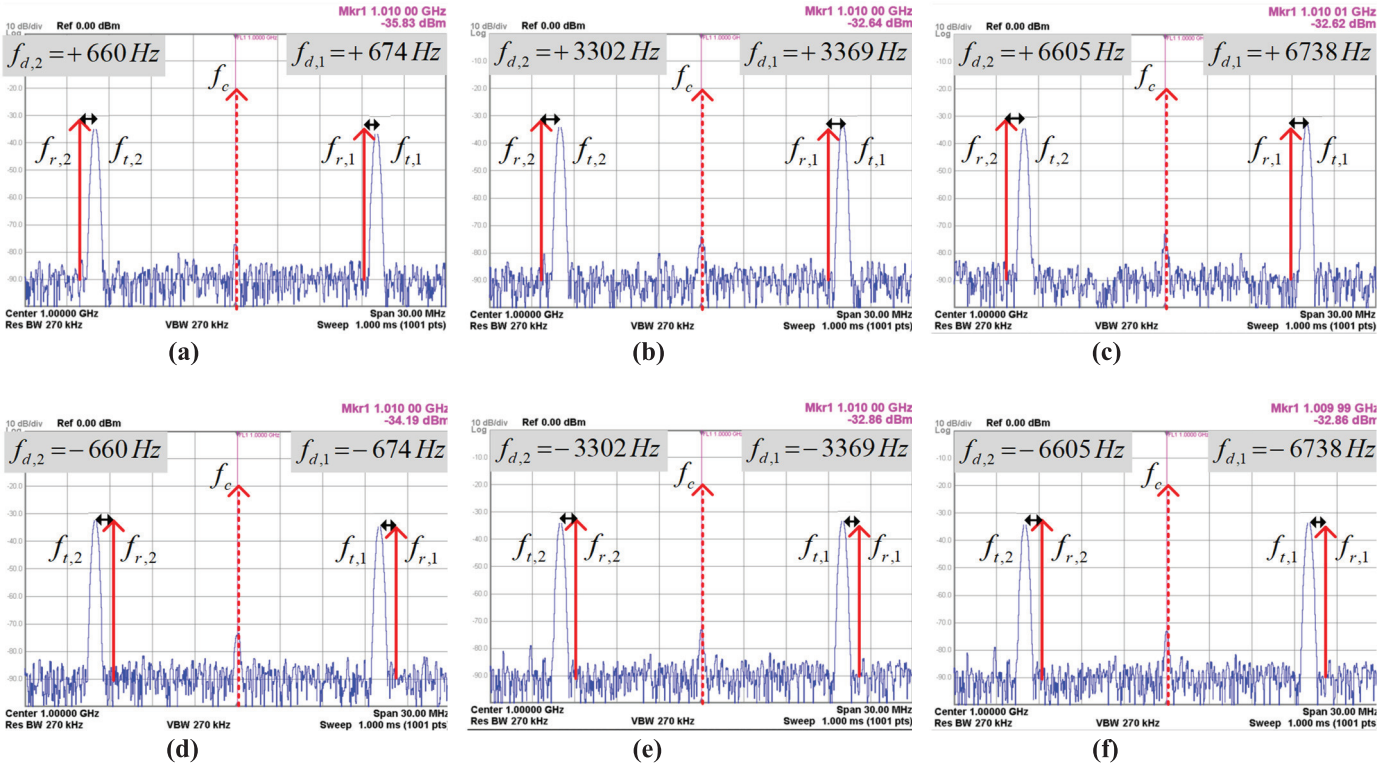

Fig. 11. Multifrequency results: (a) +100 m/s, (b) +500 m/s, (c) +1000 m/s, (d) −100 m/s, (e) −500 m/s, and (f) −1000 m/s.

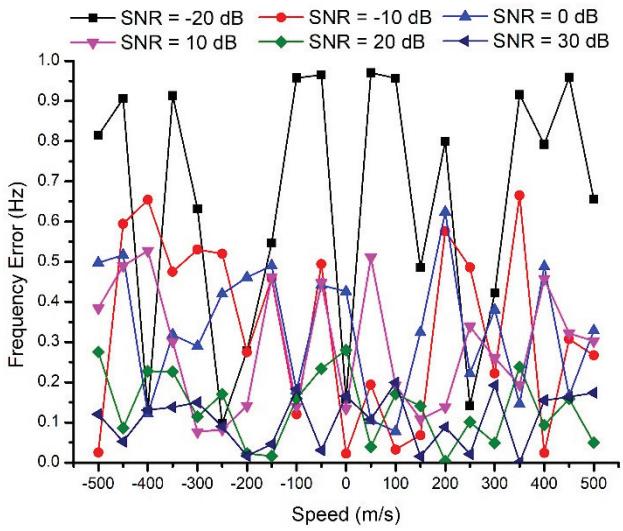

Fig. 12. Discrepancy between the theoretical and USRP simulated frequencies.

TABLE IV
RMSE OF FREQUENCY, AMPLITUDE, AND PHASE DISTORTIONS

SNR(dB)	Frequency	Amplitude	Phase
-20	0.7152	44425.29	0.8416
-10	0.4008	39338.12	0.8224
0	0.3686	23663.27	0.6841
+10	0.3232	8963.96	0.3487
+20	0.1608	3039.95	0.1855
+30	0.1245	995.06	0.1086

USRP simulated speeds of ± 100, ± 500, and ± 1000 m/s. According to (4) and Table I, the Doppler frequencies generated for a 1 GHz signal at target speeds of ±100, ± 500, and ± 1000 m/s were ± 667, ± 3336, and ± 6671 Hz, respectively, confirming that the generated signals matched these values. More importantly, Fig. 11 presents the results when a multifrequency signal containing 1.01 and 0.99 GHz was generated by the signal generator, and the USRP simulated speeds of ± 100, ± 500, and ± 1000 m/s, with the center frequency f_c set at 1 GHz. The experimental results confirmed that different Doppler frequencies were generated for the two frequencies through single interpolation, consistent with the values described in (4) and Table I. To further evaluate system performance under dynamic conditions, Fig. 11 presents experimental results obtained by varying the SNR from –20 to 30 dB in 10 dB increments at the transmitter side, while simulating Doppler frequencies corresponding to target velocities ranging from –500 to 500 m/s in 50 m/s steps at the receiver side. In Fig. 12, the measured Doppler frequencies are compared with the theoretical values derived from (4) to assess the accuracy of the proposed IRSS. The results confirm that the deviation between theoretical and measured values remained below 1 Hz under all test scenarios. Table IV summarizes the root-mean-square root mean squares (rms) errors for frequency, amplitude, and phase, offering a

TABLE V
QUANTITATIVE COMPARISON FOR VARIOUS RADAR SYSTEM SIMULATORS

	Method	Hardware Complexity			Latency	Range
		RF	Analog	Digital		
Single-Freq.	ARSS [29], [30]	1	3	1	2A + 1D	< 1MHz
	DRSS [27], [31]-[35]	2	5	2	5A	< 10MHz
	IRSS (Proposed)	2	4	2	4A + 1D	< 10MHz
Multi-Freq.	ARSS [29], [30]	N	3N + 1	N	(2A + 1D) + 1A	< 1MHz
	DRSS [27], [31]-[35]	2N	5N + 1	2N	(5A) + 1A	< 10MHz
	IRSS (Proposed)	2	4	2	4A + 1D	< 10MHz

comprehensive view of system performance. The frequency error is expressed in hertz, amplitude distortion corresponds to deviations within the 16-bit resolution, and phase error is represented in radians. Although time-domain linear interpolation may introduce slight amplitude distortion and phase discontinuity, the results indicate that such distortions were minimal and did not significantly affect overall system accuracy. This demonstrates that the proposed IRSS achieves a practical balance between hardware efficiency and simulation precision, while benefiting from the simplicity and real-time capability of the interpolation method. Moreover, all error metrics consistently decreased as the SNR increased, further validating the robustness and reliability of the system under various signal conditions.

Furthermore, Table V quantitatively compares the hardware complexity of the proposed interpolation-based RSS with that of existing analog-based RSS and digital-based RSS. To perform a numerical comparison, we conducted a comparative analysis based on the number of components required in the RF, analog, and digital domains. More specifically, we categorized the hardware components as follows: LOs under RF; Doppler Generators, Switches, and Mixers under Analog; and Doppler Calculators, DDS units, and Interpolators under Digital. This classification is clearly reflected in Table V. For single-frequency signal simulation, the ARSS [29], [30] in Fig. 2(a) has the lowest hardware complexity but is limited in frequency range by analog IC constraints. The DRSS [27], [31], [32], [33], [34], [35] in Fig. 2(b) expanded the frequency range by utilizing a direct digital synthesizer to mix Doppler frequency, but this increased hardware complexity when simulating multiple frequencies. The proposed IRSS in Fig. 8 mitigates hardware complexity while covering a wide frequency range by applying linear interpolation. More importantly, for multifrequency simulation, both ARSS [29], [30] and DRSS [27], [31], [32], [33], [34], [35] require generating separate signals for each frequency component, as shown in Fig. 3, resulting in n times greater hardware complexity compared to single-frequency structures. However, the proposed IRSS structure maintains the same hardware complexity regardless of the number of frequencies by utilizing the superposition property.

Finally, to provide a more intuitive and practical comparison of hardware complexity, Fig. 13 illustrates the relative hardware cost per supported frequency, clearly showing how

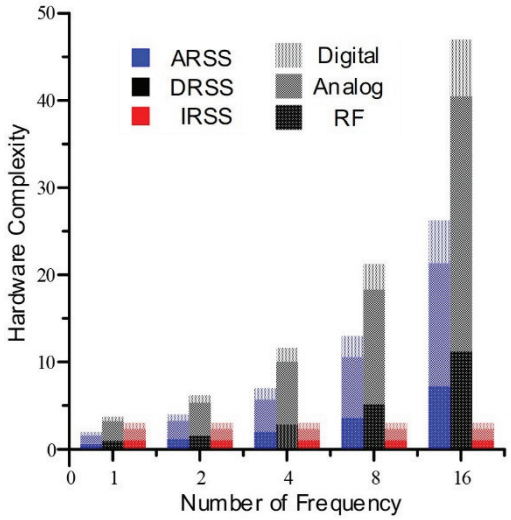

Fig. 13. Hardware cost increase rate according to the number of simulated frequencies.

overhead scales with increasing frequency count. Since it is not feasible to implement and fabricate full RF, analog, and digital subsystems for every ARSS, DRSS, and IRSS configuration, we adopted a weighted estimation model, assigning weights of 0.5, 0.3, and 0.2 to the RF, analog, and digital domains, respectively. These weights reflect general design trends and practical considerations, such as layout difficulty, sensitivity to parasitics, scalability, and integration effort, where RF components typically require complex shielding and impedance control, analog circuits involve moderate manual tuning and coupling care, and digital logic benefits from high automation and compact implementation. This weighted approach provides a more balanced and hardware-aware framework for comparing system architectures. As a result, based on the analysis presented in Fig. 13 and Table V, the proposed IRSS consistently demonstrates superior cost-efficiency and scalability, regardless of the number of supported frequencies, thereby validating the architectural advantages of the interpolation-based approach over conventional simulators.

V. CONCLUSION

In this article, we presented an IRSS designed to simulate Doppler frequencies efficiently across multiple frequencies.

The proposed system exploits the principles of linear interpolation and the superposition property to achieve frequency modification while minimizing hardware complexity. The proposed IRSS structure showed how a single interpolation can be applied to multifrequency signals, thus addressing the limitations of the previous simulators, like ARSS and DRSS, which require individual Doppler frequency generation for each frequency component. Through experimental validation using an FPGA-based USRP, we demonstrated the effectiveness of the IRSS in simulating both single and multifrequency Doppler shifts. The results confirmed that the system could accurately produce the expected Doppler frequencies for various target speeds, aligning with theoretical values. Additionally, the IRSS maintained consistent hardware complexity even as the number of supported frequencies increased, thanks to the exploitation of signal superposition. In conclusion, the IRSS offers a significant improvement in radar simulation technology by providing a method to simulate Doppler effects across multiple frequencies with reduced hardware requirements. This capability ensures that radar systems can be tested and validated in a more flexible, cost-effective, and comprehensive manner, contributing to more reliable and efficient radar system development and deployment.

ACKNOWLEDGMENT

The chip fabrication and EDA tool were supported by the IC Design Education Center (IDEC), Korea.

REFERENCES

[1] P. Bahl and V. N. Padmanabhan, "RADAR: An in-building RF-based user location and tracking system," in *Proc. 19th Annu. Joint Conf. IEEE Comput. Commun. Societies (IEEE INFOCOM)*, vol. 2, May 2000, pp. 775–784, doi: 10.1109/INFCOM.2000.832252.

[2] I. N. Bankman, "Model of laser radar signatures of ballistic missile warheads," in *Targets and Backgrounds: Characterization and Representation V*, vol. 1999. Bellingham, WA, USA: SPIE, pp. 133–137, doi: 10.1117/12.352946.

[3] H.-Y. Chen, X. Li, G.-R. Guo, and B. Jiang, "Radar feature extraction of micro-precession ballistic missile warhead," *J. Electron. Inf. Technol.*, vol. 28, pp. 643–646, May 2006.

[4] K. J. Parker, R. M. Lerner, and S. R. Huang, *Method and apparatus for using Doppler modulation parameters for estimation of vibration amplitude*, U.S. Patent 5 086 775, Feb. 11, 1992.

[5] F. Junyent and V. Chandrasekar, "Theory and characterization of weather radar networks," *J. Atmos. Ocean. Technol.*, vol. 26, no. 3, pp. 474–491, Mar. 2009, doi: 10.1175/2008jtecha1099.1.

[6] D. S. Zrnic et al., "Agile-beam phased array radar for weather observations," *Bull. Amer. Meteorological Soc.*, vol. 88, no. 11, pp. 1753–1766, Nov. 2007, doi: 10.1175/bams-88-11-1753.

[7] C. C. Wackerman, K. S. Friedman, W. G. Pichel, P. Clemente-Colón, and X. Li, "Automatic detection of ships in RADARSAT-1 SAR imagery," *Can. J. Remote Sens.*, vol. 27, no. 4, pp. 371–378, Aug. 2001.

[8] H. Kong, C. Huang, J. Yu, and X. Shen, "A survey of mmWave radar-based sensing in autonomous vehicles, smart homes and industry," *IEEE Commun. Surveys Tuts.*, vol. 27, no. 1, pp. 463–508, Feb. 2025, doi: 10.1109/COMST.2024.3409556.

[9] S. Chaudhary, A. Sharma, S. Khichar, Y. Meng, and J. Malhotra, "Enhancing autonomous vehicle navigation using SVM-based multi-target detection with photonic radar in complex traffic scenarios," *Sci. Rep.*, vol. 14, no. 1, Jul. 2024, Art. no. 17339, doi: 10.1038/s41598-024-66850-z.

[10] C. Aydogdu, M. F. Keskin, N. Garcia, H. Wymeersch, and D. W. Bliss, "RadChat: Spectrum sharing for automotive radar interference mitigation," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 1, pp. 416–429, Jan. 2021, doi: 10.1109/TITS.2019.2959881.

[11] J. Dudczyk and A. Kawalec, "Adaptive forming of the beam pattern of microstrip antenna with the use of an artificial neural network," *Int. J. Antennas Propag.*, vol. 2012, pp. 1–13, May 2012.

[12] J. Xu, L. Ren, H. Fan, E. Mao, and Q. Liu, "Clutter and range ambiguity suppression using diverse pulse train in pulse Doppler system," *Sensors*, vol. 18, no. 7, p. 2326, Jul. 2018.

[13] D. P. Jorgensen, P. H. Hildebrand, and C. L. Frush, "Feasibility test of an airborne pulse-Doppler meteorological radar," *J. Climate Appl. Meteorol.*, vol. 22, no. 5, pp. 744–757, May 1983.

[14] P. Schoeder, A. Martin, B. Meinecke, D. Werbunat, and C. Waldschmidt, "A modulation-based radar target simulator and its hardware nonidealities," in *Proc. 19th Eur. Radar Conf. (EuRAD)*, Milan, Italy, Sep. 2022, pp. 1–4, doi: 10.23919/EuRAD54643.2022.9924830.

[15] V. Gupta, "Latest trends in radar system testing," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 23, no. 5, pp. 20–25, May 2008, doi: 10.1109/MAES.2008.4523910.

[16] G. Körner, M. Hoffmann, S. Neidhardt, M. Beer, C. Carlowitz, and M. Vossiek, "Multirate universal radar target simulator for an accurate moving target simulation," *IEEE Trans. Microw. Theory Techn.*, vol. 69, no. 5, pp. 2730–2740, May 2021, doi: 10.1109/TMTT.2021.3060817.

[17] C. Birkenhauer et al., "A simple and versatile concept to improve dynamic range and enable target angle adaptability in radar target simulators," *IEEE J. Microw.*, vol. 3, no. 4, pp. 1109–1119, Oct. 2023, doi: 10.1109/JMW.2023.3296594.

[18] A. Moccia, S. Vetrella, and S. Ponte, "Passive and active calibrator characterization using a spaceborne SAR system simulator," *IEEE Trans. Geosci. Remote Sens.*, vol. 32, no. 3, pp. 715–721, May 1994, doi: 10.1109/36.297992.

[19] M. Steins and A. R. Diewald, "Implementation of delay line with fine range discretization for radar target simulations," in *Proc. 19th Int. Radar Symp. (IRS)*, Bonn, Germany, Jun. 2018, pp. 1–9, doi: 10.23919/IRS.2018.8447935.

[20] A. Diewald, T. Antes, B. Nuss, and T. Zwick, "Implementation of range Doppler migration synthesis for radar target simulation," in *Proc. IEEE 93rd Veh. Technol. Conf. (VTC-Spring)*, Apr. 2021, pp. 1–5.

[21] A. Meta, P. Hoogeboom, and L. P. Ligthart, "Signal processing for FMCW SAR," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 11, pp. 3519–3532, Nov. 2007, doi: 10.1109/TGRS.2007.906140.

[22] D. D. Blankenship et al., "Radar for Europa assessment and sounding: Ocean to near-surface (REASON)," *Space Sci. Rev.*, vol. 220, no. 5, Jun. 2024, Art. no. 51, doi: 10.1007/s11214-024-01072-3.

[23] F. Roos, J. Bechter, C. Knill, B. Schweizer, and C. Waldschmidt, "Radar sensors for autonomous driving: Modulation schemes and interference mitigation," *IEEE Microw. Mag.*, vol. 20, no. 9, pp. 58–72, Sep. 2019, doi: 10.1109/MMM.2019.2922120.

[24] I. Guvenc, F. Koohifar, S. Singh, M. L. Sichitiu, and D. Matolak, "Detection, tracking, and interdiction for amateur drones," *IEEE Commun. Mag.*, vol. 56, no. 4, pp. 75–81, Apr. 2018, doi: 10.1109/MCOM.2018.1700455.

[25] I. P. Eedara, A. Hassanien, and M. G. Amin, "Performance analysis of dual-function multiple-input multiple-output radar-communications using frequency hopping waveforms and phase shift keying signalling," *IET Radar, Sonar Navigat.*, vol. 15, no. 4, pp. 402–418, Mar. 2021, doi: 10.1049/rsn2.12043.

[26] I. P. Eedara, M. G. Amin, and A. Hassanien, "Controlling clutter modulation in frequency hopping MIMO dual-function radar communication systems," in *Proc. IEEE Int. Radar Conf. (RADAR)*, Washington, DC, USA, Apr. 2020, pp. 466–471, doi: 10.1109/RADAR42522.2020.9114840.

[27] S.-Y. Jeon et al., "W-band MIMO FMCW radar system with simultaneous transmission of orthogonal waveforms for high-resolution imaging," *IEEE Trans. Microw. Theory Techn.*, vol. 66, no. 11, pp. 5051–5064, Nov. 2018, doi: 10.1109/TMTT.2018.2867003.

[28] B. R. Mahafza, "Radar systems analysis and design using MATLAB," in *3rd, Ed., Boca Raton*, 2000.

[29] P. Eskelinen and J. Ruoskanen, "A baseband Doppler simulator for radar system development," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 21, no. 4, pp. 8–11, Apr. 2006, doi: 10.1109/MAES.2006.1626057.

[30] A. D. Droitcour, O. Boric-Lubecke, V. M. Lubecke, J. Lin, and G. T. A. Kovacs, "Range correlation and *I/Q* performance benefits in single-chip silicon Doppler radars for noncontact cardiopulmonary monitoring," *IEEE Trans. Microw. Theory Techn.*, vol. 52, no. 3, pp. 838–848, Mar. 2004, doi: 10.1109/TMTT.2004.823552.

[31] D. Meena, T. Roy, and L. Prakasam, "Design of multilevel radar target simulator," in *Proc. IEEE Radar Conf.*, Waltham, MA, USA, Apr. 2007, pp. 203–208, doi: 10.1109/RADAR.2007.374213.

[32] A. Diewald et al., "Radar target simulation for vehicle-in-the-loop testing," *Vehicles*, vol. 3, no. 2, pp. 257–271, May 2021, doi: 10.3390/vehicles3020016.

[33] S. Lutz, C. Erhart, T. Walte, and R. Weigel, "Target simulator concept for chirp modulated 77 GHz automotive radar sensors," in *Proc. 11th Eur. Radar Conf.*, Rome, Italy, Oct. 2014, pp. 65–68, doi: 10.1109/eurad.2014.6991208.
[34] M. Engelhardt, F. Pfeiffer, and E. Biebl, "A high bandwidth radar target simulator for automotive radar sensors," in *Proc. Eur. Radar Conf. (EuRAD)*, Oct. 2016, pp. 245–248.
[35] P. Roszkowski, "Carrier- and Doppler-tunable FPGA-based active reflector for radar calibration," in *Proc. 17th Int. Radar Symp. (IRS)*, Krakow, Poland, May 2016, pp. 1–6, doi: 10.1109/IRS.2016.7497266.
[36] T. Moon, J. Park, and S. Kim, "BlueFMCW: Random frequency hopping radar for mitigation of interference and spoofing," *EURASIP J. Adv. Signal Process.*, vol. 2022, no. 1, pp. 1–12, Dec. 2022, doi: 10.1186/s13634-022-00838-7.
[37] W. Scheiblhofer, R. Feger, A. Haderer, and A. Stelzer, "A low-cost multi-target simulator for FMCW radar system calibration and testing," in *Proc. Eur. Radar Conf. (EURAD)*, Nuremberg, Germany, Oct. 2017, pp. 343–346, doi: 10.23919/EURAD.2017.8249217.
[38] Z. Zhao, H. Wang, L. Cao, D. Wang, and C. Fu, "Doppler-spread target summation variability index CFAR detector for FMCW radar," *IEEE Sensors J.*, vol. 24, no. 20, pp. 32519–32532, Oct. 2024, doi: 10.1109/JSEN.2024.3432179.
[39] J. Dudczyk and A. Kawalec, "Identification of emitter sources in the aspect of their fractal features," *Bull. Polish Acad. Sci., Tech. Sci.*, vol. 61, no. 3, pp. 623–628, Sep. 2013, doi: 10.2478/bpasts-2013-0065.
[40] F. Yan, S. Zhao, S. E. Venegas-Andraca, and K. Hirota, "Implementing bilinear interpolation with quantum images," *Digit. Signal Process.*, vol. 117, Oct. 2021, Art. no. 103149.
[41] R. T. Smith and R. B. Minton, *Calculus*. New York, NY, USA: McGraw-Hill, 2002, pp. 174–177.
[42] S. McKinley and M. Levine, "Cubic spline interpolation," *College Redwoods*, vol. 45, no. 1, pp. 1049–1060, 1998.

Sukjae Yoon (Member, IEEE) received the B.S. degree in electrical engineering from Kongju National University (KNU), Cheonan, South Korea, in 2015, and the M.S. degree in electronics engineering from Chungnam National University (CNU), Daejeon, South Korea, in 2018, where he is currently pursuing the Ph.D. degree in electronic engineering, with a focus on digital signal processing, digital systems, anti-jamming, and adaptive algorithms.

Kyoduk Ku (Member, IEEE) received the B.S. degree in radio and information communications engineering from Chungnam National University (CNU), Daejeon, South Korea, in 2023, where he is currently pursuing the integrated M.S. and Ph.D. degree in electronics engineering.

His research interests include GNSS receiver hardware, jamming and anti-jamming, and digital systems.

Hoyoung Yoo (Member, IEEE) received the B.S. degree in electrical and electronics engineering from Yonsei University, Seoul, South Korea, in 2010, and the M.S. and Ph.D. degrees in electrical engineering from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea, in 2012 and 2016, respectively.

Since 2016, he has been with the Department of Electronics Engineering, Chungnam National University (CNU), Daejeon, where he is currently a Professor. Prior to joining CNU in 2016, he was with Samsung Electronics, Hwaseong, South Korea, where he was involved in research on nonbinary LDPC decoders for NAND flash memories. His current research interests include algorithms and architectures for error-correcting codes, FPGA reverse engineering, GNSS communications, and 5G communication systems.