Received XX Month, XXXX; revised XX Month, XXXX; accepted XX Month, XXXX; Date of publication XX Month, XXXX; date of current version XX Month, XXXX.

Digital Object Identifier 10.1109/XXXX.2022.1234567

An FPGA-Calibrated Modulo ADC for High-Dynamic-Range Signal Acquisition

Zeyuan Li[1] (Student Member, IEEE), Wenyi Yan[2] (Student Member, IEEE), Lu Gan[2] (Senior Member, IEEE), Guoquan Li[3] (Senior Member, IEEE), Hongqing Liu[3] (Senior Member, IEEE)

[1]The School of Integrated Circuits, Anhui University, China
[2]The College of Engineering, Design and Physical Sciences, Brunel University of London, U.K.
[3]The School of Communications and Information Engineering, Chongqing University of Posts and Telecommunications, China

Corresponding author: Lu Gan (email: lu.gan@brunel.ac.uk).

ABSTRACT Conventional analog-to-digital converters (ADCs) suffer from irreversible clipping when input amplitudes exceed their fixed full-scale range. Modulo ADCs address this by folding the input signal prior to quantization; however, stable hardware operation under deep-folding conditions and experimental validation of algorithmic recoverability remain outstanding challenges. This paper presents a calibrated FPGA-based modulo ADC platform in which a finite state machine with multi-bit update control governs folding dynamics, replacing instantaneous comparator-triggered feedback to ensure deterministic and stable operation. A controlled under-compensation calibration strategy converts fold-dependent threshold mismatch and oscillatory instability into a bounded constant residual, enabling reliable folding at high folding depths. The platform achieves a dynamic range expansion exceeding two orders of magnitude while maintaining signal fidelity comparable to that of the standalone ADC. Experimental validation across diverse waveforms confirms robust signal acquisition and demonstrates the practical feasibility of dynamic-range extension using modulo sampling principles.

INDEX TERMS Analog-to-digital converters (ADCs), modulo sampling, sampling methods, signal reconstruction, field-programmable gate array (FPGA) implementation.

I. Introduction

ANALOG-TO-DIGITAL converters (ADCs) provide the fundamental interface between analog physical signals and digital processing systems and are essential components in sensing, communication, and digital signal processing applications [1]–[4]. However, conventional ADCs operate within a fixed linear input range; signal excursions exceeding this limit inevitably lead to overload and irreversible clipping, resulting in permanent information loss [5]–[9]. While traditional conditioning techniques like automatic gain control (AGC) attempt to mitigate this issue, they rescale input amplitudes without expanding the converter's intrinsic range, often introducing unwanted switching transients and nonlinear distortion [10], [11].

To address these limitations, modulo ADCs, also identified as the unlimited sampling paradigm, utilize a hardware–algorithm co-design framework [12], [13]. In this architecture, the hardware performs a controlled folding operation to map input signal excursions back within the linear threshold of the ADC, effectively circumventing saturation. Subsequently, these folded samples are digitized, and digital reconstruction algorithms are employed to recover the original waveform from the modulo-sampled data.

Early designs relied on purely analog architectures, typically utilizing integrator loops to realize signal folding [14], [15]. While structurally simple, these self-oscillatory systems lack explicit control over feedback dynamics, making them inherently sensitive to component mismatch and temperature-induced drift, with folding depth and precision fixed at the circuit level. Mixed-signal architectures attempted to improve upon these designs by integrating microcontroller- and digital-to-analog converter (DAC)-based feedback, yet they remain fundamentally reactive [16], [17]. Relying on instantaneous threshold-crossing triggers, these prototypes suffer from loop-latency-induced distortions and oscillatory instabilities. Rigid threshold configurations

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

and limited DAC resolution further constrain the achievable folding depth, restricting the practical dynamic range extension. Collectively, these non-idealities necessitate prohibitively high oversampling factors (OF) [18]–[20], forcing reconstruction algorithms to rely on dense sampling to compensate for stochastic folding errors and undermining the intrinsic efficiency of modulo sampling.

To address these limitations, we propose an FPGA-based architecture that transforms modulo folding into a deterministic state-space evolution, enhancing control over folding dynamics while employing a calibration mechanism to improve signal fidelity. The architecture achieves dynamic range expansion exceeding two orders of magnitude without necessitating prohibitive oversampling factors. The main contributions are as follows:

- **State-space controlled modulo folding architecture:** The FPGA serves as a synchronous digital controller governing modulo folding through a state-space mechanism, replacing instantaneous comparator-triggered feedback to ensure stable folding dynamics.
- **Controlled under-compensation calibration:** A calibration scheme that intentionally enforces bounded under-compensation during folding transitions, transforming stochastic overshoot and oscillatory behavior into a deterministic constant residual within the tolerance of reconstruction algorithms, thereby simplifying digital recovery.
- **High-fidelity folding for efficient signal reconstruction:** Featuring configurable thresholds, the architecture enables reliable reconstruction across diverse amplitudes using standard algorithms without excessive oversampling, with hardware validation confirming stable operation for signals exceeding 100 times the ADC linear range.

The remainder of this paper is organized as follows. Section II reviews related hardware and recovery methods, while Section III presents the proposed mixed-signal modulo ADC platform. Section IV details the calibration scheme and analyzes the system bandwidth limitations. Section V reports the experimental results, and Section VI concludes the paper.

II. Modulo ADC Principles and Related Work

A. Modulo Folding Principles

Consider a real-valued input signal $g(t)$ and a modulo ADC with folding threshold $\lambda > 0$. The modulo operation and the resulting folded signal $y(t)$ are defined as [14]

$$y(t) = \mathcal{M}_\lambda\{g(t)\} = g(t) - 2\lambda \left\lfloor \frac{g(t)+\lambda}{2\lambda} \right\rfloor, \qquad (1)$$

where $\lfloor \cdot \rfloor$ denotes the floor function. In the proposed hardware prototype, the folding count is tracked by a digital counter $C_f[n]$, updated synchronously at each FPGA clock cycle, such that the folded signal is expressed as

$$y(t) = g(t) + v_f(t), \qquad (2)$$

where $v_f(t) = 2\lambda C_f[n]$ denotes the feedback voltage held constant over each interval $t \in [nT_{\text{clk}}, (n+1)T_{\text{clk}})$. The design and control of $C_f[n]$ are detailed in Section III.

This approach differs fundamentally from conventional dynamic-range extension techniques. AGC prevents ADC overload by reducing front-end gain in the presence of large signal components; however, such global attenuation may suppress weak features below the quantization noise floor [21], [22]. Oversampling $\Delta\Sigma$ converters improve in-band signal-to-noise ratio (SNR) through noise shaping and decimation, which is effective at moderate bandwidths, but the required oversampling ratio and decimation complexity increase significantly with bandwidth and resolution [23]–[25]. Successive-approximation-register (SAR) ADCs offer high energy efficiency and can extend dynamic range via range switching or multi-step conversion, yet remain constrained by a fixed full-scale input limit [26]–[28].

B. Hardware Architectures for Modulo ADC Systems

The dynamic range expansion factor ρ quantifies the dynamic range extension of a modulo ADC and is defined as [29]

$$\rho \triangleq \frac{\|g\|_\infty}{\lambda}, \qquad (3)$$

where $\|g\|_\infty$ denotes the peak signal amplitude. Larger values of ρ indicate a greater dynamic-range expansion.

Early purely analog architectures achieved folding through uncontrolled feedback loops [14], [15]. Despite their structural simplicity, these designs suffer from limited precision and stability, restricting their practical dynamic range expansion. Subsequent mixed-signal implementations integrated digital feedback paths with DAC-based compensation to improve folding control [16], [17]. These prototypes have demonstrated dynamic range expansion factors of $\rho \approx 8$–10 at bandwidths below 10 kHz. However, further scaling of these architectures is constrained by practical limitations such as control-loop latency and threshold resolution, which cap the achievable ρ and operating bandwidth. Moreover, neither class of hardware has provided systematic experimental validation of signal reconstruction algorithms under real folded hardware data, particularly at large folding depths.

C. Signal Recovery Approaches

After folding, the signal is sampled to produce discrete-time observations $y[k] = \mathcal{M}_\lambda(g(kT))$, where T denotes the sampling period, and the original signal must be recovered from these folded samples. Representative approaches include finite-difference-based methods [14], [29], sparse recovery formulations [30], [31], and spectral techniques [32].

Beyond these foundational methods, recent research has explored diverse architectures, including 1-bit unlimited sampling [33]–[40] and multi-channel folding configurations [41]–[44] with hardware implementation [45]. While these architectures offer unique advantages for specific modalities, our primary focus is establishing a baseline high-fidelity single-channel hardware platform for modulo ADC

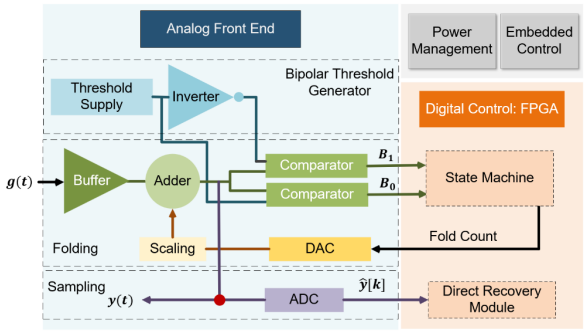

FIGURE 1. **Block diagram of the FPGA-based modulo ADC prototype.**

systems, and reconstruction algorithms are employed solely as a validation tool to assess the quality and robustness of the folded hardware signals.

Accordingly, two practical reconstruction scenarios are considered in this work: **(a)** when folding information is explicitly recorded by the FPGA, the waveform is directly reconstructed in real time to verify system synchronization and confirm that the hardware folding operates correctly (Section III-C); **(b)** when such information is unavailable, representative reconstruction algorithms [14], [29]–[32] are applied to the folded data to enable quantitative evaluation of signal recovery performance (Section V).

III. Modulo ADC Architecture and Calibration

A. System Architecture and Modeling

As illustrated in Fig. 1, the system operates as a closed feedback loop: a digital folding-count register $C_f[n]$, where n denotes the index after the n-th FPGA clock edge, drives a DAC to generate a compensation voltage $v_f(t)$, which is summed with the input $g(t)$ to produce the folded signal $y(t) = g(t) + v_f(t)$. Since the DAC output is updated synchronously with the FPGA clock and maintained via a zero-order hold (ZOH) mechanism, the feedback voltage is modeled as

$$v_f(t) = G_{\text{total}} V_{\text{DAC,step}} C_f[n], \quad t \in [nT_{\text{clk}}, (n+1)T_{\text{clk}}) \tag{4}$$

where $V_{\text{DAC,step}}$ is the DAC output increment per folding event and G_{total} is the cumulative feedback gain. To ensure seamless folding, the system is calibrated such that $G_{\text{total}} V_{\text{DAC,step}} = 2\lambda$, so that each unit increment in C_f produces an exact 2λ shift at the analog summing node.

The folded signal $y(t)$ is monitored by a high-speed window comparator against predefined thresholds $\pm\lambda$. Upon a threshold crossing, the comparator generates digital status flags synchronized to the FPGA domain at each rising clock edge, driving the feedback controller to update $C_f[n+1]$ to track the excursion of $y(t)$ relative to the thresholds.

Finally, the folded waveform $y(t)$ is digitized by an on-board ADC to produce discrete samples $\hat{y}[k]$. Real-time signal integrity is monitored using an integrated logic analyzer, while an on-chip recovery module validates the folding consistency by fusing $\hat{y}[k]$ with the concurrent folding count C_f.

TABLE 1. **List of hardware components.**

Component	Model Number	Manufacturer
Comparator	LT1715	Analog Devices
DAC	AD9744	Analog Devices
Voltage Reference	REF5050	Texas Instruments
Adder / Buffer	OPA892	Texas Instruments
Inverter	ADA4522	Analog Devices
ADC	AD9288	Analog Devices
FPGA	ZYNQ 7000 XC7Z100	Xilinx

The physical implementation of the proposed architecture is demonstrated in the fabricated prototype (Fig. 2), with hardware specifications summarized in Table 1.

B. Modulo ADC Hardware Implementation

The proposed modulo ADC platform comprises two tightly coupled components: an analog front end and an FPGA-based digital control loop. The analog front end generates folding feedback to extend the usable amplitude range, while the digital control loop detects threshold crossings, updates the folding count, and synchronizes the feedback. The following subsections describe their joint design.

1) Threshold Selection

In a practical hardware implementation, the maximum achievable amplitude is bounded by the supply voltage V_{supply}. Assuming a rail-to-rail analog front end, the maximum input amplitude satisfies $\|g\|_\infty \lesssim V_{\text{supply}}$, leading to

$$\rho \lesssim \frac{V_{\text{supply}}}{\lambda}.$$

This establishes a fundamental scaling law: for a fixed supply voltage, the dynamic range expansion scales inversely with the threshold, i.e., $\rho \propto 1/\lambda$, revealing a trade-off between folding depth and robustness. Smaller thresholds enable deeper folding (larger ρ) but increase sensitivity to noise and circuit non-idealities, whereas larger thresholds improve robustness at the cost of reduced folding resolution.

In this design, a nominal threshold range of $[100, 500]$ mV is selected to balance folding depth and system stability. For example, with $V_{\text{supply}} = 12$ V and $\lambda = 100$ mV, the theoretical maximum dynamic range expansion is approximately $\rho \approx 120$ under ideal rail-to-rail operation.

The threshold generator employs a zero-drift amplifier (ADA4522) and a precision voltage reference generator (REF5050) to generate the threshold within the desired range. These components are chosen for their excellent thermal stability and low offset, maintaining voltage balance within $\pm 0.2\%$ and effectively eliminating temperature-induced drift. Threshold detection is performed using a dual-channel high-speed comparator (LT1715) featuring a 4 ns propagation delay, 150 MHz bandwidth, and a 3.5 mV Schmitt-trigger hysteresis, which suppresses chattering near

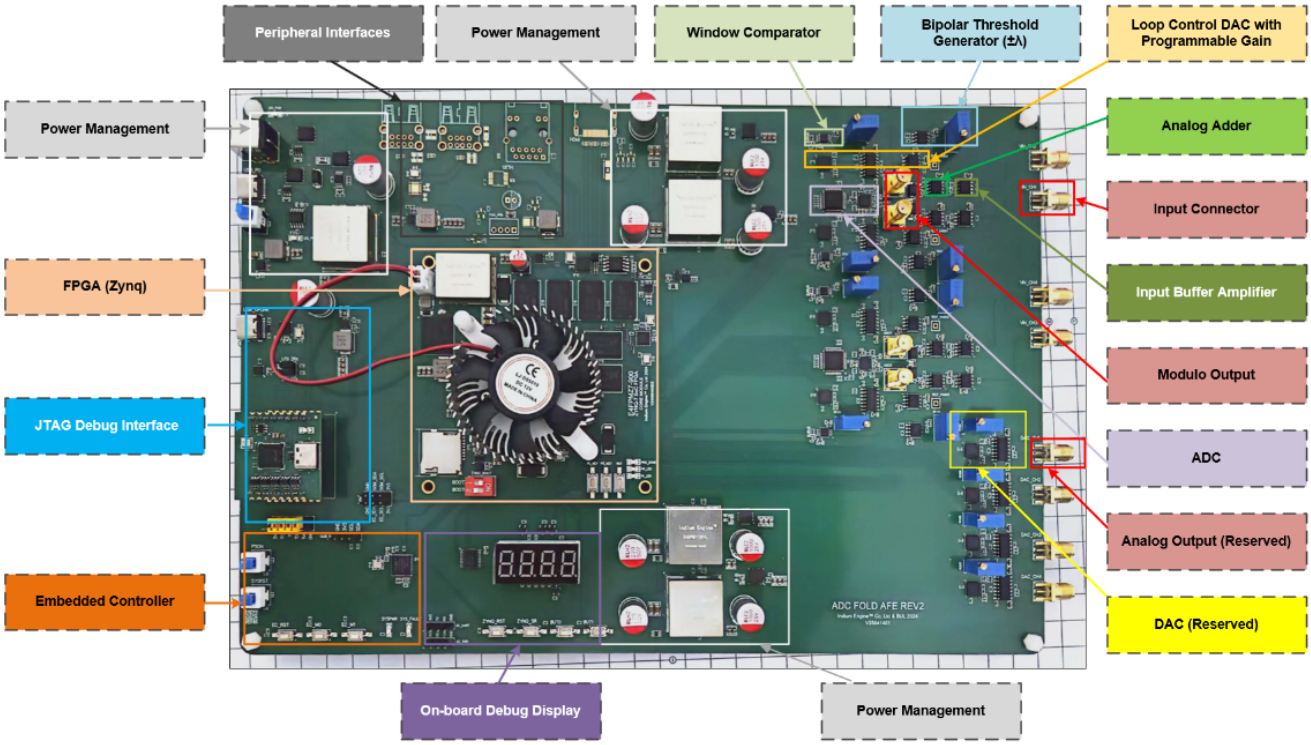

FIGURE 2. Photograph of the fabricated FPGA-based modulo ADC prototype. Key components of a single channel are labeled, including the FPGA-based digital loop controller, window comparator, bipolar threshold generator, loop-control DAC with programmable gain stage, adder, and input buffer. Additional on-board modules provide ADC/DAC functionality, power management, and peripheral interfaces for stand-alone operation, with extra channels reserved for future expansion.

threshold while the matched dual channels ensure synchronized detection of both polarities. The comparator outputs are encoded as a 2-bit flag $[B_1B_0]$, where:

- $[B_1B_0] = [00]$: $y(t) \in [-\lambda, +\lambda)$.
- $[B_1B_0] = [01]$: $y(t) > +\lambda$ (positive over-range).
- $[B_1B_0] = [10]$: $y(t) < -\lambda$ (negative over-range).

2) Folding Feedback Generation and Depth Configuration

To ensure robust high-rate folding and facilitate streamlined calibration, the analog front end is designed for flexible threshold control. In the prototype implementation, the feedback path consists of a 14-bit current-steering high-speed DAC AD9744 (200 MSPS, 1 bit reserved for sign), a Thévenin-equivalent resistor network, a programmable-gain amplifier VCA824 with gain $G_{\mathrm{PGA}} \in [0, 2]$ V/V, and a fixed $\times 10$ buffer stage, yielding a cumulative gain range of $G_{\mathrm{total}} \in [0, 20]$ V/V. This provides sufficient headroom to satisfy the gain-matching constraint $G_{\mathrm{total}} V_{\mathrm{DAC,step}} = 2\lambda$ across various threshold settings, with G_{total} adjusted to accommodate different values of λ.

To optimize the DAC's dynamic range utilization, a programmable multi-bit scheme is employed where each folding event corresponds to 2^q DAC codes. The resulting DAC output step is defined as:

$$V_{\mathrm{DAC,step}} = 2^q V_{\mathrm{LSB}} = 2^q \frac{V_{\mathrm{FS}}}{2^{B_m}}, \tag{5}$$

where $V_{\mathrm{FS}} = 1.0$ V is the DAC full-scale output voltage, q is a configurable parameter providing real-time control over the effective resolution, and B_m denotes the number of magnitude bits. Since the AD9744 is a 14-bit DAC with 1 bit reserved for sign, we have $B_m = 13$, and the maximum achievable folding depth $C_{f,\max}$ is constrained by the available digital headroom:

$$C_{f,\max} = \left\lfloor \frac{2^{B_m} - 1}{2^q} \right\rfloor \approx 2^{B_m - q} - 1. \tag{6}$$

By selecting q, the DAC step size, the required analog gain, and the maximum folding depth can be jointly configured. Specifically, a larger q increases $V_{\mathrm{DAC,step}}$, reducing the required G_{total} to meet the 2λ condition and thereby minimizing noise amplification in the feedback loop, at the cost of reduced $C_{f,\max}$. For a target dynamic range expansion ratio ρ, the hardware must satisfy $C_{f,\max} > \rho/2$, and q is selected to balance these competing demands while keeping G_{total} within the available hardware range.

This trade-off is illustrated in Table 2 for $\lambda = 300$ mV. Smaller step sizes (e.g., 2^7 codes) enable deeper folding but require a gain exceeding the 20 V/V hardware limit, while larger steps (e.g., 2^9 codes) offer a practical balance between folding depth and gain headroom. For example, targeting $\rho = 108$ requires $C_{f,\max} > \rho/2 = 54$; selecting $q = 7$ yields $C_{f,\max} = 63$, which satisfies this constraint while keeping G_{total} within the available hardware range. Furthermore, this programmable mechanism accommodates varying

TABLE 2. Trade-off between DAC Step Size, Fold Count, and Required Gain for $\lambda = 300$ mV.

Step Size	$V_{\mathrm{DAC,step}}$ (mV)	$C_{f,\max}$	G_{total} (V/V)
2^7 codes	15.62	63	38.41†
2^9 codes	62.46	15	9.60
2^{11} codes	249.84	3	2.40

†Exceeds the maximum available gain of 20 V/V in this design.

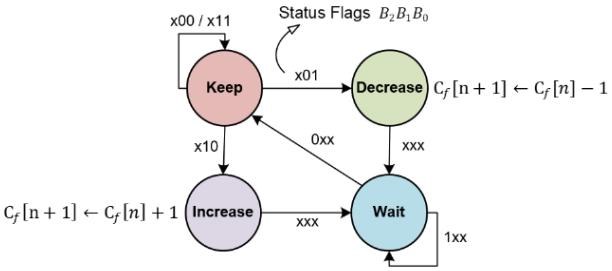

FIGURE 3. Control loop FSM: States KEEP/WAIT (hold $C_f[n]$), INCREASE/DECREASE ($C_f[n]\pm1$). Transitions depend on $[B_2B_1B_0]$ and "x" means don't-care.

threshold settings and provides additional operational margin for system calibration.

3) Folding Controller and FSM Logic

The folding count C_f is controlled by a finite state machine (FSM) implemented in the FPGA logic, which processes the synchronized outputs of the analog front-end comparators. Its state transitions are determined by a 2-bit status vector $[B_1B_0]$, indicating the window-comparator thresholds, together with a settling flag B_2. The B_2 signal, generated by a programmable timer, confirms that analog transients (e.g., DAC settling and amplifier slew) have decayed, ensuring stable feedback operation. The FSM, illustrated in Fig. 3, transitions between four primary functional states:

- KEEP: The register maintains its current state, expressed as $C_f[n+1] = C_f[n]$, when the input signal remains within the linear range ($\pm\lambda$). This condition corresponds to the status bits $[B_2B_1B_0] \in \{\mathrm{x00}, \mathrm{x11}\}$, where x denotes a don't-care bit.
- INCREASE/DECREASE: When a threshold crossing is detected, denoted by $[B_2B_1B_0] = [\mathrm{x10}]$ for $y(t) < -\lambda$ or $[\mathrm{x01}]$ for $y(t) > +\lambda$, the FSM updates the folding count as $C_f[n+1] = C_f[n] \pm 1$.
- WAIT: To suppress spurious switching and limit-cycle oscillations induced by loop propagation delays, the FSM transitions to a WAIT state upon $B_2 = 1$. During this interval, C_f remains frozen, allowing the analog residue to settle within the comparator window prior to the subsequent decision cycle. This latency, typically set to several clock cycles via iterative empirical tuning, is negligible relative to the period of the maximum input frequency, thereby ensuring that distortion introduced by the wait-state operation remains insignificant.

TABLE 3. FPGA Power Consumption Summary.

Part	Power (W)	Percentage
FPGA Static	0.245	40%
Folding Controller Logic	0.01	2%
Clock Distribution System	0.109	18%
I/O Bank	0.241	40%
Total	0.605	100%

A fundamental constraint underlying the FSM design is that the input signal should not traverse more than one folding interval within two adjacent clock decisions. Otherwise, multiple threshold-crossing events may occur between two FPGA updates and the discrete-time controller may fail to track the correct folding count.

Let $t_0 \in [nT_{\mathrm{clk}}, (n+1)T_{\mathrm{clk}})$ and $t_1 \in [(n+1)T_{\mathrm{clk}}, (n+2)T_{\mathrm{clk}})$. For a bandlimited input $g \in PW_\Omega$, the signal variation satisfies [46]–[48]

$$|g(t_1) - g(t_0)| \le 2\sin\left(\frac{\Omega|t_1 - t_0|}{2}\right)\|g\|_\infty.$$

Since $|t_1 - t_0| \le 2T_{\mathrm{clk}}$, we obtain

$$|g(t_1) - g(t_0)| \le 2\sin(\Omega T_{\mathrm{clk}})\|g\|_\infty.$$

Therefore, to ensure single-step folding updates, the controller parameters should satisfy

$$2\sin(\Omega T_{\mathrm{clk}})\|g\|_\infty < 2\lambda.$$

Under this condition, the input variation between two adjacent clock decisions is smaller than one folding step, and the folding count can be updated by at most one level, i.e.,

$$|C_f[n+1] - C_f[n]| \le 1.$$

This constraint guides the joint selection of T_{clk}, λ, and the admissible input bandwidth and amplitude range.

For the 200 MHz FPGA clock used in this prototype, $T_{\mathrm{clk}} = 5$ ns. Applying the small-angle approximation $\sin(\Omega T_{\mathrm{clk}}) \approx \Omega T_{\mathrm{clk}}$ for $\Omega T_{\mathrm{clk}} \ll 1$, the bandwidth constraint reduces to

$$f < \frac{\lambda}{2\pi T_{\mathrm{clk}}\|g\|_\infty}.$$

For example, with $\lambda = 100$ mV and $\|g\|_\infty = 9.2$ V (corresponding to $\rho = 92$), this yields a theoretical upper bound of approximately 346 kHz. In practice, however, the achievable bandwidth is substantially lower, as DAC settling delay and threshold jitter introduce additional latency that further restricts reliable folding operation. A detailed analysis of these bandwidth limitations is presented in Section IV.

The power consumption of the digital folding controller is less than 0.01 W at 200 MHz, with the full FPGA power breakdown summarized in Table 3.

Remark. As a proof-of-concept prototype, this work prioritizes demonstrating the feasibility of stable HDR folding over power efficiency. The current implementation has not been optimized for power consumption, and significant reductions are expected in future optimized realizations.

C. Synchronization and Observation Support

1) System Clocking and Synchronization

Precise synchronization among the ADC, folding controller, and loop-control DAC is required under rapid folding dynamics. A unified clock network is implemented using an FPGA-integrated phase-locked loop, where a 50 MHz reference is multiplied to generate phase-aligned clocks for all subsystems. The ADC operates on a 100 MHz clock advanced by 60° to align sampling with the settled feedback signal, while the controller and DAC use in-phase 200 MHz clocks.

2) On-Chip Signal Observation and Reconstruction

To enable real-time monitoring, the on-chip module reconstructs the high-dynamic-range (HDR) signal by combining the digitized residue $\hat{y}[k]$ with the synchronized folding-count register $C_f[n]$. Since the ADC sampling clock is slower than the FPGA control clock, the k-th ADC sample corresponds to FPGA clock index $n_k = \lfloor t_k / T_{\text{clk}} \rfloor$, where t_k is the k-th ADC sampling instant. The reconstructed signal is therefore

$$\tilde{g}[k] = \hat{y}[k] - 2\lambda\, C_f[n_k - d], \tag{7}$$

where d denotes the delay compensation in clock cycles.

IV. Calibration Strategy and System Bandwidth Analysis

A. Calibration for Folding Fidelity

1) Folding Instability: Observation and Modeling

To motivate the calibration strategy, we first experimentally characterize the hardware non-idealities present in the folding process. We characterize the actual folding thresholds as $\lambda_{e,+}$ and $\lambda_{e,-}$ for positive and negative events, respectively, with their statistical performance summarized in Table 4. The results reveal significant discrepancies between $\mathbb{E}[\lambda_{e,\pm}]$ and the nominal threshold λ, alongside non-negligible stochastic jitter $\text{Std}(\lambda_{e,\pm})$. The mean deviation arises primarily from feedback loop latency, while jitter scales positively with folding depth ρ: at $\lambda = 370$ mV, $\text{Std}(\lambda_{e,\pm})$ escalates from 10.48 mV to 25.85 mV as ρ increases, indicating that aggressive dynamic-range expansion drives the feedback loop into a regime prone to transient instability. Fig. 4 further confirms this polarity-dependent bias for $\lambda = 370$ mV through the measured threshold probability density distributions.

The statistical characterization above reveals a systematic dependence on folding depth, suggesting an underlying instability mechanism. We now analyze this analytically, beginning with the ideal folding process. When $y(t)$ increases and crosses the positive threshold λ at time $t_0 \in [nT_{\text{clk}}, (n+1)T_{\text{clk}})$, the system initiates a folding event $C_f[n+1] = C_f[n] - 1$, such that the ideal folded signal satisfies

$$y(t_0) = g(t_0) + 2\lambda(C_f[n] - 1) = -\lambda. \tag{8}$$

In practice, the folding event is applied with a total loop latency τ_d, and the effective threshold deviates from its nominal value due to hardware non-idealities. Define $\delta_g = g(t_0 + \tau_d) - g(t_0)$ as the signal increment during τ_d. Let $\lambda_{d,+}(t) = \lambda_{e,+}(t) - \lambda$ denote the time-varying positive threshold deviation, which arises from two sources: the residual signal evolution during the loop latency τ_d, and DAC non-idealities including gain mismatch and incomplete

TABLE 4. Statistical Characterization of Folding Thresholds under Various λ and ρ.

λ (mV)	ρ	$\lambda_{e,+}$ (mV)		$\lambda_{e,-}$ (mV)	
		$\mathbb{E}[\lambda_{e,+}]$	$\text{Std}(\lambda_{e,+})$	$\mathbb{E}[\lambda_{e,-}]$	$\text{Std}(\lambda_{e,-})$
100	2.56	102.80	1.39	-99.40	1.64
	3.64	103.62	5.94	-102.00	5.29
370	1.57	377.70	10.48	-371.60	8.06
	2.94	369.45	25.85	-378.08	22.52
480	4.25	490.80	6.50	-491.60	4.08
	10.33	475.84	9.45	-476.32	11.90

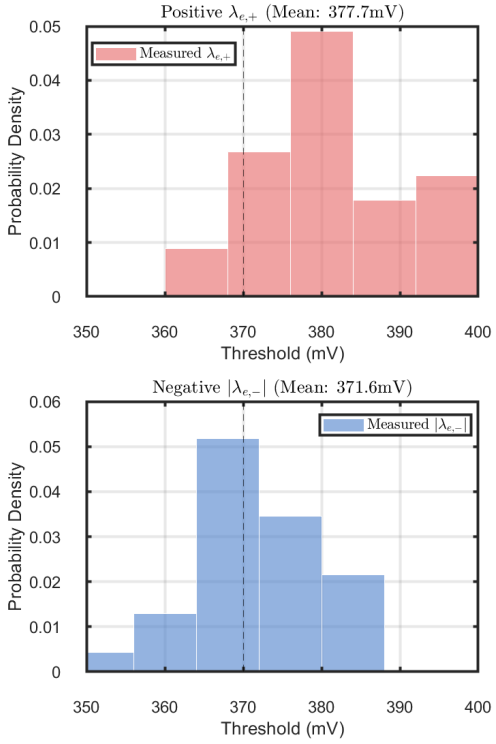

FIGURE 4. Statistical distribution and Gaussian fitting of operational folding thresholds ($\lambda = 370$ mV, $\rho = 1.57$).

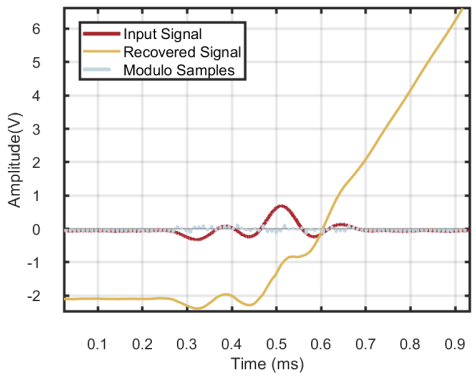

FIGURE 5. **Hardware experimental result demonstrating folding failure with input signal $B = 10\ \mathrm{kHz}$, $\lambda = 100\ \mathrm{mV}$.**

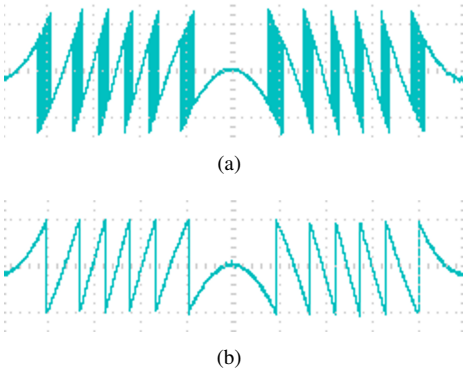

FIGURE 6. **Folded waveforms ($\lambda = 100\ \mathrm{mV}$): (a) Before calibration, overshoot and oscillations distort the folding transitions. (b) After controlled under-compensation, transitions are stabilized.**

settling, both of which prevent the feedback voltage from exactly realizing the nominal 2λ shift. The actual post-update modulo signal is then

$$y_a(t_0+\tau_d) \approx -\lambda + \delta_g + 2\lambda_{d,+}(t_0+\tau_d)(C_f[n]-1). \quad (9)$$

For a bandlimited signal $g \in PW_\Omega$, the delay-induced term is bounded by $|\delta_g| \le 2\sin(\Omega\tau_d/2)\|g\|_\infty$ [46]–[48], which remains small when $\Omega\tau_d \ll 1$. The dominant instability driver is the threshold mismatch term $2\lambda_{d,+}(t)(C_f[n]-1)$, whose magnitude scales with $|C_f[n]|$: deeper folding progressively amplifies this error and drives the system toward instability.

Since $|\lambda_{d,\pm}(t)|$ is small relative to λ, the effective negative threshold $\lambda_{e,-}(t)$ is well approximated by $-\lambda$. The oscillation condition $y_a(t_0+\tau_d) < \lambda_{e,-}(t)$ therefore reduces to $y_a(t_0+\tau_d) < -\lambda$, which gives

$$2\lambda_{d,+}(t_0+\tau_d)(C_f[n]-1) < -\delta_g < 0. \quad (10)$$

The mismatch term $2\lambda_{d,+}(t_0+\tau_d)(C_f[n]-1)$ directly displaces $y_a(t_0+\tau_d)$ below $-\lambda$ and triggers oscillation. When this condition holds, each update drives $y_a(t)$ across the opposite threshold, triggering a reverse correction and yielding the limit-cycle evolution

$$C_f[n] \to C_f[n]-1 \to C_f[n] \to \cdots. \quad (11)$$

By symmetry, an analogous instability arises for negative threshold crossings, with the roles of $\lambda_{d,+}(t)$ and $\lambda_{d,-}(t)$ exchanged.

Fig. 5 illustrates a failure scenario where folding oscillations induced by loop latency and threshold mismatch exceed the error tolerance of the recovery algorithm, resulting in catastrophic signal reconstruction errors. This instability necessitates the calibration strategy developed in the following subsection.

2) Calibration Strategy

To suppress the mismatch-induced oscillation, we propose a direction-dependent under-compensation mechanism. Considering first the case where $y(t)$ is increasing and crosses λ at t_0, we define the calibrated signal as

$$y_{\mathrm{cal}}(t_0+\tau_d) = y_a(t_0+\tau_d) + |C_f[n]|\Delta V, \quad (12)$$

where $\Delta V > 0$ is an empirically selected constant. Substituting the expression for $y_a(t_0+\tau_d)$, we obtain

$$y_{\mathrm{cal}}(t_0+\tau_d) \approx -\lambda + \delta_g + 2\lambda_{d,+}(t_0+\tau_d)(C_f[n]-1) + |C_f[n]|\Delta V. \quad (13)$$

We analyze the two instability scenarios separately.

(1) $C_f[n] > 1$, $\lambda_{d,+}(t_0+\tau_d) < 0$: The mismatch term satisfies $2\lambda_{d,+}(t_0+\tau_d)(C_f[n]-1) = -2|\lambda_{d,+}(t_0+\tau_d)|(|C_f[n]|-1)$, so the calibrated signal becomes

$$y_{\mathrm{cal}} \approx -\lambda + \delta_g - 2|\lambda_{d,+}(t_0+\tau_d)|\,(|C_f[n]|-1) + |C_f[n]|\Delta V. \quad (14)$$

Setting $\Delta V = 2|\lambda_{d,+}(t_0+\tau_d)|$ yields $y_{\mathrm{cal}} \approx -\lambda + \delta_g + 2|\lambda_{d,+}| > -\lambda$, confirming that oscillation is suppressed.

(2) $C_f[n] < 0$, $\lambda_{d,+}(t_0+\tau_d) > 0$: The mismatch term satisfies $2\lambda_{d,+}(t_0+\tau_d)(C_f[n]-1) = -2|\lambda_{d,+}(t_0+\tau_d)|(|C_f[n]|+1)$, so the calibrated signal becomes

$$y_{\mathrm{cal}} \approx -\lambda + \delta_g - 2|\lambda_{d,+}(t_0+\tau_d)|\,(|C_f[n]|+1) + |C_f[n]|\Delta V. \quad (15)$$

Setting $\Delta V = 2|\lambda_{d,+}(t_0+\tau_d)|$ yields a bounded residual $y_{\mathrm{cal}} \approx -\lambda + \delta_g - 2|\lambda_{d,+}|$, which is independent of $|C_f[n]|$ and remains within the tolerance of the reconstruction algorithm. In both cases, selecting $\Delta V \approx 2|\lambda_{d,+}(t)|$ converts the mismatch-driven oscillation into a bounded constant residual, ensuring stability at arbitrary folding depth.

In practice, ΔV is set to a fixed constant calibrated as the empirical mean of the observed positive and negative threshold deviations, i.e., $\Delta V = \mathbb{E}[|\lambda_{d,+}(t)| + |\lambda_{d,-}(t)|]$, ensuring that the compensation remains effective across both polarities. The unified compensation formula is therefore

$$y_{\mathrm{cal}}(t_0+\tau_d) = y_a(t_0+\tau_d) - \mathrm{sgn}(\Delta C_f[n])\,|C_f[n]|\,\Delta V, \quad (16)$$

where $\Delta C_f[n] = C_f[n+1] - C_f[n] \in \{-1, +1\}$ indicates the folding direction: $\mathrm{sgn}(\Delta C_f[n]) = -1$ for a positive threshold crossing and $+1$ for a negative threshold crossing, ensuring that the compensation always opposes the

mismatch-induced displacement. Since the resulting residual is bounded and independent of $|C_f[n]|$, it remains within the tolerance of standard reconstruction algorithms.

Fig. 6 shows an experimental result from the fabricated modulo ADC platform. A folding threshold of $\lambda = 100$ mV and an under-compensation offset of $\Delta V = 10$ mV are selected. Before calibration, the folded waveform exhibits overshoot and oscillatory behavior at the folding edges. Applying controlled under-compensation stabilizes the folding transitions, which is suitable for subsequent signal recovery.

B. Bandwidth Limitation

The bandwidth limitation of the proposed system is fundamentally governed by the finite delay in the feedback loop. In an ideal modulo sampler, the folding action is applied instantaneously at the moment of threshold crossing. In practice, however, a non-negligible delay τ_d exists between threshold detection and DAC compensation. During this interval, the input signal continues to evolve, which may invalidate the folding operation.

To analyze this effect, consider a positive threshold crossing at time t_0, i.e., $y(t_0) = \lambda$. Due to the loop delay τ_d, the DAC compensation is applied at $t_0 + \tau_d$, where the input has evolved by $\delta_g = g(t_0 + \tau_d) - g(t_0)$. After one folding update, the modulo signal becomes

$$y(t_0 + \tau_d) = g(t_0 + \tau_d) - 2\lambda = -\lambda + \delta_g.$$

For this folding action to be valid, the modulo signal must lie within the admissible range $[-\lambda, \lambda]$, which requires

$$0 \le \delta_g \le 2\lambda.$$

This condition reveals two distinct failure mechanisms.

(1) Under-folding: If $\delta_g > 2\lambda$, the input increases so rapidly during τ_d that a single folding update is insufficient to bring the modulo signal back into the admissible range, yielding $y(t_0 + \tau_d) > \lambda$.

(2) Over-folding: If $\delta_g < 0$, the input reverses direction within the delay interval, so the compensation is applied after the signal has already begun to decrease, yielding $y(t_0 + \tau_d) < -\lambda$. This causes an erroneous folding event that corrupts the folding count.

Both failure mechanisms arise when the signal variation δ_g during τ_d becomes comparable to or exceeds the folding step size 2λ. For a bandlimited signal $g \in PW_\Omega$, the variation over the delay satisfies [46]–[48]

$$|\delta_g| \le 2 \sin\left(\frac{\Omega\tau_d}{2}\right) \|g\|_\infty.$$

To avoid under-folding, a sufficient condition is

$$2 \sin\left(\frac{\Omega\tau_d}{2}\right) \|g\|_\infty \le 2\lambda,$$

showing that the achievable bandwidth decreases with both the loop delay τ_d and the signal amplitude $\|g\|_\infty$.

Fig. 7 illustrates this failure mechanism using a 610 kHz triangular input. Due to the large slew rate, the input continues to rise significantly during τ_d, resulting in $\delta_g > 2\lambda$ and preventing the DAC update from fully compensating the signal excursion. Furthermore, the triangular waveform undergoes slope reversal within the delay interval, so the delayed update is applied after the input begins to decrease, yielding $\delta_g < 0$ and driving the modulo signal below $-\lambda$. The combined effect corrupts the folding count and ultimately limits the system bandwidth.

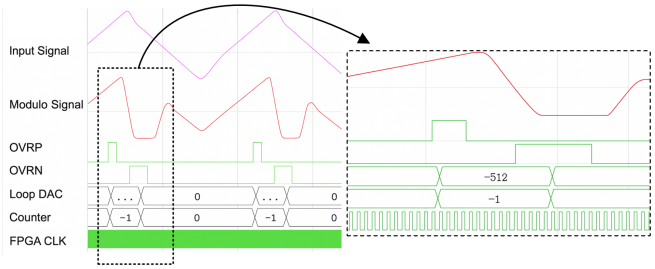

FIGURE 7. Timing-limited folding for a 610 kHz triangular input ($\lambda = 360$ mV). The full waveform (left) and a zoom near the threshold crossing (right) are shown. Loop delay causes overfolding below $-\lambda$, resulting in folding failure.

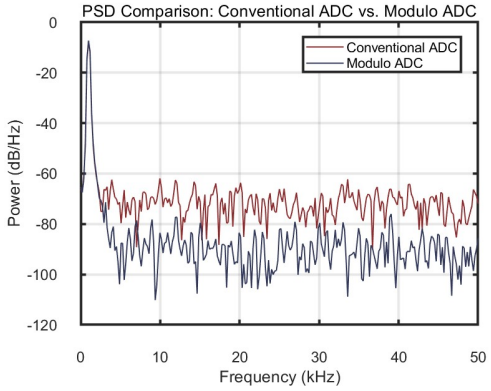

FIGURE 8. PSD of a 1 kHz sinusoidal input obtained using conventional ADC and modulo sampling system, with an effective resolution of $b = 7$ bits/sample.

V. Experimental Results

This section presents experimental results for the FPGA-based modulo ADC, focusing on dynamic range extension, folding fidelity under deep folding, and recovery robustness using real hardware data.

A. Comparison with Conventional ADCs

1) Spectral Purity and Noise Floor Analysis

Following the evaluation methodology in [15], [17], [49], we benchmark the acquisition performance of the proposed system against a conventional ADC under identical input conditions. Both systems employ $b = 7$-bit resolution for a 1 kHz sinusoidal input with peak amplitude 9.2 V. The conventional ADC operates at full scale to prevent clipping, while the modulo sampling system employs a folding threshold of $\lambda = 100$ mV, achieving a dynamic range expansion of $\rho = 92$. The folded signals, captured by the proposed hardware prototype, are reconstructed using the revised second-order difference (RSoD) method [29].

TABLE 5. Performance Metrics Comparison (SINAD, ENOB, SFDR, and THD) Between Conventional ADC and Modulo Sampling System ($\lambda = 100$ mV) Under Various Sinusoidal Input Frequencies f_m, Folding Depths ρ, and ADC Resolutions b.

ρ	f_m (kHz)	b (bits)	Conventional ADC				Modulo Sampling Systems			
			SINAD (dB)	ENOB (bits)	SFDR (dB)	THD (dB)	SINAD (dB)	ENOB (bits)	SFDR (dB)	THD (dB)
2.84	100	3	20.01	3.03	26.72	-30.75	**25.45**	**3.93**	**32.87**	**-39.07**
		5	**31.72**	**4.97**	**39.68**	**-42.13**	29.83	4.66	36.71	-41.72
		7	**39.45**	**6.26**	**48.26**	**-48.71**	30.75	4.81	38.71	-42.18
22.2	10	3	20.10	3.05	24.85	-31.12	**41.45**	**6.59**	**45.09**	**-50.13**
		5	31.78	4.98	38.68	-48.77	**46.73**	**7.47**	**54.24**	**-60.50**
		7	38.90	6.16	47.65	-49.62	**47.28**	**7.56**	**53.97**	**-60.84**
102	1	3	19.58	2.96	26.64	-27.90	**55.46**	**8.92**	**63.14**	**-62.75**
		5	31.42	4.92	39.45	-40.27	**57.47**	**9.25**	**64.19**	**-64.01**
		7	39.08	6.20	48.45	-52.74	**57.55**	**9.26**	**64.22**	**-64.05**

As illustrated by the power spectral density (PSD) in Fig. 8, the conventional ADC exhibits an elevated noise floor due to limited full-scale utilization. In contrast, the modulo sampling system folds the HDR signal into the optimal quantization range, preserving in-band spectral integrity. This architecture achieves an approximately 20 dB lower noise floor over the 5–45 kHz band compared to conventional operation. Such a spectral advantage confirms that the calibrated folding process effectively extends the dynamic range while avoiding the nonlinear distortions typically associated with signal clipping.

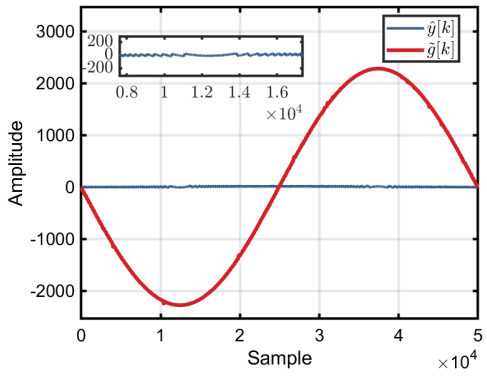

FIGURE 9. Folded signal $\hat{y}[k]$ and unfolded waveform $\tilde{g}[k]$ at $\rho = 92$ ($\lambda = 100$ mV).

2) Dynamic Range and Linearity Metrics

To quantitatively evaluate the proposed architecture, dynamic performance metrics including signal-to-noise and distortion ratio (SINAD), effective number of bits (ENOB), spurious-free dynamic range (SFDR), and total harmonic distortion (THD) are computed using standard MATLAB functions. These metrics are analyzed across various sinusoidal input frequencies f_m, folding depths ρ, and ADC resolutions b, as summarized in Table 5, where bold entries indicate the better-performing system for each configuration. The conventional ADC range is matched to the peak input amplitude to prevent clipping, while the modulo sampling system maintains a fixed threshold of $\lambda = 100$ mV with signal reconstruction performed using the RSoD method [29].

The results demonstrate that the performance advantage of the modulo system scales significantly with folding depth. Most notably, at $\rho = 102$, the modulo system with only 3-bit quantization achieves a SINAD of 55.46 dB (8.92 bits ENOB), substantially outperforming the 7-bit conventional ADC at 39.08 dB (6.20 bits ENOB). These results confirm that modulo sampling effectively concentrates quantization resolution within the folding threshold, maintaining high-fidelity acquisition at a fraction of the dynamic range required by conventional fixed-range converters.

3) Performance Validation via On-Board Acquisition

To evaluate the system performance in a fully integrated environment, the folded waveforms are digitized by the on-board AD9288 ADC and stored in the FPGA memory in real time. This experiment examines the hardware-level folding fidelity where both the analog folding front-end and the digital quantization process are active. As illustrated in Fig. 9, the folding threshold is set to ± 100 mV, corresponding to a digital representation of ± 25 LSBs.

At $\rho = 92$, the modulo sampling system achieves a SINAD of 44 dB, closely aligning with the AD9288's nominal 46 dB SINAD within its linear range. The marginal 2 dB discrepancy confirms that the calibrated folding process introduces negligible noise and distortion even under extreme HDR conditions. This validates that the architecture maintains near-ideal fidelity while extending the effective dynamic range by nearly two orders of magnitude.

B. Reconstruction on Hardware Folded Data

Four representative reconstruction methods are tested on experimentally folded hardware data: difference-based (RSoD [29]), iterative (ITER-SIS [30]), optimization-based (LASSO–B^{2}R^2 [31]), and spectral (USLSE [32]).

TABLE 6. Reconstruction of sine and periodic sinc signals versus ρ, B, and f_m ($\lambda = 100$ mV). SNR is used as the performance metric; OF denotes the oversampling factor.

Input Signal Type	Input Signal Parameters		RSoD [29]		ITER-SIS [30]		LASSO-B^{2}R^2 [31]		USLSE [32]	
Sine	ρ	f_m (kHz)	OF	SNR (dB)	OF	SNR (dB)	OF	SNR (dB)	OF	SNR (dB)
	2.84	100	6.58	23.44	11.11	29.44	8.33	23.38	7.04	22.44
	22.2	10	17.86	22.57	71.42	20.24	71.42	22.30	62.5	20.54
	102	1	41.67	23.37	62.5	6.37	50	0.49	50	0.33
Sinc	ρ	B, f_m (kHz)	OF	SNR (dB)	OF	SNR (dB)	OF	SNR (dB)	OF	SNR (dB)
	3.24	410, 23	3.81	24.31	2.97	23.86	3.93	22.22	3.48	21.76
	9.16	99, 5	7.01	23.09	11.48	27.79	12.63	22.78	12.63	24.31
	29.80	18, 1	15.43	16.43	46.30	17.13	46.30	16.07	46.30	13.47
Bandlimited	ρ	B (kHz)	OF	SNR (dB)	OF	SNR (dB)	OF	SNR (dB)	OF	SNR (dB)
	4.19	10	4.46	29.64	4.46	32.84	4.17	29.81	4.03	28.05
	6.91	100	5.95	20.72	13.89	24.32	20.83	20.39	20.83	20.31
	22.13	1	25.00	22.64	31.25	19.11	62.5	30.11	37.87	29.18

TABLE 7. Reconstruction results for modulated test waveforms under varying B and ρ (SNR as performance metric; OF indicates sampling condition).

Input Signal Parameters			RSoD [29]		ITER-SIS [30]		LASSO-B^{2}R^2 [31]		USLSE [32]	
Type	ρ	B (kHz)	OF	SNR (dB)	OF	SNR (dB)	OF	SNR (dB)	OF	SNR (dB)
QAM	10.40	4	12.50	28.06	25.00	34.70	25.00	23.10	25.00	20.88
BPSK	5.20	2	17.86	23.44	22.73	33.64	22.73	23.39	19.23	23.55
FSK	8.00	2	20.83	23.24	20.83	33.76	20.83	22.63	41.67	21.48

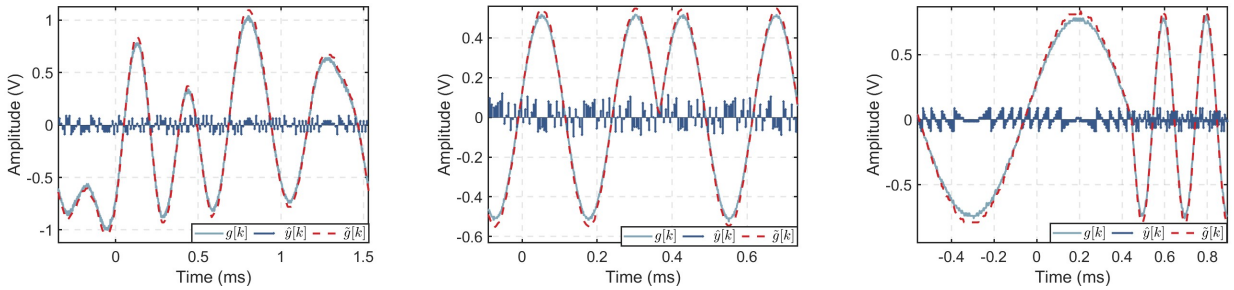

FIGURE 10. Waveforms reconstructed by RSoD [29] from folded baseband modulated test signals: (a) QAM, (b) BPSK, and (c) FSK.

Performance is quantified by the SNR between the ground-truth signal $g[k]$ and the reconstructed signal $\tilde{g}[k]$,

$$\mathrm{SNR} = 10\log_{10}\frac{\sum_k |g[k]|^2}{\sum_k |g[k]-\tilde{g}[k]|^2}. \tag{17}$$

The input signal and folded waveform were acquired using a Tektronix TDS 1012C-EDU 8-bit oscilloscope.

1) Basic Test Signals

Single-tone sinusoids, periodic sinc, and general bandlimited signals were used to evaluate modulo folding performance. The dynamic range factor ρ, signal frequency f_m, and main-lobe bandwidth B were varied to induce different folding depths, while the folding threshold was fixed at $\lambda = 100$ mV. The oversampling factor (OF), defined as $\mathrm{OF} = f_s/f_{\mathrm{Nyq}}$ with f_{Nyq} denoting the Nyquist frequency, indicates the sampling requirement for successful reconstruction. Performance is quantified using the SNR defined in (17).

As summarized in Table 6, all reconstruction methods achieve similar SNR at moderate folding depths ($\rho < 10$). At larger folding depths ($\rho > 100$), only the RSoD method reliably reconstructs the signal, while the performance of other approaches degrades.

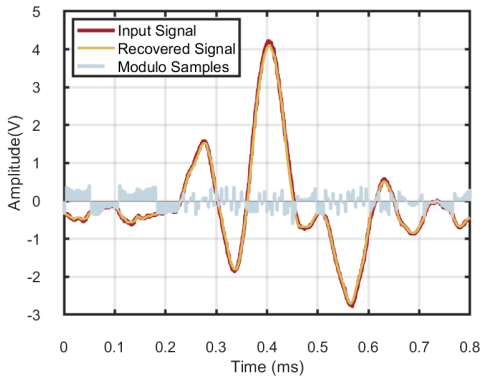

FIGURE 11. Reconstruction from noisy folded measurements ($B = 10$ kHz, $\rho = 11.80$, $\lambda = 360$ mV, OF $= 10.00$) at input SNR $= 15$ dB, achieving a reconstruction SNR $= 29.32$ dB .

2) Modulated Test Waveforms

To examine robustness under structured wideband excitation, modulated signals including quadrature amplitude modulation (QAM), binary phase-shift keying (BPSK), and frequency-shift keying (FSK) were evaluated. These waveforms were generated as baseband signals in the experimental setup. As summarized in Table 7, all reconstruction methods achieve reliable recovery, confirming stable hardware folding under practical conditions. Representative waveforms reconstructed using RSoD are shown in Fig. 10.

C. Noise Robustness Evaluation

To assess reconstruction performance under noisy conditions, additive white Gaussian noise is introduced to a bandlimited input with $B = 10$ kHz, $\rho = 11.80$, $\lambda = 0.36$ V, and OF $= 10.00$. As shown in Fig. 11, the RSoD algorithm successfully recovers the original waveform at an input SNR $= 15$ dB, achieving a reconstruction SNR of 29.32 dB, confirming the robustness of the calibrated folding architecture under practical noisy acquisition conditions.

VI. Conclusion and Future Work

This paper presents a calibrated FPGA-based modulo ADC platform for robust HDR signal acquisition. The proposed architecture replaces instantaneous comparator-triggered feedback with a state-space controlled folding mechanism governed by a finite state machine with multi-bit update control, ensuring deterministic and stable folding dynamics. A controlled under-compensation calibration strategy is further introduced to convert fold-dependent threshold mismatch and oscillatory instability into a bounded constant residual, enabling reliable operation at high folding depths. Together, these contributions overcome the practical hardware constraints that have limited prior modulo sampling prototypes, namely restricted folding depth due to limited DAC resolution, loop-latency-induced instability, and the absence of systematic experimental validation at large folding depths.

Experimental results confirm that the platform achieves dynamic range expansion exceeding two orders of magnitude while maintaining signal fidelity comparable to that of the standalone ADC within its linear range. Reconstruction of diverse waveforms including sinusoidal, bandlimited, and modulated signals further validates the platform's reliability under practical wideband excitation. As a proof-of-concept baseline, this work establishes a high-fidelity single-channel hardware foundation for modulo ADC systems.

Future work will focus on reducing loop latency to extend the operational bandwidth, and on leveraging the platform's multi-channel capabilities to investigate synchronized HDR acquisition for array-based sensing.

DATA AVAILABILITY STATEMENT

The data supporting the findings of this study are included within the article. No additional datasets were generated or analysed.

OPEN ACCESS STATEMENT

For the purpose of open access, the authors have applied a Creative Commons Attribution (CC BY) licence to any Author Accepted Manuscript version arising from this submission.

REFERENCES

[1] A. Esmailiyan, J. Du, T. Siriburanon, F. Schembari, and R. B. Staszewski, "Dickson-charge-pump-based voltage-to-time conversion for time-based ADCs in 28-nm CMOS," *IEEE Open J. Circuits Syst.*, vol. 2, pp. 23–31, 2021.

[2] M. Bagheri, F. Schembari, H. Zare-Hoseini, R. B. Staszewski, and A. Nathan, "Interchannel mismatch calibration techniques for time-interleaved SAR ADCs," *IEEE Open J. Circuits Syst.*, vol. 2, pp. 420–433, 2021.

[3] L. Zhang, B. Gao, K.-W. Park, K. Edrian Lozada, R. Mabilangan, H. Kim, J. Wu, and S.-T. Ryu, "DR loss-free dithering-based digital background linearity calibration for SAR-assisted multi-stage ADCs with digital input-interference cancellation," *IEEE Open J. Circuits Syst.*, vol. 5, pp. 349–364, 2024.

[4] G. Kim, "Design space exploration of single-lane OFDM-based serial links for high-speed wireline communications," *IEEE Open J. Circuits Syst.*, vol. 3, pp. 134–146, 2022.

[5] R. Koma, M. Fujiwara, S. Kimura, and N. Yoshimoto, "Wide dynamic range reception of TDM-based DCO-OFDM signals using optical domain power equalization and symmetrical clipping techniques," *IEEE J. Lightw. Technol.*, vol. 33, no. 8, pp. 1623–1629, 2015.

[6] Y. M. Jo, D. H. Woo, S. G. Kang, and H. C. Lee, "Very wide dynamic range ROIC with pixel-level ADC for SWIR FPAs," *IEEE Sensors J.*, vol. 16, no. 19, pp. 7227–7233, 2016.

[7] G. Guarnieri, S. Marsi, and G. Ramponi, "High dynamic range image display with halo and clipping prevention," *IEEE Trans. Image Process.*, vol. 20, no. 5, pp. 1351–1362, 2011.

[8] M. Laghaei, H. Shakiba, and A. Sheikholeslami, "A companding technique to reduce peak-to-average ratio in discrete multitone wireline receivers," *IEEE Open J. Circuits Syst.*, vol. 5, pp. 302–313, 2024.

[9] D. Dardari, "Joint clip and quantization effects characterization in OFDM receivers," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 53, no. 8, pp. 1741–1748, 2006.

[10] L. Cai, X. Song, Z. Lu, X.-P. Yu, K.-S. Yeo, J.-M. Chen, and B. K. Thangarasu, "A linear-in-decibel automatic gain control amplifier with dual mode continuous gain tuning," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 70, no. 7, pp. 2752–2761, 2023.

[11] M. El-Shennawy, N. Joram, and F. Ellinger, "Nonlinear modelling of automatic gain control loops considering loop dynamics and stability,"

in *Proc. IEEE Conf. Ph.D. Res. Microelectron. Electron. (PRIME)*, pp. 1–4, 2016.
[12] A. Bhandari, F. Krahmer, and R. Raskar, "On unlimited sampling," in *Proc. Int. Conf. Sampling Theory Appl. (SampTA)*, pp. 31–35, 2017.
[13] A. Bhandari, F. Krahmer, and R. Raskar, "Unlimited sampling of sparse signals," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, pp. 4569–4573, 2018.
[14] A. Bhandari, F. Krahmer, and R. Raskar, "On unlimited sampling and reconstruction," *IEEE Trans. Signal Process.*, vol. 69, pp. 3827–3839, 2020.
[15] A. Bhandari, F. Krahmer, and T. Poskitt, "Unlimited sampling from theory to practice: Fourier–Prony recovery and prototype ADC," *IEEE Trans. Signal Process.*, vol. 70, pp. 1131–1141, 2022.
[16] S. Mulleti, E. Reznitskiy, S. Savariego, M. Namer, N. Glazer, and Y. C. Eldar, "A hardware prototype of wideband high-dynamic-range analog-to-digital converter," *IET Circuits Devices Syst.*, vol. 17, no. 4, pp. 181–192, 2023.
[17] J. Zhu, J. Ma, Z. Liu, F. Qu, Z. Zhu, and Q. Zhang, "A modulo sampling hardware prototype and reconstruction algorithms evaluation," *IEEE Trans. Instrum. Meas.*, vol. 74, pp. 1–11, 2025.
[18] D. Florescu, F. Krahmer, and A. Bhandari, "The surprising benefits of hysteresis in unlimited sampling: Theory, algorithms, and experiments," *IEEE Trans. Signal Process.*, vol. 70, pp. 616–630, 2022.
[19] D. Florescu and A. Bhandari, "Unlimited sampling with local averages," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, pp. 5742–5746, 2022.
[20] Y. Zhu, R. Guo, and A. Bhandari, "Ironing the modulo folds: Unlimited sensing with 100× bandwidth expansion," in *Proc. Int. Conf. Sampling Theory Appl. (SampTA)*, 2025.
[21] R. Bolaños-Pérez, J. M. Rocha-Pérez, A. Díaz-Sánchez, J. Ramirez-Angulo, and E. Tlelo-Cuautle, "CMOS analog AGC for biomedical applications," *Electronics*, vol. 9, no. 5, p. 878, 2020.
[22] M. Rovetta, J. F. R. Baggio, and R. Moraes, "An automatic gain control circuit to improve ECG acquisition," *Res. Biomed. Eng.*, vol. 33, no. 4, pp. 378–389, 2017.
[23] T. Johnson and S. P. Stapleton, "RF class-D amplification with bandpass sigma–delta modulator drive signals," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 53, no. 12, pp. 2507–2520, 2006.
[24] Z. Tan, C.-H. Chen, Y. Chae, and G. C. Temes, "Incremental Delta–Sigma ADCs: A tutorial review," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 67, no. 12, pp. 4161–4173, 2020.
[25] D. L. Carni, D. Grimaldi, and L. Serratore, "Sigma–Delta ADC-based frequency-error measurement in single-carrier digital modulations," *IEEE Trans. Instrum. Meas.*, vol. 55, no. 5, pp. 1523–1529, 2006.
[26] B. Yang, T. Caldwell, and A. Chan Carusone, "An energy-efficient pipeline-SAR ADC using linearized dynamic amplifiers and input buffer in 22nm FDSOI," *IEEE Open J. Circuits Syst.*, vol. 6, pp. 50–62, 2025.
[27] F. Yuan, "A comparative study of dynamic comparators for low-power successive approximation ADC," *IEEE Open J. Circuits Syst.*, vol. 6, pp. 241–256, 2025.
[28] K. E. Lozada, B. Gao, R. Mabilangan, C. Tahar, K.-W. Park, Y.-H. Moon, K.-H. Song, and S.-T. Ryu, "A SAR-assisted continuous-time M-0 MASH delta-sigma modulator with digital-domain noise leakage shaping," *IEEE Open Journal of Circuits and Systems*, pp. 1–1, 2026.
[29] W. Yan, Z. Li, L. Gan, H. Liu, and G. Li, "Difference-based recovery for modulo sampling: Tightened bounds and robustness guarantees," *arXiv preprint arXiv:2509.12971*, 2025.
[30] R. Guo and A. Bhandari, "ITER-SIS: Robust unlimited sampling via iterative signal sieving," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, pp. 1–5, 2023.
[31] S. B. Shah, S. Mulleti, and Y. C. Eldar, "LASSO-based fast residual recovery for modulo sampling," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, pp. 1–5, 2023.
[32] Q. Zhang, J. Zhu, F. Qu, and D. W. Soh, "Line spectral estimation via unlimited sampling," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 60, no. 5, pp. 7214–7231, 2024.
[33] V. Pavlíček and A. Bhandari, "1-bit unlimited sampling beyond fourier domain: Low-resolution sampling of quantization noise," *IEEE Journal of Selected Topics in Signal Processing*, vol. 19, no. 6, pp. 1133–1145, 2025.
[34] O. Graf, A. Bhandari, and F. Krahmer, "One-bit unlimited sampling," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5102–5106, 2019.
[35] A. Eamaz, K. V. Mishra, F. Yeganegi, and M. Soltanalian, "UNO: Unlimited sampling meets one-bit quantization," *IEEE Transactions on Signal Processing*, vol. 72, pp. 997–1014, 2024.
[36] A. Eamaz, K. V. Mishra, F. Yeganegi, and M. Soltanalian, "Unlimited sampling via one-bit quantization," in *2023 International Conference on Sampling Theory and Applications (SampTA)*, pp. 1–5, 2023.
[37] A. Eamaz, F. Yeganegi, and M. Soltanalian, "Streamlining UNO: A generalized sampling approach to optimal one-bit modulo sensing," in *ICASSP 2025 - 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5, 2025.
[38] A. Eamaz, F. Yeganegi, K. V. Mishra, and M. Soltanalian, "Unlimited sampling of FRI signals with dithered one-bit quantization," in *2023 57th Asilomar Conference on Signals, Systems, and Computers*, pp. 1430–1435, 2023.
[39] A. Bhandari and F. Krahmer, "HDR imaging from quantization noise," in *2020 IEEE International Conference on Image Processing (ICIP)*, pp. 101–105, 2020.
[40] A. Eamaz, F. Yeganegi, and M. Soltanalian, "HDR imaging with one-bit quantization," in *2024 IEEE 13rd Sensor Array and Multichannel Signal Processing Workshop (SAM)*, pp. 1–5, 2024.
[41] L. Gan and H. Liu, "High dynamic range sensing using multi-channel modulo samplers," in *2020 IEEE 11th Sensor Array and Multichannel Signal Processing Workshop (SAM)*, pp. 1–5, 2020.
[42] Y. Gong, L. Gan, and H. Liu, "Multi-channel modulo samplers constructed from gaussian integers," *IEEE Signal Processing Letters*, vol. 28, pp. 1828–1832, 2021.
[43] W. Yan, L. Gan, S. Hu, and H. Liu, "Towards optimized multi-channel modulo-ADCs: Moduli selection strategies and bit depth analysis," in *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 9496–9500, 2024.
[44] W. Yan, L. Gan, and Y. D. Zhang, "Threshold sensitivity in two-channel modulo ADCs: Analysis and robust reconstruction," in *ICASSP 2025 - 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5, 2025.
[45] Z. Li, W. Yan, R. Zhu, L. Gan, and H. Liu, "Live demonstration: Real-time high-amplitude signal acquisition with 2-channel modulo ADC," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, 2025.
[46] S. Bernstein, *Sur l'ordre de la meilleure approximation des fonctions continues par des polynômes de degré donné*, vol. 4. Hayez, imprimeur des académies royales, 1912.
[47] A. Aleksandrov and V. V. Peller, "Functions of perturbed unbounded self-adjoint operators. operator bernstein type inequalities," *Indiana University Mathematics Journal*, pp. 1451–1490, 2010.
[48] B. Y. Levin, *Lectures on entire functions*, vol. 150. American Mathematical Soc., 1996.
[49] Y. Zhu and A. Bhandari, "Unleashing dynamic range and resolution in unlimited sensing framework via novel hardware," in *2024 IEEE SENSORS*, pp. 1–4, 2024.