dAJC: A 2.02-mW 50-Mb/s Direct Analog to MJPEG Converter for Video Sensor Nodes Using Switched Capacitor MAC-Quantizer With Process Calibration

Gourab Barik, *Graduate Student Member, IEEE*, Gaurav Kumar K., *Graduate Student Member, IEEE*, Baibhab Chatterjee, *Member, IEEE*, Shovan Maity, *Member, IEEE*, Sumon Bose, *Member, IEEE*, and Shreyas Sen, *Senior Member, IEEE*

***Abstract*—With the advancement in the field of the Internet of Things (IoT) and Internet of Bodies (IoB), video camera applications utilizing video sensor nodes (VSNs) have become increasingly significant. These applications are pivotal in areas such as autonomous driving, health monitoring, machine vision, and security. Notable examples include their use in uncrewed aerial vehicles (UAVs), body-worn cameras, and surveillance systems. However, these applications often face challenges related to power consumption, primarily due to the large data volumes generated by analog-to-digital converters (ADCs) during the transmission of digitized high-resolution video signals. This data deluge introduces additional processing and storage overheads, exacerbating the problem of resource-constrained sensor nodes having limited battery capacity. To address this challenge, we propose a low-power solution that performs video compression directly at the sensor while shifting computation from the digital domain to the energy-efficient analog domain (suitable for peak signal-to-noise ratio (PSNR) values below 40–50 dB). Unlike conventional architectures, which rely on digital signal processing (DSP) blocks for computation and compression after the ADC stage, our approach eliminates the need for these power-intensive digital blocks at the sensor. Instead, we employ a switched capacitor (SC)-based computation unit operating in the analog domain, significantly reducing overall power consumption. Additionally, we utilize a dynamically activated ADC that activates only for significant samples, which constitute a small fraction (≤5%) of the total captured analog video samples. This approach enables us to achieve approximately ∼20× lower analog-to-digital (A/D) conversion energy. By combining analog computation with the dynamically activated ADC, our proposed solution achieves at least >2× reduction in power consumption compared to a digital implementation, with only a ∼5-dB degradation in PSNR—an imperceptible difference to the human eye. We also propose a calibration scheme to mitigate the impact of variations introduced by analog computation, thereby enhancing the PSNR of the decoded video or image.**

***Index Terms*—Analog computing, discrete cosine transform (DCT), dynamically activated analog-to-digital converter (ADC), human body communication (HBC), in-sensor computing, Motion Joint Photographics Experts Group (MJPEG), multiply accumulation (MAC), quantization matrix (Q-matrix), process calibration, switched capacitor (SC), video sensor nodes (VSNs).**

I. Introduction

EMERGING applications in the Internet of Things (IoT) field, such as machine vision, health monitoring, autonomous driving, and security cameras, necessitate the capture of high-resolution video data [1]. Traditionally, these captured analog video data are digitized using front-end analog-to-digital converters (ADCs) before further processing or communication, consuming a significant amount of ADC energy during conversion while generating enormous data volumes.

Standard wireless communication technologies, such as Bluetooth, while energy efficient, are limited by their low data transfer rates, making them unsuitable for handling large volumes of data generated by a video sensor node (VSN). In contrast, 2.4- and 5-GHz Wi-Fi systems offer significantly higher data rates up to 600 Mb/s and 1.3 Gb/s [2], respectively, making them capable of managing this data-intensive task. However, these higher data rates come at the cost of increased energy consumption, with a high pJ/bit communication energy [3], [4]. This results in substantial power requirements for communication, creating a critical bottleneck for the widespread deployment of VSNs in resource-constrained environments.

The current demand for VSNs is threefold. They must handle high data volumes, maintain a compact form factor, and consume low power/energy. Additionally, they need to efficiently communicate captured video data with nearby hubs, highlighting the importance of efficient communication and computing mechanisms in these devices [5]. However, detailed communication aspects are beyond the scope of this article.

Received 8 March 2024; revised 12 December 2024, 18 April 2025, and 8 June 2025; accepted 4 July 2025. This article was approved by Associate Editor Priyanka Raina. This work was supported by Quasistatics Inc. *(Corresponding author: Shreyas Sen.)*

Gourab Barik, Gaurav Kumar K., and Shreyas Sen are with the Elmore Family School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN 47907 USA (e-mail: shreyas@purdue.edu).

Baibhab Chatterjee is with the Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL 32611 USA.

Shovan Maity and Sumon Bose are with Quasistatics Inc., West Lafayette, IN 47906 USA.

Color versions of one or more figures in this article are available at https://doi.org/10.1109/JSSC.2025.3588828.

Digital Object Identifier 10.1109/JSSC.2025.3588828

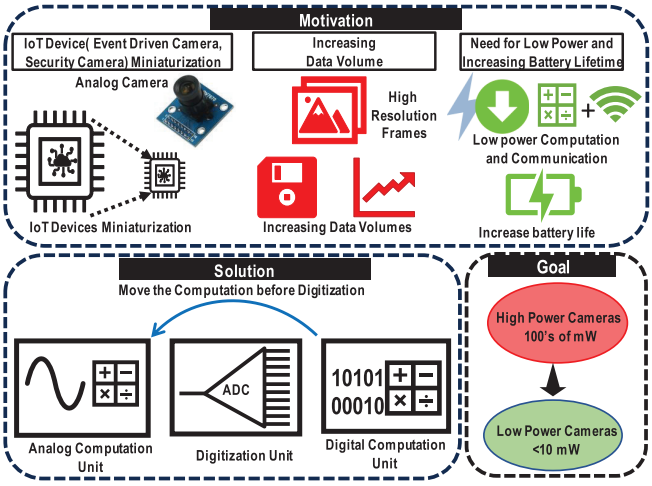


Fig. 1. Miniaturization of IoT devices and implementation of cameras in smaller form factor along with the increase in the data volume due to increasing video frame size requires low-power processing cameras to remove the need for frequent battery replacement and extend battery lifetime.

As shown in Fig. 1, the shrinking size of IoT devices results in smaller batteries, leading to challenges such as frequent recharging or replacement. Although power gating and other low-power design techniques mitigate power consumption, higher video quality and faster frame rates significantly increase data processing demands and power usage. Therefore, an alternative solution is needed to handle the rising data rates within a constrained power budget, meeting the energy demands of these devices.

In a typical IoT system, wirelessly transmitting sensor data to a processing hub for storage requires careful consideration of communication power to optimize energy usage [6]. As shown in Eq. (1), the power consumption for wireless transmission is directly proportional to the number of transmitted bits and the energy efficiency of the communication method [7]. For example, conventional Wi-Fi communication achieves an energy efficiency of 1–10 nJ/bit

$$\text{Tx power} = \text{No. of bits} \times \text{Energy Efficiency}. \quad (1)$$

Considering a scenario where a 2K RGB video at 30 frames/s with 8-bit digitization results in a substantial data volume of 1.5 Gb/s. Transmitting these data via Wi-Fi would consume 1.5–15 W of power according to Eq. (1). However, employing compression techniques such as Motion Joint Photographics Experts Group (MJPEG) [8] can significantly reduce the data rate to 600 Mb/s, thus lowering the power consumption to 0.6–6 W during Wi-Fi transmission. This demonstrates a notable decrease in communication power through video compression before communication.

Moreover, adopting more energy-efficient communication modes, such as human body communication (HBC) or ultra-wideband (UWB), offers additional power savings. HBC, suitable for body-worn devices, and UWB, ideal for personal area networks (PANs), boast energy efficiencies of 10–100 pJ/bit, making them 100 times more efficient than standard Wi-Fi. While HBC [9] may not support gigabit-per-second data rates, its suitability for transmitting compressed data in the megabit-per-second range ensures better communication energy efficiency.

In this work, we propose in-sensor data compression in the analog domain [10], [11] using switched capacitor (SC) circuits, performed before ADC. Compressing analog video data at this stage reduces the overall communication data volume, eliminating the need for digital signal processing (DSP)-based compression units post-ADC. Additionally, we employ a dynamically activated ADC to improve energy efficiency by digitizing only significant samples, exploiting the inherent spatial sparsity in compressed image data. This integrated approach minimizes power consumption, reduces storage overhead, and extends battery life, ensuring sustained performance in high-data-rate IoT applications.

As shown in Fig. 2, a typical VSN employs an image sensor that captures images as analog voltage samples. Standard cameras, which consume 50–100 mW of power, utilize a front-end ADC to convert the analog samples into digital bits, followed by a digital compression engine to reduce the number of bits transmitted. To estimate the overall data volume flow in the standard architecture (front-end ADC + digital MJPEG engine), consider a grayscale video at 2K resolution and 30 frames/s, which generates data at a rate of 186 MS/s. Using an 8-bit ADC, this results in a data volume of 1.5 Gb/s. The data are then compressed using the compression engine at a factor of 20×, reducing the data to 75 Mb/s. However, this process results in high power consumption in the DSP and requires increased storage and processing overhead.

In contrast, our approach, dAJC with dynamically activated ADC, performs in-sensor compression of the sensed analog image samples. The analog samples, captured at 186 MS/s, are first compressed in the analog domain to 9.33 MS/s before digitization. Since this compression uses an SC-based analog MJPEG encoding scheme, the overall power consumption is significantly lower than that of digital compression. Additionally, the compressed analog samples exhibit sparsity, containing only a few significant samples, while the remaining samples are small and insignificant. These insignificant samples can be discarded without any perceivable degradation to the image. The dynamically activated ADC exploits this inherent sparsity by digitizing only the significant samples while remaining inactive for insignificant data, thereby reducing the overall analog-to-digital (A/D) conversion energy.

The implemented system is designed for JPEG compression of analog images, where image pixels are represented as analog voltages [12]. This system can be extended to videos that rely on intraframe dependencies, such as MJPEG. Other compression methods, like H.264 [13], leverage inter-frame redundancies and offer higher compression ratios. However, H.264 has a complex decoding scheme and poor error tolerance, making it unsuitable for latency-sensitive applications and those operating in error-prone communication environments. In contrast, MJPEG benefits from per-frame compression, offers better error tolerance, and has a less complex decoding scheme, making it ideal for real-time inference and decision-making. Furthermore, MJPEG enables simpler hardware design compared to H.264, making it more suitable for energy-constrained nodes.

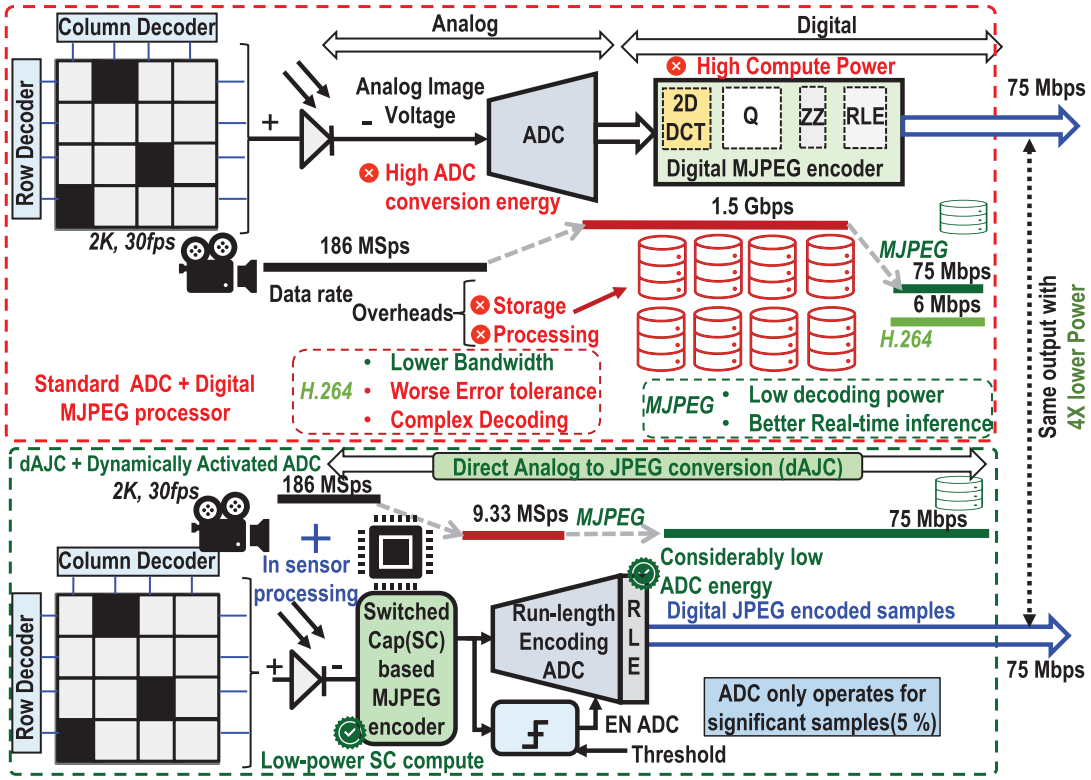


Fig. 2. Comparison between standard MJPEG compression approaches against the dAJC approach (presented in this article). dAJC has the advantage of compressing the information in the analog domain before digitizing, which reduces the intermediate data (reducing intermediate storage and processing elements). The addition of the dynamically activated ADC reduces the ADC energy since the ADC is turned on only for significant samples.

A. Contributions of Our Work

The key contributions of this proposed work are as follows:

1) This work, dAJC, presents the first end-to-end implementation of MJPEG compression in the mixed-signal domain using SC circuits and dynamically activated ADC.
2) We achieve > 2× lower power consumption than its digital counterparts, illustrating the benefits of analog computing.
3) The SC-based low-power 2-D discrete cosine transformation (2D-DCT) core consumes ∼ 12× less power compared to other reported implementations in the literature.
4) The dynamically activated ADC provides > 20× reduction in A/D conversion energy by converting only significant JPEG compressed samples, which typically constitute ≤ 5% of the total captured analog samples.
5) We proposed a *process calibration* scheme to update the quantization matrix (Q-matrix) during decoding, compensating for process variations and parasitic effects in the SC design.
6) Additionally, the power expended for communication/transmission is lowered by compressing the signal in the analog domain, with the communication energy decreasing linearly as the number of transmitted bits is reduced.

B. Organization of This Article

The remainder of this article is organized as follows. Section II discusses the proposed dAJC architecture and provides a detailed circuit diagram of all the JPEG compression blocks, including 2D-DCT, quantization, zig-zag (ZZ) traversal, run-length encoder (RLE), and dynamically activated ADC, along with the non-overlapping phase generation circuit that performs the JPEG compression. In Section III, various design considerations are examined, including capacitor sizing for the discrete cosine transform (DCT) matrix and the Q-matrix, image performance evaluation metrics using peak signal-to-noise ratio (PSNR), variation tolerance using Q-sense, and the overall input image processing performed for testing purposes. Section IV presents the measurement results for the dAJC integrated circuit (IC). Section V compares the proposed design with the related works reported in the literature and its digital implementation. Section VI discusses noise analysis and fundamental energy limitations, the scope and impact of the dynamically activated ADC, future prospects for Q-matrix calibration, and the potential application scenario of dAJC. Section VII concludes this article.

II. Proposed dAJC Architecture and Detailed Circuit-Level Implementation

This section discusses the proposed architecture of MJPEG video compression [14] implemented on the dAJC IC, highlighting underlying operations such as 2D-DCT, quantization, ZZ traversal, and run-length encoding, as shown in Fig. 3 and explaining their circuit-level SC-based implementation. We have chosen the voltage-mode implementation using SC-based circuits.

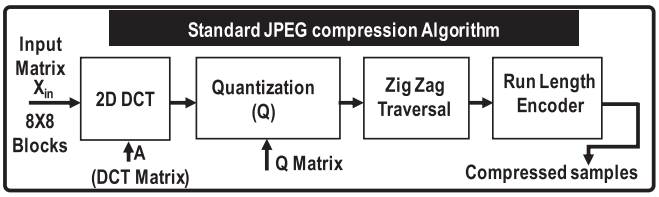


Fig. 3. Standard JPEG compression algorithm and its constituting blocks.

Fig. 4(a) shows the overall architecture consisting of the detailed circuit diagram of the 2D-DCT block, Q-matrix, ZZ traversal block, and RLE + dynamically activated ADC. Fig. 4(b) shows the cycle-by-cycle operation of the implemented 2D-DCT for a 4 × 4 input matrix size.

A. 2-D Discrete Cosine Transformation

The first step of JPEG compression consists of performing DCT [15], [16], which transforms the image data from the spatial domain to the frequency domain. Due to the sparsity of the DCT basis, image information in the frequency domain is sparse, meaning that the frequency-domain representation contains more zeros than non-zero values. As a result, the original image data can be represented with fewer samples [17]. Mathematically, DCT is a 2-D matrix of DCT coefficients applied to blocks of 2-D 8 × 8 input image matrices. Thus, the 2D-DCT operation involves two-stage matrix multiplication, as shown in the following equation:

$$[Z]_{\text{2D-DCT}} = [C].[X_{\text{in}}].[C^{\text{T}}] \tag{2}$$

where C represents the DCT matrix, X_{in} represents the input matrix, and C^{T} represents the transpose of the DCT matrix.

1) SC-Based MAC: Since the 2D-DCT operation involves matrix multiplication, it consists of MAC operations to generate the $Z_{\text{2D-DCT}}$ matrix, which can be implemented using SC-based circuits. The realization of the MAC operation using an SC circuit is shown in Fig. 5 for an SC-based MAC unit. In phase ϕ_0, voltages V_1 and V_2 are sampled onto capacitors C_1 and C_2, respectively. In phase ϕ_1, the voltages on capacitors C_1 and C_2 are accumulated onto capacitor C_2 (accumulation capacitor), generating the output voltage V_{out}, which is the resulting MAC operation of voltages V_1 and V_2. The multiplication/DCT coefficients a_1 and a_2 can be controlled by adjusting the values of C_0, C_1, and C_2.

2) Stage-1 and Stage-2 DCT Operation: To perform the 2D-DCT on the input matrix, we divide the process into two stages: *Stage-1 DCT* and *Stage-2 DCT* as shown in Fig. 4(a). The *Stage-1 DCT* block receives a column-wise serial input of the image and performs a MAC operation with the DCT matrix. The intermediate result $[C] \times [X_{\text{in}}]$ is obtained at the output of the *Stage-1 DCT* block. Fig. 4(b) shows the cycle-by-cycle 2D-DCT operation for a 4 × 4 matrix. For 4 × 4 2D-DCT computations, four slices, each containing capacitors to store the DCT coefficient values C_{ij} (for $i = 0$ and $j = 0$–3) in each slice $i = 0$–3, are used. These capacitors act as the memory for storing the DCT coefficients, and their sizing is discussed in detail in Section III.

The different partial products are calculated and stored on the corresponding capacitors in the slices during subsequent, non-overlapping cycles. Thus, during the ith cycles, i partial products are stored in the ith slice. In the fifth cycle, the partial products are summed up and stored in a larger accumulator capacitor. Note that the reset phase of the accumulator capacitor is merged with one of the sampling phases since they are fundamentally non-overlapping. The outputs of *Stage-1 DCT* are fed in parallel to *Stage-2 DCT*, which performs the remaining 4 × 4 matrix multiplication of the result ($[C] \times [X_{\text{in}}]$) with the transpose of the DCT matrix ($[C^{\text{T}}]$). Each partial product computation in *Stage-2 DCT* depends on the output of *Stage-1 DCT*, and thus, we require a total of $(4 \times 4) + 4$ cycles to complete the 2D-DCT operation. The outputs of the 2D-DCT block are read row-wise to emulate the transpose of the DCT matrix.

Extending this for the JPEG algorithm with 8 × 8 matrix, inputs are fed in 8 × 1 form, where each set of partial product calculations takes one cycle. This requires eight cycles to perform 8 × 8 multiplication, and one additional cycle is required to accumulate the voltage on the accumulator capacitor. The accumulator capacitor is then reset during one of the sampling phases before the result is fed to the next stage. The output of *Stage-1 DCT* is fed to *Stage-2 DCT* through an intermediate buffer, which provides signal gain and helps improve the system's SNR, as discussed in detail in the following. Consequently, *Stage-1 DCT* takes a total of nine data cycles, eight cycles for sampling, and one for accumulation to produce the output for each input column as shown in Fig. 6. The *Stage-2 DCT* block operates on a slower clock than the *Stage-1 DCT* block to sample its output. Therefore, the total time to perform 2D-DCT of 8 × 8 block is 72 cycles (9 cycles ×8).

Since the computations are implemented using SC circuits and clock phases, issues such as clock feed-through and charge injection are mitigated using standard circuit techniques. These techniques include employing transmission gates as switches and utilizing dummy capacitors. Additionally, a differential architecture is used to cancel out common-mode switching noise. A non-overlapping clock generator generates the non-overlapping clock phases required for sampling, accumulation, and resetting, based on the input clock.

3) Non-Overlapping Phase Generator: Fig. 7 shows the non-overlapping phase generator circuit, which generates the necessary non-overlapping phases for the 2D-DCT block to perform sampling, accumulation, and resetting operations. The circuit takes the input clock (CLK_{in}) and divides it by 2 to generate the operating clock (CLK). This operating clock is then used to generate various clock phases using 18 delay blocks, IN1–IN17.

Consequently, eight non-overlapping phases (ϕ_0–ϕ_7) are generated to sample the capacitors. The non-overlapping period between the consecutive phases is controlled by the delay introduced in the path of PH1–PH8 before they are fed to the next stage. These delay paths are scan-controlled to adjust the non-overlapping period, compensating for variations due to process differences. The operation of the non-overlapping phase generation block adheres to the design principles

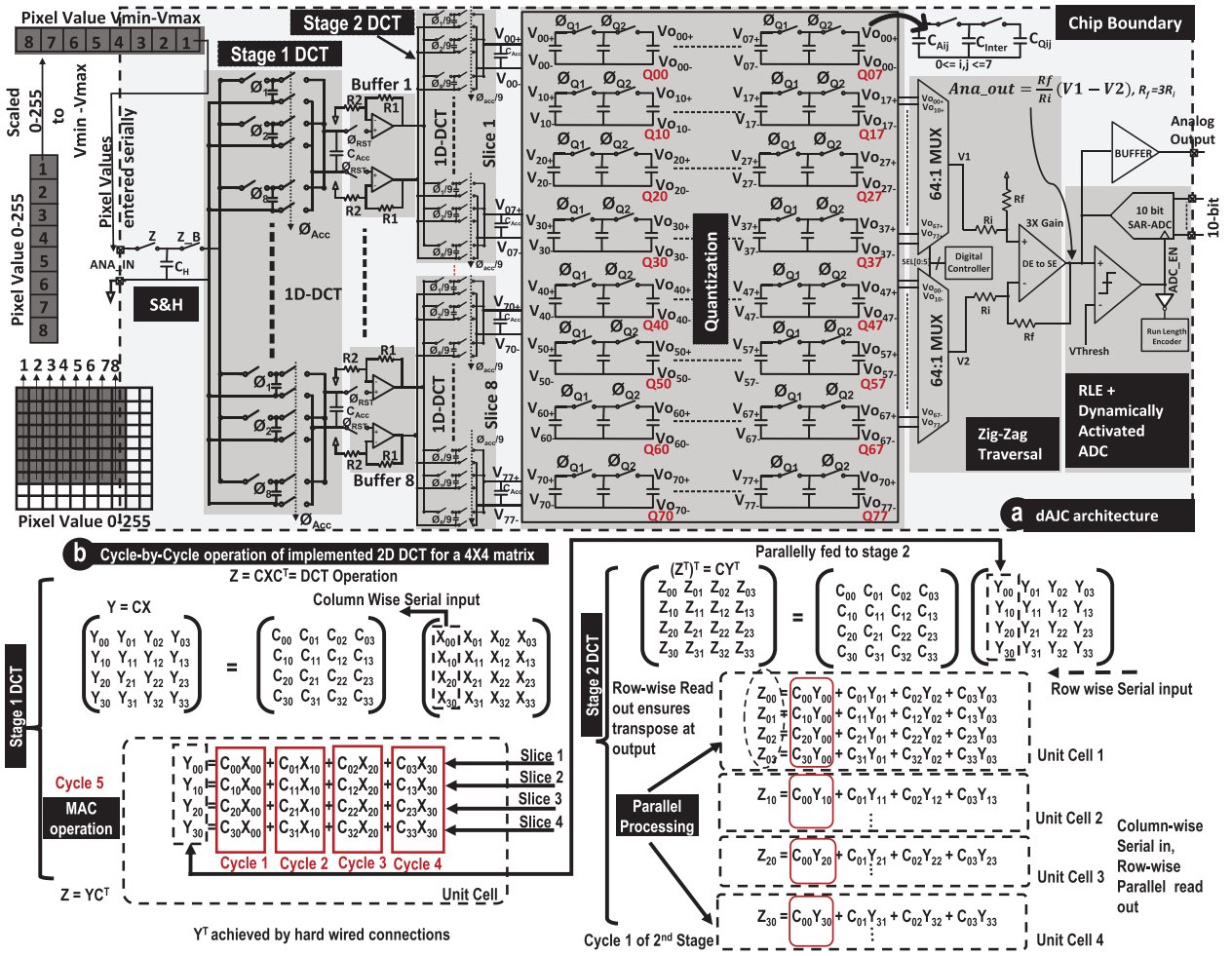


Fig. 4. (a) Overall circuit-level diagram of the 2D-DCT implementation. (b) Cycle-by-cycle operation of the implemented 2D-DCT for a 4 × 4 matrix, which can be extended for an 8 × 8 matrix (implemented in dAJC).

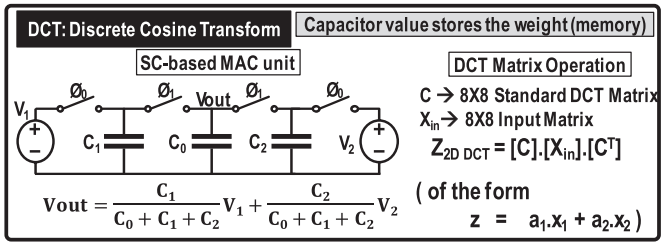


Fig. 5. DCT operation consists of matrix multiplication requiring MAC operations. The schematic shows the MAC operation being implemented by an SC-based circuit.

outlined in [18]. Additionally, the *Z* and *Z'* signals are generated by the non-overlapping phase generator circuit, based on PH2–PH8, to perform sample and hold (S&H) at the input. An initial latency of nine clock cycles of (CLK) is required to ensure that all the phases are available. These clock phases, with the correct timing, are illustrated in Fig. 6 and are used to perform the 2D-DCT operation.

4) Intermediate Buffer: The 2D-DCT architecture performs MAC operations on input image pixels and DCT coefficients using the principle of charge redistribution in SC circuits. As a result, the voltage accumulated on a capacitor may reach the VDD rail, leading to saturation. Additionally, charge sharing may result in the signal being buried in noise, degrading the

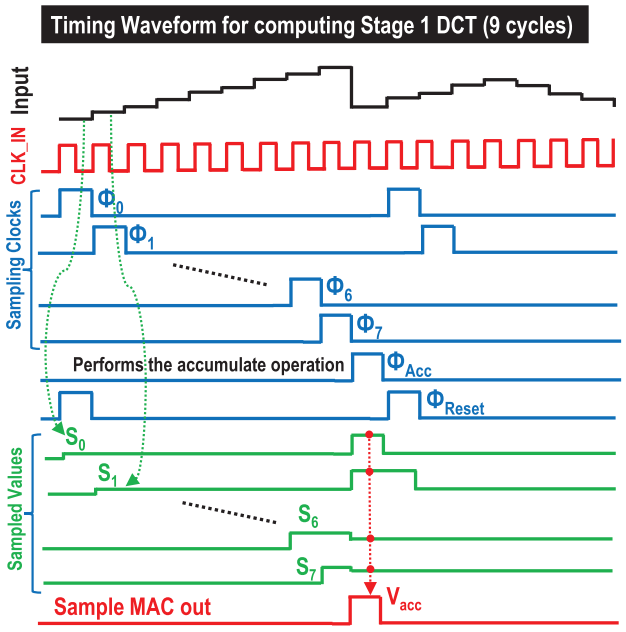


Fig. 6. Timing waveform for 1D-DCT operation.

accuracy. To mitigate these issues, an optimal gain amplifier is essential to prevent both saturation and SNR limitations,

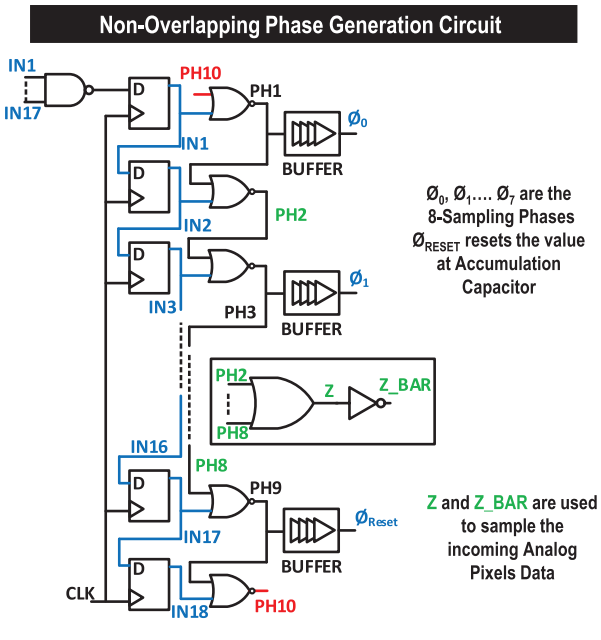


Fig. 7. Non-overlapping phase generation circuit.

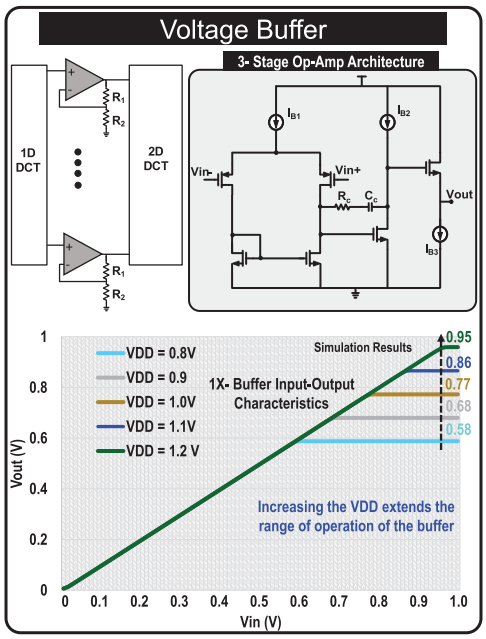


Fig. 8. Op-amp circuit diagram used for 2× gain amplifier. Plots showing the output of the op-amp in unity gain configuration.

ensuring reliable performance and preserving the integrity of the computation.

In the analog camera implementation (detailed discussion in Section III), the image pixels (0–255) are mapped to an input voltage range of $V_{\min}$ to $V_{\max}$ of 0.5 V. From an 8 × 8 standard DCT matrix [19], the first row consists of eight positive coefficients valued at 0.35. Considering that all eight-input pixels are white (0.5 V), the accumulated values after eight MAC operations equal 0.5 × 8 × 0.35 V =1.4 V, which saturates the output. Hence, while performing the *Stage-1 DCT* calculation, an intentional attenuation (μ) of 5 is added such that the maximum voltage level reached equals 0.28 V. Input range reduction to solve the saturation problem is avoided to maximize the input dynamic range. To compensate for this attenuation, an intermediate buffer with a gain of "2" is added in between *Stage-1 DCT* and *Stage-2 DCT* to boost the voltage range to 0.56 V, improving the SNR at the input of *Stage-2 DCT*.

Employing a passive amplifier for voltage gain in an SC circuit, where charge boosting is achieved through a series–parallel combination of capacitors, suffers from leakage and increased area. Implementations using active voltage buffers, such as the flipped voltage follower, folded flipped follower, and common drain [20], suffer from non-linearities. To address these issues, an op-amp in a feedback configuration is used in the current design to improve linearity.

The buffer circuit consists of three stages: the first two stages provide the necessary gain, while the last stage is a voltage follower that drives the high input capacitance (~500 fF) of the 2D-DCT stage. A PMOS-based op-amp has been implemented since the input and the DCT stage output have a low common-mode voltage. Fig. 8 shows the buffer placement between 1D-DCT and 2D-DCT and the detailed circuit diagram of the PMOS-based op-amp. The supply range can be increased for handling a higher input range, as shown in Fig. 8. Since the buffer is configured to provide a gain of 2, resistors R_1 and R_2 are configured in a 1:1 ratio (50 kΩ). Scan control is provided to compensate for process variations, and an interdigitized layout has been used to minimize mismatches.

Notably, this initial implementation employs a continuous-time buffer, which increases power consumption and reduces energy efficiency. Since all computation and processing occur in the discrete domain, switching to a discrete-time buffer in future versions could substantially improve energy efficiency.

B. Quantization

The quantization operation performs element-wise division of the output of the 2D-DCT block by the Q-matrix [21] to selectively reduce precision for less perceptually significant parts of the image, thereby achieving compression. Depending on the amount of compression, the Q-matrix varies. For example, Q_0 provides maximum compression and minimum image quality. However, Q_{90} provides the minimum compression and better image quality. To achieve a tradeoff between the image quality and the degree of compression, the Q_{50} matrix is implemented on the dAJC IC. The output of the quantization is given by the following equation:

$$[Y]_{\text{Quantized}} = \frac{[Y]_{\text{2D-DCT}}}{Q_{\text{matrix}}}. \tag{3}$$

Fig. 9 shows a unit division operation using an SC circuit. To perform division, the computed MAC output (represented as V_{in}) of the 2D-DCT matrix is stored on the accumulation capacitor C_1, and capacitor C_2 is reset to zero. For the division operation, the charge from C_1 is transferred to C_2 so that the resulting voltage (represented as V_{out}) is the required division operation. The capacitor values C_1 and C_2 control the divisor value.

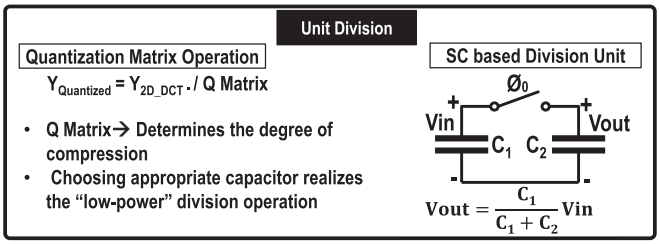


Fig. 9. Quantization operation and division implementation using SC circuits.

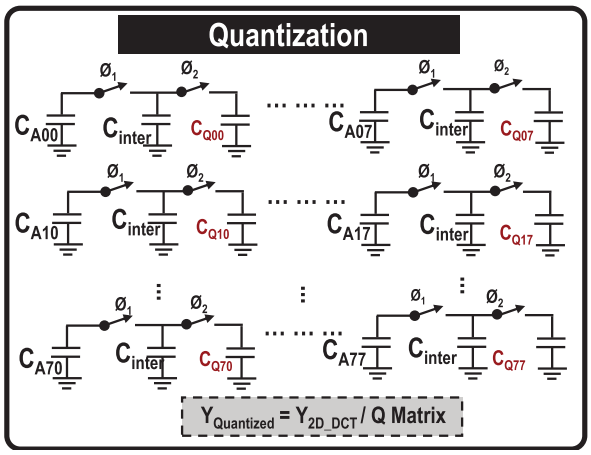


Fig. 10. Circuit-level diagram of the quantization block showing two-step division operation.

The 8 × 8 Q-matrix implemented performs element-wise division in two steps instead of a single-stage division using two non-overlapping phases ϕ_1 and ϕ_2. The sizing of the Q-matrix capacitors and the storage of the divisor values in terms of capacitor value are discussed in Section III. To perform the division operation in phase ϕ_1, the 64-output values on accumulation capacitors C_{Aij} at the output of the 2D-DCT block are connected to the intermediate capacitor C_{inter} to perform the first step of the division, as shown in Fig. 10. In phase ϕ_2, the partially divided value is connected to C_{Qij} to perform the remaining division. Here, $0 \le i \le 7$ and $0 \le j \le 7$. Performing division operations using a two-step process helps reduce the capacitor sizes required and, consequently, reduces the area (details in Section III). However, this leads to the requirement of two phases to perform the division operation.

C. ZZ Traversal

The output of the quantization block consists of significant and insignificant samples in an 8 × 8 matrix form. The quantization output must be reorganized to enable run-length encoding serially. This is achieved by the ZZ traversal block, which rearranges the matrix elements into a 1-D sequence such that the significant samples are at the start of the series, followed by insignificant samples. Out of the 64 samples in the quantized matrix, in most cases, only the top-left samples are the most significant and carry the most information about that particular frame, as shown in Fig. 11. However, this block does not perform the final compression since the output of the block still consists of 64 samples. The traversal is implemented using

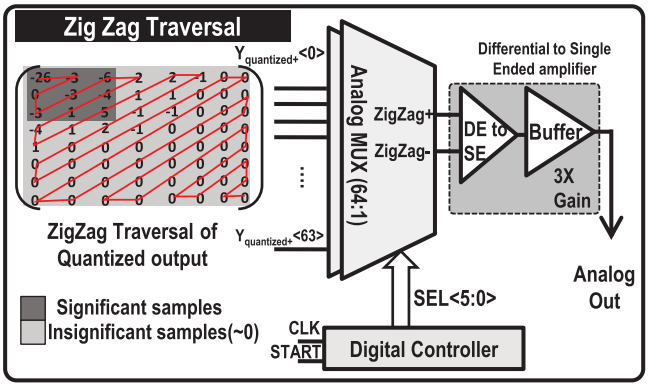


Fig. 11. Circuit-level diagram for the ZZ traversal block.

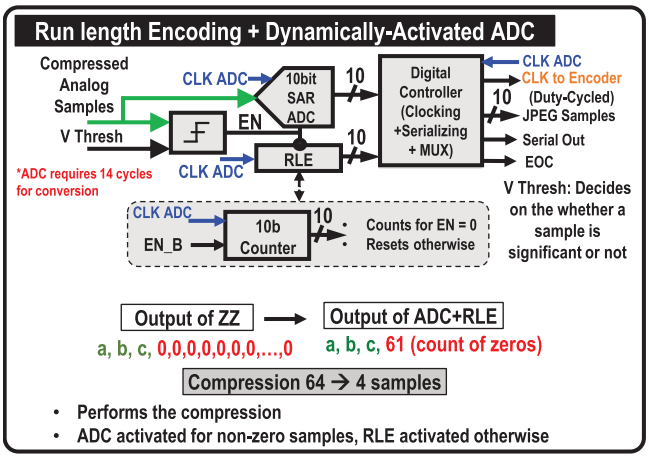


Fig. 12. Circuit-level diagram for the RLE along with the dynamically activated ADC.

a digital controller and a 64:1 analog multiplexer (MUX). Since all the operations up to ZZ traversal are differential to cancel the common-mode switching noise, two sets of such 64:1 analog MUXes are used. The serialized differential output from the ZZ traversal is converted to single ended using a differential-ended-to-single-ended (DE-to-SE) amplifier. A 3× gain buffer follows the DE-to-SE to boost the signal level, which drives the ADC input and the input of the source follower circuit used to observe the compressed analog samples.

D. RLE + Dynamically Activated ADC

The RLE receives the serialized output from the ZZ traversal block, which consists of significant samples at the front, followed by insignificant samples. As shown in Fig. 12, given the 64 serialized samples from the ZZ block, the RLE [22] generates the output containing the significant samples a, b, and c, followed by 61 insignificant samples.

Hence, perform a 16× compression. Depending on the samples from the ZZ traversal block, the dynamically activated ADC is activated only for the significant samples, while the rest of the time, the RLE counts the number of insignificant samples, thereby reducing ADC power consumption by exploiting the sparse nature of the analog samples from the ZZ traversal block. The decision block to turn on the ADC is

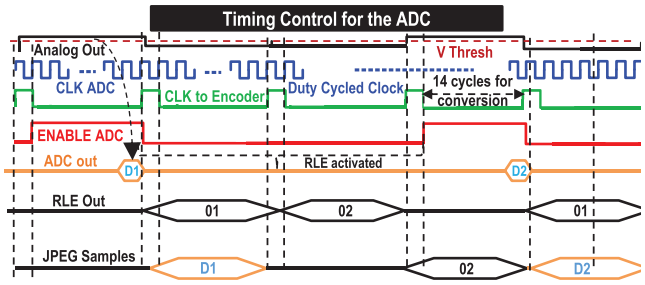


Fig. 13. Timing diagram showing the dynamically activated ADC operation.

realized using a sampler that samples the compressed analog sample output serialized by the ZZ block and compares it with a threshold (V_{thresh}) (indirectly controls the image quality). The implemented sampler is a PMOS-based sampler selected based on the voltage level available at that stage.

1) Dynamically Activated ADC Operation: The implemented ADC depends on the sparsity of the analog input image; as a result, after JPEG compression, depending on whether the sample is significant, such that it affects the visual perception of the image, it is digitized by the ADC; therefore, the ADC working is aware of the input sparsity. This is different from a duty-cycle ADC because the ADC is always turned on and off for a specific duration of the clock period and does not depend on the sparsity of the input. The timing control for the ADC is illustrated in Fig. 13.

When the incoming compressed analog sample exceeds V_{thresh} of the sampler, the ADC is enabled. In cases where analog samples fall below the V_{thresh} threshold, the ADC is deactivated and the RLE block is activated. This block executes run-length encoding and accumulates the count of insignificant samples until the subsequent significant analog sample is detected. Upon arrival of the next significant sample, the RLE outputs the count of insignificant samples, such as "02" in Fig. 13. Overall, the dynamically activated ADC digitizes JPEG samples, followed by the count of insignificant samples. A 10-bit successive approximation register (SAR) ADC operating at 14× higher speed than the input sample rate is used such that it outputs the digitized output after every 14 cycles, once enabled.

2) 10-Bit SAR ADC: A 10-bit SAR-based differential ADC is implemented for digitizing JPEG-compressed samples. The circuit-level diagram of the SAR ADC is shown in Fig. 14. The architecture incorporates a hybrid 10-bit digital-to-analog converter (DAC), which consists of a 5-bit charge-scaling capacitive DAC [CAPDACM, for the most significant bits (MSBs)] and a 5-bit voltage-scaling resistive DAC [for the least significant bits (LSBs)]. The use of a CAPDACM for the MSBs is motivated by the superior matching accuracy of capacitors compared to resistors, thereby improving the overall DAC precision. Furthermore, the resistive DAC ensures monotonic behavior for the LSBs.

The unit capacitance in the CAPDACM is designed to be 9.8 fF. To minimize the differential nonlinearity (DNL), a thermometer decoder is employed to select the appropriate capacitors in the array. The ADC design supports configuration

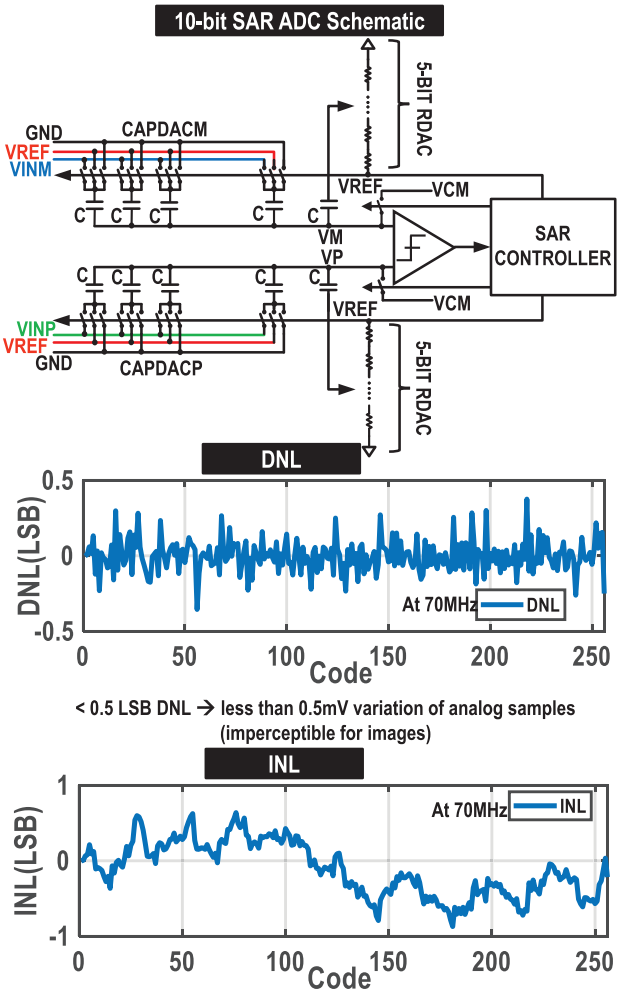
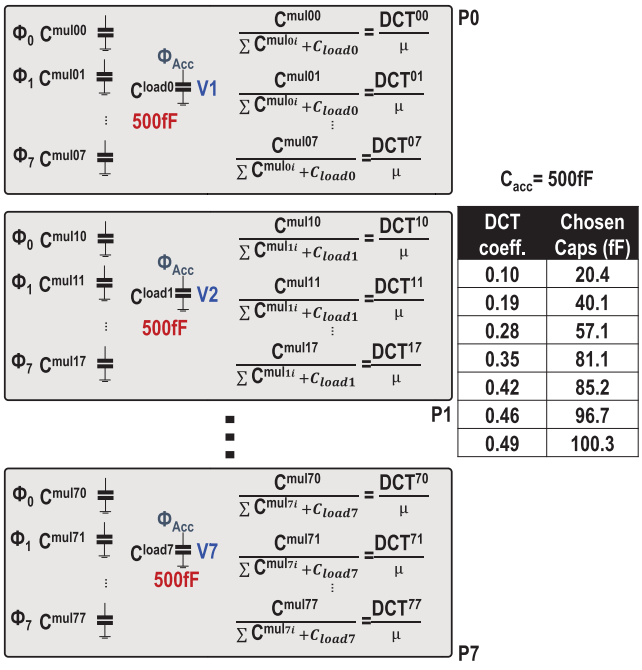


Fig. 14. Circuit-level diagram of the 10-bit SAR ADC with the INL and DNL plots.

as a single-ended ADC by connecting the reference voltage (V_{REF}) and the negative input (V_{INM}) of the CAPDACM to the ground. The overall ADC operation is based on charge redistribution principles, utilizing the top plate of the capacitor array [23].

During the sampling and tracking phase (first three clock cycles), the top plates of the capacitors are connected to the common-mode voltage (V_{CM}), while the bottom plates are connected to the input signal. The charge stored in the capacitor array during this phase is proportional to the input signal. The subsequent ten clock cycles are allocated for the conversion process, starting with the MSB. Conversion begins by connecting the MSB capacitor to (V_{REF}), and the SAR controller iteratively adjusts the DAC output to converge to the input signal. An additional cycle is utilized to flush the digitized data, resulting in the ADC requiring 14 cycles for the A/D conversion process. Hence, the ADC operates at 70 MHz, which is 14× the input sample rate of 5 MS/s.

The performance of the ADC is evaluated in terms of DNL and integral nonlinearity (INL), as shown in Fig. 14. The DNL remains below 0.5 LSB, indicating a variation of less than 0.5 mV in analog samples, which is imperceptible for image processing applications. The INL also remains within 1 LSB,



DCT coeff.	Chosen Caps (fF)
0.10	20.4
0.19	40.1
0.28	57.1
0.35	81.1
0.42	85.2
0.46	96.7
0.49	100.3

Fig. 15. Design considerations for selecting the capacitor values to implement 2D-DCT coefficients.

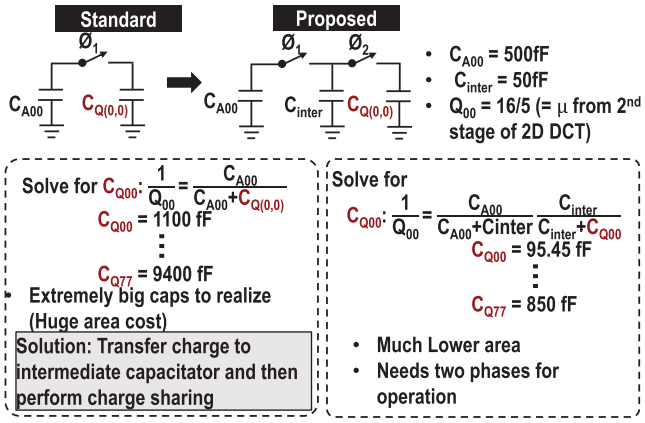


Fig. 16. Design considerations for selecting the quantization capacitor values.

further validating the linearity and accuracy of the ADC at a sampling rate of 70 MHz.

III. Design Considerations

This section outlines the critical design considerations and architectural decisions to achieve an optimized design. The focus is on addressing process variations, minimizing area, and ensuring robust performance. Key aspects such as the choice between voltage- and current-mode architectures, performance metrics for image evaluation, the selection of capacitors for the 2D-DCT matrix and Q-matrix, input image processing, and the calibration and characterization of the Q-matrix are discussed in this section.

A. Choice of Capacitor for DCT Coefficients

The capacitor values are selected based on the DCT coefficient values, as shown in Fig. 15. Each capacitor acts as a memory element to store the corresponding DCT coefficient value. A typical 8 × 8 DCT matrix includes seven unique coefficient values. To avoid saturation during the MAC operation, an intentional attenuation factor ($\mu = 5$) is introduced.

The accumulation capacitor (C_{Acc}) and the sampling capacitors ($C\text{mul}_{ij}$ where $i, j = 0$–7) are sized such that the effective ratios of these capacitors realize the required DCT coefficients. Fig. 15 illustrates this sizing approach. The DCT operation is divided into eight slices (P0–P7), each corresponding to eight coefficients (effectively representing the 8 × 8 DCT matrix). Incoming serial batches of eight samples are sampled onto the $C\text{mul}_{ij}$ capacitors (2D-DCT input capacitors). After eight cycles, these capacitors are connected to C_{Acc} or C_{load} to perform the MAC operation. For instance, to implement the smallest DCT coefficient (0.1), the corresponding capacitor value is 20.4 fF. Based on this value, the sizing of other capacitors is done such that for the largest coefficient (0.49), the capacitor value is 100.3 fF.

The smallest capacitor value is chosen to ensure that the thermal noise (kT/C) remains below 500 μV. Additionally, selecting a smaller capacitor for the smallest coefficient reduces the overall capacitance needed for higher coefficients, thereby minimizing the implementation area. By appropriately sizing the capacitors, the design achieves efficient DCT coefficient realization and area optimization.

B. Choice of Capacitor for Q-Matrix

As discussed in Section II-A, the output of the 2D-DCT is divided by the quantization factor to selectively reduce precision for high- and low-frequency image components. In this implementation, the $Q50$ quantization table is used to balance compression efficiency with image quality. All calculations are based on realizing the $Q50$ matrix, with considerations for compensating for the intentional attenuation factor ($\mu = 5$) added during the DCT computation. The capacitors in this circuit, similar to those used for the DCT coefficients, serve as memory elements to store the quantization table values. The division is performed using an SC-based circuit in a two-step process rather than a single-step division, as shown in Fig. 16. For example, to implement $Q_{00} = 16$, the value is represented as "16/5" to account for μ. In the single-step division method, the required capacitor values are determined by

$$C_{Q00} = \frac{1}{Q_{00}} \cdot \frac{C_{A00}}{C_{A00} + C_{Q(0,0)}} \tag{4}$$

where C_{Q00} corresponds to 16/5 and C_{A00} = 500 fF is the accumulation capacitor from the output of the 2D-DCT. Using this approach, the calculated capacitor size for the smallest quantization value ($Q_{00} = 16$) is 1.1 pF. However, for Q_{77}, a large capacitor of 9.4 pF is required, which is challenging to realize on-chip due to its large area. Also, the other 62 capacitors are in the "pF" range, which are impractical to implement on the chip.

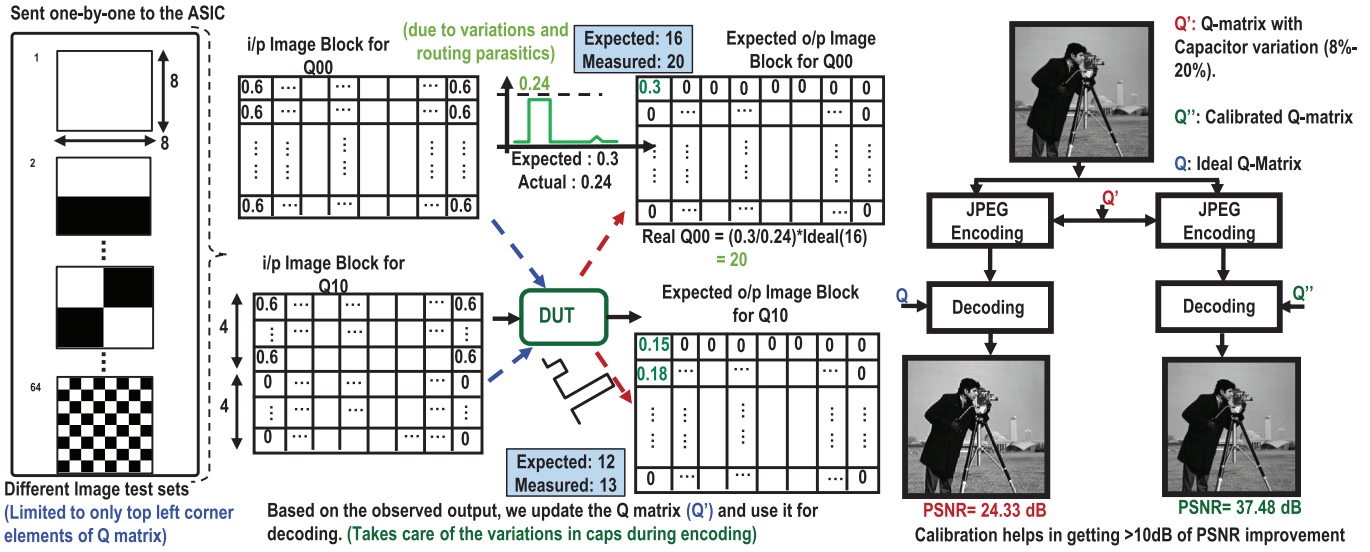


Fig. 17. Calibration process of the inverse-Q-matrix to counter process variation and effect of parasitics (PSNR numbers are from system-level simulations).

To address this issue, a two-step division process is proposed, where the capacitor values are calculated using:

$$C_{Q00} = \frac{1}{Q_{00}} = \frac{C_{A00}}{C_{A00} + C_{\text{inter}}} \cdot \frac{C_{\text{inter}}}{C_{\text{inter}} + C_{Q00}}. \quad (5)$$

Here, $C_{\text{inter}} = 50$ fF is an intermediate capacitor, which helps perform the two-step division, which is sized to reduce (kT/C) noise (<300 μV), prevent leakage discharge, and minimize clock feed through. Using this two-step process, the capacitor size for $Q_{00} = 16$ is reduced to 95.45 fF, and for Q_{77}, the required capacitor size is reduced to 850 fF. This results in a significant area reduction, achieving nearly 10× smaller capacitance values for each quantization value (~ 640 times overall). The tradeoffs of this approach include the need for a two-phase operation and the additional C_{inter} capacitor. However, these overheads are acceptable considering the substantial area savings achieved.

C. Characterization and Calibration for Q-Matrix

The Q-matrix values are stored in capacitance elements; however, these capacitors are prone to process variations and parasitic capacitance effects. Consequently, the on-chip implementation of the Q-matrix (Q') deviates from the ideal Q-matrix (Q). This deviation causes the reconstructed image at the receiver, decoded using the ideal Q-matrix (Q), to have a lower PSNR. To address this issue, the Q-matrix is calibrated to a corrected matrix (Q'') at the receiver, improving the PSNR and mitigating the effects of process variations.

For example, for an input 8 × 8 image of all white pixels (0.6 V), the expected compressed analog samples at (row = 0 and column = 0) is 0.24 V given $Q_{00} = 16$. However, due to process variations, the actual compressed sample is 0.3 V. From this, the effective Q_{00} can be recalculated as (0.3/0.24) ×16 = 20. To calibrate, Q_{00} is replaced with 20 during decoding instead of 16. Similarly, other values in the Q-matrix are calibrated using test images (formed by the combination of black and white pixels), thereby improving the PSNR from 24.33 to 37.48 dB by using the modified inverse Q-matrix (Q''), as shown in Fig. 17. This method demonstrates a significant ~10-dB improvement in PSNR through Q-matrix calibration. Depending on the accuracy needed and when calibration is required, 64 sets of (8×8) test images are sent to calibrate the Q'' matrix at the receiver. Once the optimized matrix is known, it can be used for decoding, automatically canceling the variations in the quantization values. Depending on the number of quantization values, we require 72 calibration cycles for each value. Hence, for calibrating 64 values of the Q-matrix, 64 × 72 cycles are needed. However, it is observed that calibrating the top-left corner of the Q-matrix values (nine values, which perform quantization of the significant samples) improves the PSNR by the highest margin (~10 dB), over a total of 9 × 72 cycles.

D. Input Image Processing

The targeted application operates on the principle that the sensed image pixels are represented in terms of voltage, otherwise known as the analog camera. The camera OV7670 is an analog camera module by Omnivision, which generates the image in the form of an analog voltage signal. First, the image is sensed by the image sensor in the form of an $M \times N$ frame size. Then, the internal hardware serializes the $M \times N$ matrix column-wise in blocks of $1 \times N$, and the hardware operates on a set of 8 × 8 matrices, where each matrix element represents a voltage level corresponding to a particular pixel value. A grayscale image is used as input to emulate the functionality of an analog camera. The grayscale image values are mapped to a voltage scale ranging from V_{min} to V_{max} such that a pixel value of "0" corresponds to V_{min} and a pixel value of "255" corresponds to V_{max}, using MATLAB. The pixel values are then serialized either row-wise or column-wise and transmitted to the test IC using an arbitrary waveform generator (AWG) as a continuous-time signal at a specific data rate. Typically, the pixels are read serially, column-wise, from the imager, which is why a single input port is used, and a serial architecture

is chosen. Compared to a parallel architecture, although the speed is lower, the serial architecture offers the advantages of reduced area and eliminates the need for synchronization between multiple I/O ports.

E. Metrics for Image Performance Evaluation—PSNR

To evaluate the quality of each frame, we use the standard performance evaluation metric, PSNR, to compare the input video frame with the output reconstructed frame (which is obtained by performing inverse quantization and then the inverse DCT (IDCT) operation on the compressed samples). The PSNR is calculated using the following equation:

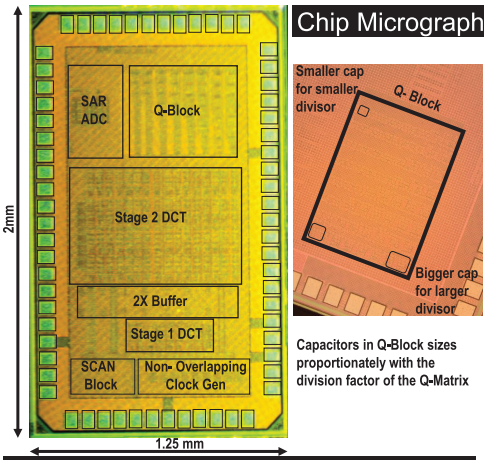
$$\text{PSNR} = 10\log_{10}\left(\frac{R^2}{\text{mse}}\right) \tag{6}$$

where R is the maximum possible fluctuation in the input image and mse represents the mean squared error between the input and reconstructed images. PSNR helps assess the overall quality of the frame, with higher values indicating better-recovered image quality. In the proposed design, image quality is preserved by controlling the threshold voltage (V_{thresh}), which is validated by monitoring the PSNR values.

IV. Measurement Results

A. Chip Micrograph and Test Setup

To validate the dAJC IC's functionality and demonstrate its operation of MJPEG video compression, the dAJC IC is fabricated in TSMC 65-nm LP CMOS technology. Fig. 18 shows the chip micrograph and chip specifications of the fabricated IC. The design occupies a 1.25×2 mm^2 area with an active area of 0.875×1.639 mm^2. The chip micrograph is divided into various critical blocks and highlighted to show the placement of the blocks and the flow of the signals. The hardware implementation of the 64 quantization table values is shown on the IC, where the capacitors are arranged in an 8×8 matrix form. There are separate power supplies for each block, allowing for the measurement of power consumption for each block. The chip specifications table shows the detailed power performance, maximum operating frequency, and the active area of each block. The detailed measurement setup, the various test equipment, and the test printed circuit board (PCB) are shown in Fig. 19. The analog input image samples are provided serially from the AWG, as discussed in Section III-D. The compressed JPEG data are accessible as analog voltage samples and in digital form using the digital output pins JPEG/ADC (0:9). The JPEG-compressed data are processed in MATLAB to reconstruct the image. This process involves multiplying the quantized data by the Q-matrix ("Q"), performing the IDCT, and rearranging the pixels in the reverse order in which they were initially arranged when sent as input to the IC. Finally, the reconstructed image is compared with the original image to calculate the PSNR. The overall performance of the system is measured using the PSNR metric.



Technology Specs and Design Summary

Test Chip (Max. Data Rate = 5MSps Max. Clk. Rate = 20MHz Max. Data Out = 50 Mbps)	Process	65nm 1P9M CMOS LP
	Package	Gold Wire-bond with COB packaging, FR4 PCB
	Active Area	1.734mm^2 (875 um X 1639 um)
	Performance	Sub mW Direct Analog JPEG compression for 500 kSps input analog image
2D DCT (Max. Clk. Rate Stage 1 DCT = 10 MHz Stage 2 DCT = 1.11 MHz)	Supply	0.8V-1V (DCT); 1.2V (2X Buffer)
	Active Area	0.5574mm^2
	Performance	Sub 100uW DCT processor using charge/Voltage mode operation
Quantization (Max. Clk. Rate = 1.11 MHz) + Zig Zag Traversal (Max. Clk. Rate = 5 MHz) + Sampler (Max. Clk. Rate = 10 MHz)	Supply	0.8V-1V
	Active Area	0.2142mm^2
ADC+RLE (Max. Clk. Rate = 120 MHz)	Supply	0.8V-1V
	Active Area	0.0893mm^2
	Performance	10-bit SAR ADC with <100uW at 80 MHz clock (14 cycles for conversion)
Clocking for 2D DCT (Max. Clk. Rate = 20 MHz)	Supply	0.8V-1V
	Active Area	0.0945mm^2

Fig. 18. Chip micrograph with highlighted Q-matrix on-chip and the chip technology specifications.

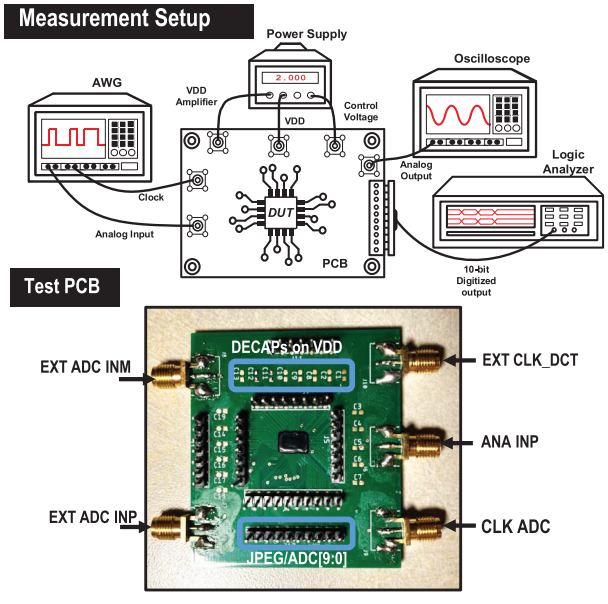


Fig. 19. Measurement setup and the test PCB.

B. Measured Results

Fig. 20 shows the measurement results for the implemented IC. Fig. 20(a) shows the power consumption variation with the input data rate; the total power consumption increases with the increase in the input data rate. For a maximum

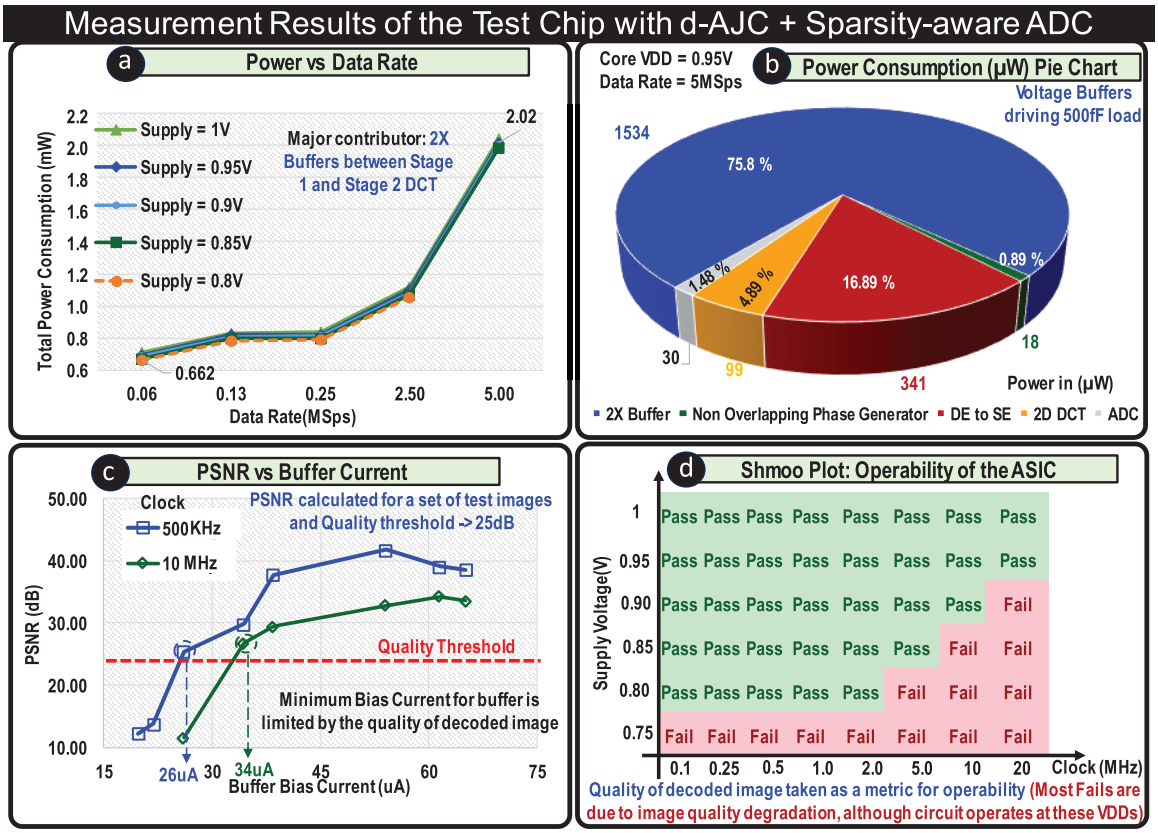


Fig. 20. Overall measurement results. (a) Power versus data rate. (b) Power consumption pie chart. (c) PSNR versus buffer current. (d) Shmoo plot.

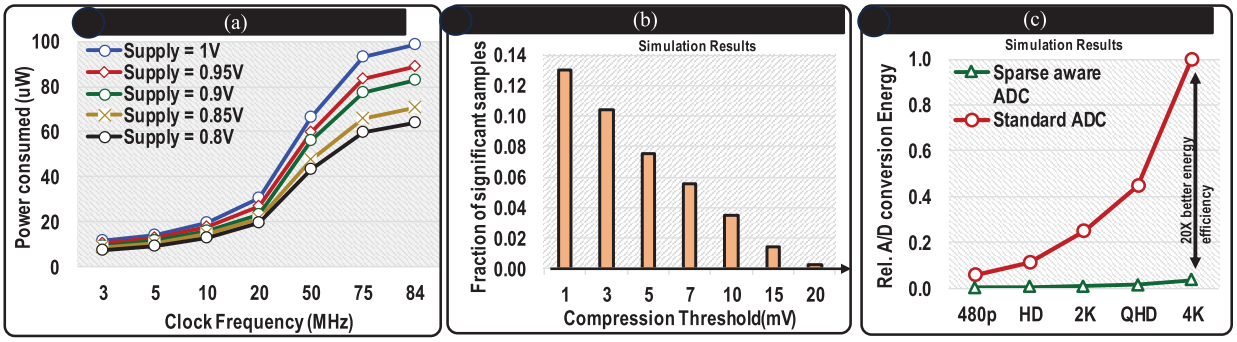


Fig. 21. ADC results. (a) ADC power consumption versus operating frequency for various supply voltages. (b) Simulation result showing the fraction of significant samples versus the compression threshold. (c) A/D conversion energy versus frame size showing 20× improvement in energy efficiency.

input data rate of 5 MS/s, the total power consumption is 2.02 mW. Fig. 20(b) shows the power consumption pie chart highlighting the power consumption of various blocks for a VDD of 0.95 V at a 5-MS/s data rate. It can be observed that the power consumption is predominantly influenced by the op-amp-based voltage buffers, primarily due to their high slew rate requirements, wider bandwidth demands, and the large load capacitance of 500 fF. Fig. 20(c) illustrates the PSNR performance for sampling clock frequencies of 500 kHz and 10 MHz, evaluated using a black-and-white test image to exploit the maximum input range. The PSNR is measured for various unit buffer bias currents, and a quality threshold of approximately 25 dB is established. The threshold helps determine the minimum bias current required for the buffer to ensure accurate compression and faithful reconstruction of the image. It shows that a higher bias current is required to maintain the PSNR above 25 dB for a higher clocking frequency. Therefore, for a clock frequency of 500 kHz and 10 MHz, a minimum bias current of 26 and 34 μA is required, respectively. Finally, the operability area of the ASIC has been tested and measured using the shmoo plot. It shows that the minimum required VDD increases with higher clock frequencies. Therefore, for a clock frequency of 20 MHz, the minimum required supply voltage is 0.95 V; however, for a clock frequency of 2 MHz, the minimum required supply voltage is 0.8 V. However, most failures shown in the shmoo plot are due to image quality degradation, although the circuit operates as required for the tested VDDs.

Fig. 21 shows the ADC measurement results. Fig. 21(a) shows the ADC power consumption versus the ADC clock frequency, which shows an increase in the power consumption with the increase in the clock frequency. For example, at the

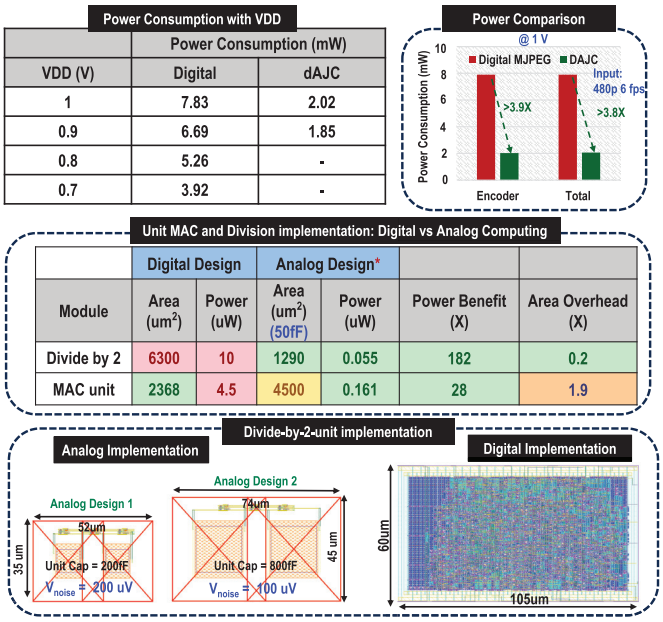


Fig. 22. Comparison of the current implementation with its digital version in terms of power and area.

clock frequency of 84 MHz, the ADC consumes 98.5 μW at a VDD of 1 V. Fig. 21(b) demonstrates that increasing the compression threshold reduces the fraction of significant samples. This reduction is leveraged by the dynamically activated ADC, which digitizes only the significant samples, resulting in substantial energy savings. Finally, in Fig. 21(c), the relative energy efficiency of the dynamically activated ADC is compared to a standard ADC across various video frame sizes varying from 480 pixels to 4K resolution. The dynamically activated ADC achieves up to 20× energy improvement by digitizing only significant samples, which constitute less than 5% of the total samples, whereas the standard ADC digitizes all samples.

V. Comparison With Related Works

In this section, the design is compared against its digital implementation and state-of-the-art implementations.

A. Digital Versus Analog Implementation

To highlight the reduction in power consumption by performing compression in the analog domain instead of digital, a digital MJPEG is simulated in 65-nm technology, implementing all circuit components as shown in Fig. 4 using digital arithmetic. Fig. 22 shows the analog versus digital comparison. In the digital implementation, power consumption scales with the supply voltage (VDD), consuming 7.83 mW at VDD =1 V and 3.92 mW at VDD =0.7 V. Comparing, the analog implementation (dAJC) achieves a power consumption of approximately 2 mW, demonstrating a $> 2\times$ improvement in power efficiency. The observed improvement in power efficiency is primarily attributed to avoiding digitization at the front end. While state-of-the-art ADCs may achieve low conversion energy (~10 pJ/conversion), performing compression on digitized data requires the digital processing block to operate at 10× higher frequency when assuming 10-bit digitization. This increased operating frequency significantly increases the power consumption of the digital implementation. Furthermore, front-end digitization results in a substantial data deluge, leading to increased intermediate data storage and processing overheads, further elaborated in Section VI for various video frame sizes.

Comparing based on PSNR, the baseline digital MJPEG achieves a 33-dB PSNR at 3.92 mW, which is ~5 dB better than our analog implementation (28-dB PSNR at 1.85 mW) but at ~2× higher power. Adding bit truncation (2 bits) lowers the digital implementation PSNR to ~28 dB, matching the analog implementation, but reduces the power by only 40% (~2.4 mW), leaving the analog still 30% more energy efficient. Future efficient analog techniques can further improve this power benefit.

A detailed comparison is also conducted for the fundamental building blocks of the compression unit, namely the divide-by-2 unit and the MAC unit. The analog implementation of the divide-by-2 unit exhibits a 182× power improvement over its digital counterpart, while the MAC unit achieves a 28× power improvement. In terms of area, the analog implementation of the MAC unit offers a 2× reduction in area overhead. In contrast, the divide-by-2 unit requires only half the area of its digital equivalent despite using a large capacitor to suppress kT/C noise. These observations underscore the advantages of analog processing before A/D conversion, which enable significant power savings and eliminate the need to digitize all sensed analog samples (consequently improving A/D conversion energy).

B. Comparison Against State of the Art

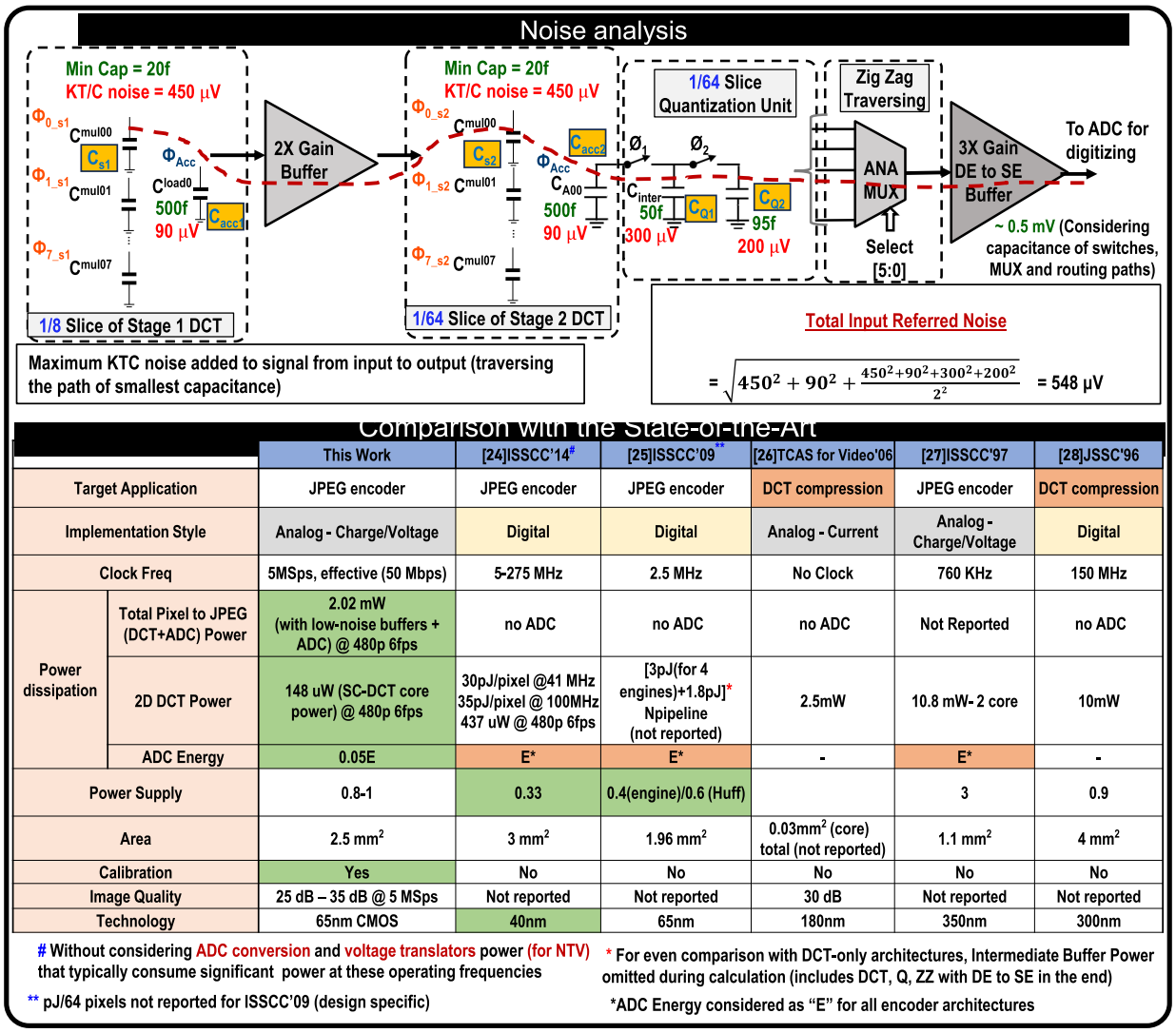
In this section, the dAJC IC is compared against the previous IC implementations [24], [25], [26], [27], [28], as shown in Fig. 23. This work is the first analog implementation to achieve an input data rate of 5 Mb/s. The SC-based 2D-DCT computing core consumes 148-μW power, which is ~12× lower than that of other reported work, which consumes 2.5-mW power [26]. In this work, we provided a calibration technique to counter the process variation compared to other works that rely on design techniques (which are still prone to process variations and parasitics). For total pixel-to-JPEG compression using analog implementation, 2.02-mW power is consumed, which is lower than the 2D-DCT power of other implementations [26], [27], [28]. Considering the front-end ADC energy as "E" for other works performing digital MJPEG compression, our work has a 20× reduction in the ADC energy, as it is enabled for ~5% of the input analog samples (those significant for the video signal integrity).

VI. Discussion

This section delves into potential application areas of this work, focusing on noise analysis for the SC circuit, the impact of dynamically activated ADCs, and potential system considerations for Q-matrix calibration.

1) Noise Analysis and Fundamental Energy:

This section provides a detailed examination of noise accumulation in the design, particularly within the SC circuit.



Comparison with the State-of-the-Art

		This Work	[24]ISSCC'14#	[25]ISSCC'09**	[26]TCAS for Video'06	[27]ISSCC'97	[28]JSSC'96
Target Application		JPEG encoder	JPEG encoder	JPEG encoder	DCT compression	JPEG encoder	DCT compression
Implementation Style		Analog - Charge/Voltage	Digital	Digital	Analog - Current	Analog - Charge/Voltage	Digital
Clock Freq		5MSps, effective (50 Mbps)	5-275 MHz	2.5 MHz	No Clock	760 KHz	150 MHz
Power dissipation	Total Pixel to JPEG (DCT+ADC) Power	2.02 mW (with low-noise buffers + ADC) @ 480p 6fps	no ADC	no ADC	no ADC	Not Reported	no ADC
	2D DCT Power	148 uW (SC-DCT core power) @ 480p 6fps	30pJ/pixel @41 MHz 35pJ/pixel @ 100MHz 437 uW @ 480p 6fps	[3pJ(for 4 engines)+1.8pJ]* Npipeline (not reported)	2.5mW	10.8 mW- 2 core	10mW
	ADC Energy	0.05E	E*	E*	-	E*	-
Power Supply		0.8-1	0.33	0.4(engine)/0.6 (Huff)		3	0.9
Area		2.5 mm²	3 mm²	1.96 mm²	0.03mm² (core) total (not reported)	1.1 mm²	4 mm²
Calibration		Yes	No	No	No	No	No
Image Quality		25 dB – 35 dB @ 5 MSps	Not reported	Not reported	30 dB	Not reported	Not reported
Technology		65nm CMOS	40nm	65nm	180nm	350nm	300nm

Without considering ADC conversion and voltage translators power (for NTV) that typically consume significant power at these operating frequencies

** pJ/64 pixels not reported for ISSCC'09 (design specific)

* For even comparison with DCT-only architectures, Intermediate Buffer Power omitted during calculation (includes DCT, Q, ZZ with DE to SE in the end)

*ADC Energy considered as "E" for all encoder architectures

Fig. 23. Noise analysis for our proposed architecture and comparison with the state-of-the-art implementations.

Fig. 23 (top) provides the noise analysis and its reduction due to the placement of buffers between the sampling capacitors. In this design, we have chosen the minimum capacitance to be 20 fF, as allowed by the foundry, trading off between the occupied area and the effect of parasitics (routing and device capacitance). Lower technology nodes with denser capacitances facilitate smaller capacitors, which can significantly reduce the overall design area.

To analyze the effect of noise from each capacitor [29], [30], the smallest capacitor in each design stage is selected, which is supposed to introduce the maximum noise to the signal of interest. The input-referred noise voltage for the system, encompassing intentional attenuation and the intermediate buffer gain, is shown in Eq. (7) and (8), bottom of the page.

1) *Stage-1 DCT:* The smallest capacitor C_{s1} = 20 fF = C_u introduces thermal noise of 450 μV, while the accumulator capacitor C_{acc1} = 500 fF = 25C_u contributes 90 μV of noise.
2) *Stage-2 DCT:* Similarly, C_{s2} = 20 fF = C_u and C_{acc2} = 500 fF = 25C_u are used, with noise contributions of 450 μV and 90 μV, respectively.
3) *Quantization Stage:* The quantization stage uses C_{q1} = 50 fF = 2C_u and C_{q2} = 95 fF = 5C_u, contributing 300 and 200 μV of noise.

$$V_{\text{noise,total}} = \sqrt{V^2_{\text{noise},s1} + V^2_{\text{noise,acc1}} + \frac{V^2_{\text{noise},s2} + V^2_{\text{noise,acc2}} + V^2_{\text{noise},q1} + V^2_{\text{noise},q2}}{G^2}} \tag{7}$$

$$V_{\text{noise,total}} = \sqrt{\frac{kT}{C_{s1}} + \frac{kT}{C_{\text{acc1}}} + \left(\frac{kT}{C_{s2}}\right)\left(\frac{1}{G}\right)^2 + \left(\frac{kT}{C_{\text{acc1}}}\right)\left(\frac{1}{G}\right)^2 + \left(\frac{kT}{C_{q1}}\right)\left(\frac{1}{G}\right)^2 + \left(\frac{kT}{C_{q2}}\right)\left(\frac{1}{G}\right)^2} \tag{8}$$

The intermediate buffer between the two stages introduces a 2× gain (G), significantly reducing the noise accumulation in later stages. However, note that increasing the buffer gain further could lead to signal saturation, introducing computation errors, as discussed in Section II-A. The total input-referred noise in the system is calculated to be approximately ~ 548 μV [using Eq. (7)].

The energy efficiency of the system is influenced by the power consumption of the intermediate gain buffer, which drives the input capacitance of Stage-2 DCT. To estimate this fundamental energy requirement, we normalized all capacitors to a unit capacitance (C_u). This abstraction simplifies the capacitor sizing across the DCT stages and quantization blocks, which is further elaborated in Section III. The intermediate buffer, designed to drive a capacitive load of approximately 500 fF ($25C_u$), is required to meet a specific slew rate [10% of the clock period (20 MHz)] for the signal transition. This results in a power consumption of approximately 50 μW per buffer.

The total system power consumption includes contributions from the buffer stages (the dominant component, with 22 buffers in total—2 for input S&H, 16 as intermediate buffers, and 4 for differential-to-single-ended conversion), as well as from the clocking and DCT circuitry, which adds extra 500 μW, as calculated from measurement data.

From this analysis, the total power and corresponding energy efficiency for varying values of C_u are shown in Fig. 24. Based on these calculations, the fundamental energy consumption of the system is estimated at 320 pJ/pixel (22 × 50 μW +500 μW)/5 M), representing a 2.5× improvement over the digital implementation (the digital power consumption of 3.92 mW is at 5 MS/s and 0.7 V and the energy efficiency is calculated to be 784 pJ/pixel).

While the continuous-time buffer architecture (current version) increases power consumption, relaxing the slew rate and adopting discrete-time buffers can significantly reduce power, as shown by the dotted lines in Fig. 24 for an ON time of 50% in a future implementation of dAJC.

2) Scope and Impact of the Dynamically Activated ADC:

In this work, we present a dynamically activated ADC that achieves approximately 20× improvement in energy efficiency by tailoring its operation to the sparsity of the input data. For an input sampling rate of 5 MS/s and an ADC conversion energy of 10 pJ/conversion [31], the power consumption of a conventional ADC is 50 μW. In contrast, our implementation achieves a drastically reduced power consumption of 2.5 μW. While the ADC power in this specific scenario is a relatively small fraction of the overall system power (~2 mW), its contribution becomes increasingly significant with higher video resolutions. For example, in 4K video applications, the power consumption of a standard ADC can escalate to the milliwatt range, thereby representing a substantial portion of the total system power budget.

Table I highlights this trend by showing the power consumption of standard and dynamically activated ADCs as video frame sizes increase. It is important to note that our analysis is based on an assumed ADC energy efficiency of 10 pJ/conversion.

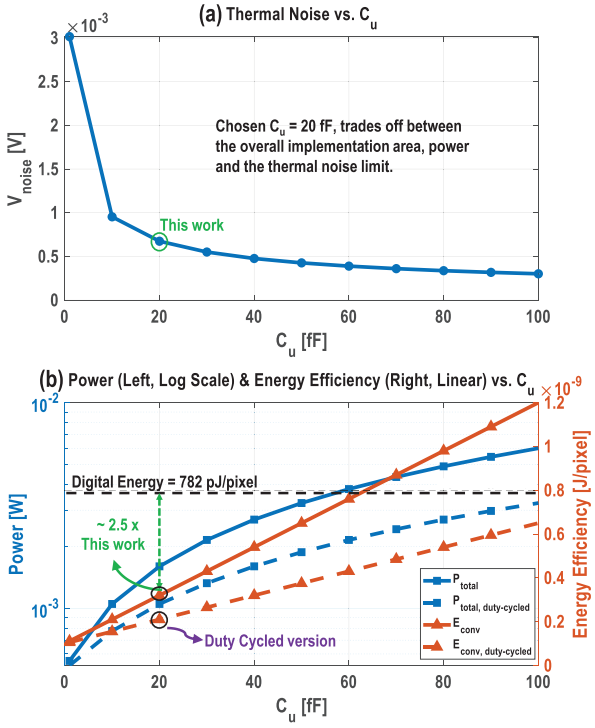


Fig. 24. (a) Analytical thermal noise in the entire system for different values of C_u. We chose C_u = 20 fF for our implementation. (b) Total power consumption and energy efficiency for different values (C_u), comparing continuous-time buffer (solid line) and duty-cycled buffer (dotted line), showing at least a 2.5× improvement over the digital implementation. This shows the fundamental tradeoff of noise, area (C_u), and power.

TABLE I

COMPARISON OF STANDARD VERSUS DYNAMICALLY ACTIVATED ADC POWER AND STORAGE REQUIREMENTS FOR VARIOUS VIDEO RESOLUTIONS

Video	Video Sample Rate	ADC Power		Intermediate Storage
		Standard ADC	dynamically-activated ADC	
480p RGB 6 fps	5 MSps	50 μW	2.5 μW	50 Mb
HD RGB 30 fps	80 MSps	800 μW	40 μW	800 Mb
2K 30 fps	233 MSps	2.33 mW	116 μW	2.3 Gb
4K RGB 30 fps	320 MSps	3.2 mW	160 μW	3.2 Gb

Additionally, digital implementation requires digitizing the input signals through a front-end ADC. For effective operation at 10-bit resolution, the digital system would need to operate at approximately 10× the input data sample rate. For 4K video at 320 MS/s, this corresponds to an operating frequency of 3.2 GHz (serial architecture equivalent to our proposed design), which poses significant challenges at the 65-nm technology node. Alternatively, if digitization and processing were performed at a lower frequency, intermediate storage (≤3.2 GB) would be necessary. This approach, however, incurs additional energy costs associated with memory access and processor fetching operations, further exacerbating overall system power consumption.

3) Future Prospects of the Q-Matrix and Process Calibration:

To address process variations in the capacitors storing the quantization and DCT matrices, the current implementation necessitates calibration by sending a set of 64 8 × 8 test images to the IC. Based on the compressed output data, the Q-matrix (Q'') used for decoding is calibrated. While this

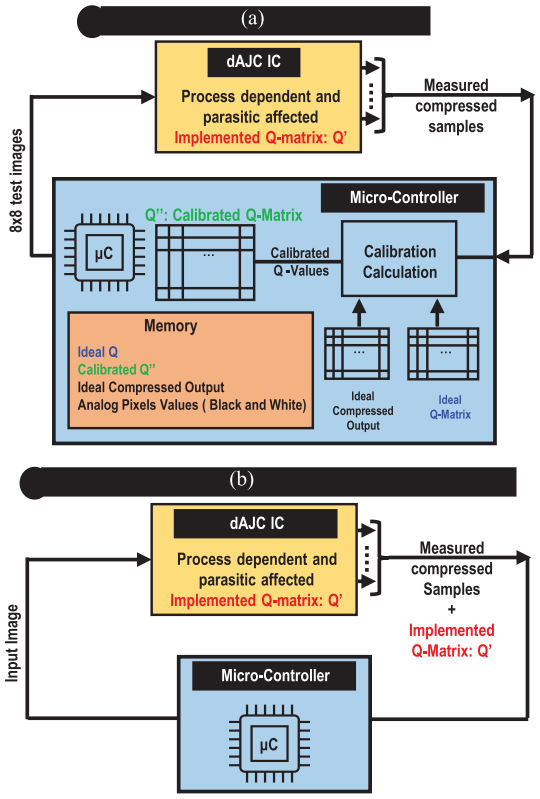


Fig. 25. Comparison of the current implementation with its digital version in terms of power and area. (a) Q-calibration with microcontroller. (b) Future technique by exporting Q'; no calibration required.

calibration process has been implemented in MATLAB, it can be easily automated using a microcontroller-based system as illustrated in Fig. 25(a). In this method, the test/calibration images, the ideal Q-matrix, the ideal compressed output samples, and the calibrated Q-matrix (Q'') are stored in memory. This setup enables automated calibration calculations through a microcontroller, streamlining the process.

For future implementations, a more efficient approach can be adopted, similar to the software-based JPEG standard. In this solution, the Q-matrix used during compression (Q'), which inherently reflects process variations and parasitics, would be exported alongside the compressed analog samples. As depicted in Fig. 25(b), this approach eliminates the need for calibration entirely. The concatenated output containing the compressed data and the implemented Q-matrix (Q') can be directly used during decoding. This ensures that process variations and parasitics are inherently accounted for during the decoding phase, simplifying the workflow, reducing overhead, and aligning with practical deployment scenarios.

4) Potential Application Scenario for dAJC:

We emphasize that analog processing, while not universally superior, can provide significant power savings in certain resource-constrained in-sensor applications where PSNR requirements are modest (e.g., below 40–50 dB) (video references for 25-dB PSNR threshold [32], [33], [34]). This observation aligns with insights from prior work [5]. Analog computing has the potential to enhance the operational lifespan of VSNs where resource constraints often outweigh the need for perfect fidelity, as illustrated by the following examples.

1) *Body Cameras:* With constrained battery life, body cameras can capture video, apply analog compression, and transmit compressed samples to an on-body hub [35] for storage and processing. This significantly conserves energy due to the large disparity between computing and communication energy costs (~10^4 times) [7]. Compressing the information before communicating reduces the power consumption of the sensor node.
2) *Uncrewed Aerial Vehicles (UAVs):* They benefit from lighter and smaller form factors, and onboard analog compression can extend flight durations by reducing computational and data storage demands. Compressed data can be relayed wirelessly to ground systems for detailed analysis.
3) *Surveillance Cameras:* Continuous recording for surveillance applications can leverage analog compression to substantially reduce power consumption and data storage requirements.

These examples highlight the potential of analog-domain compression in scenarios involving wireless transmission, where a minimum PSNR of 25 dB is required, as shown in [36] and [37]. While this approach does not claim universal applicability, it is particularly effective for these applications. For instance, the proposed dAJC, capable of processing 5 MS/s, can compress a single-channel 480-pixel video at 18 frames/s. Applications requiring a resolution of 480 pixels with a PSNR (<40 dB) can effectively utilize the proposed IC.

VII. Conclusion

In conclusion, the first end-to-end implementation of MJPEG compression in the analog domain, utilizing SC circuits, was realized using 65-nm CMOS technology. This work demonstrates a shift of the compression from the digital to the analog domain, significantly reducing digital computation power consumption. The approach achieves a $> 2\times$ improvement in energy efficiency by leveraging analog-domain operations over digital. Additionally, relocating compression to the analog domain enables in-sensor processing, allowing direct signal analysis at the sensor level without relying on data transmission to a centralized processing hub. The implementation demonstrates a 20-fold reduction in ADC conversion energy through input-dependent ADC operation while substantially lowering the overall power consumption associated with communication. Additionally, a system-level algorithm analysis was conducted to develop a calibration scheme tailored to the inherent properties of the MJPEG compression algorithm. By using a set of calibration images, this scheme mitigates the impact of process variations during MJPEG decoding, further enhancing the robustness of the proposed analog design.

References

[1] Y. Kondo, H. Maruno, H. Tominaga, H. Soya, and T. G. Etoh, "Ultra high-speed video camera and its applications," *Proc. SPIE*, vol. 4948, pp. 53–58, Aug. 2003.
[2] K. Lee, J. Lee, Y. Yi, I. Rhee, and S. Chong, "Mobile data offloading: How much can WiFi deliver?," *IEEE/ACM Trans. Netw.*, vol. 21, no. 2, pp. 536–550, Apr. 2013.

[3] B. Chatterjee, N. Cao, A. Raychowdhury, and S. Sen, "Context-aware intelligence in resource-constrained IoT nodes: Opportunities and challenges," *IEEE Design Test.*, vol. 36, no. 2, pp. 7–40, Apr. 2019.

[4] K. G. Kumar, B. Chatterjee, and S. Sen, "CS-Audio: A 16 pJ/b 0.1–15 Mbps compressive sensing IC with DWT sparsifier for audio-AR," *IEEE J. Solid-State Circuits*, vol. 57, no. 7, pp. 2220–2235, Jul. 2022.

[5] R. Sarpeshkar, "Analog versus digital: Extrapolating from electronics to neurobiology," *Neural Comput.*, vol. 10, no. 7, pp. 1601–1638, Oct. 1998.

[6] K. Gaurav Kumar, B. Chatterjee, and S. Sen, "A 16 pJ/bit 0.1–15Mbps compressive sensing IC with on-chip DWT sparsifier for audio signals," in *Proc. IEEE Custom Integr. Circuits Conf. (CICC)*, Apr. 2021, pp. 1–2.

[7] B. Chatterjee et al., "Context-aware collaborative intelligence with spatio-temporal in-sensor-analytics for efficient communication in a large-area IoT testbed," *IEEE Internet Things J.*, vol. 8, no. 8, pp. 6800–6814, Apr. 2021.

[8] S. Okada, Y. Matsuda, T. Watanabe, and K. Kondo, "A single chip motion JPEG codec LSI," *IEEE Trans. Consum. Electron.*, vol. 43, no. 3, pp. 418–422, Mar. 1997.

[9] S. Sen, S. Maity, and D. Das, "The body is the network: To safeguard sensitive data, turn flesh and tissue into a secure wireless channel," *IEEE Spectr.*, vol. 57, no. 12, pp. 44–49, Dec. 2020.

[10] M. Verhelst and A. Bahai, "Where analog meets digital: Analog?to?Information conversion and beyond," *IEEE Solid State Circuits Mag.*, vol. 7, no. 3, pp. 67–80, Summer 2015.

[11] A. Kipnis, Y. C. Eldar, and A. J. Goldsmith, "Analog-to-digital compression: A new paradigm for converting signals to bits," *IEEE Signal Process. Mag.*, vol. 35, no. 3, pp. 16–39, May 2018.

[12] K. G. Kumar, G. Barik, B. Chatterjee, S. Bose, S. Maity, and S. Sen, "A 65 nm 2.02 mW 50 Mbps direct analog to MJPEG converter for video sensor nodes using low-noise switched capacitor MAC-quantizer with automatic calibration and sparsity-aware ADC," in *Proc. IEEE Custom Integr. Circuits Conf. (CICC)*, Apr. 2023, pp. 1–2.

[13] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560–576, Jul. 2003.

[14] G. K. Wallace, "The JPEG still picture compression standard," *IEEE Trans. Consum. Electron.*, vol. 38, no. 1, pp. 18–34, Jan. 1992.

[15] S. A. Khayam, "The discrete cosine transform (DCT): Theory and application," *Michigan State Univ.*, vol. 114, no. 1, p. 31, 2003.

[16] E. Y. Lam and J. W. Goodman, "A mathematical analysis of the DCT coefficient distributions for images," *IEEE Trans. Image Process.*, vol. 9, no. 10, pp. 1661–1666, Oct. 2000.

[17] S. Saha, "Image compression—From DCT to wavelets: A review," *XRDS, Crossroads, ACM Mag. Students*, vol. 6, no. 3, pp. 12–21, 2000.

[18] A. K. Mal and R. Todani, "Non overlapping clock (NoC) generator for low frequency switched capacitor circuits," in *Proc. IEEE Technol. Students' Symp.*, Jan. 2011, pp. 226–231.

[19] N. Ahmed, T. Natarajan, and K. R. Rao, "Discrete cosine transform," *IEEE Trans. Comput.*, vol. C-23, no. 1, pp. 90–93, Jan. 1974.

[20] R. G. Carvajal et al., "The flipped voltage follower: A useful cell for low-voltage low-power circuit design," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 52, no. 7, pp. 1276–1291, Jul. 2005.

[21] A. B. Watson, "DCT quantization matrices visually optimized for individual images," *Proc. SPIE*, vol. 1913, pp. 202–216, Sep. 1993.

[22] D. van Voorhis, "An extended run-length encoder and decoder for compression of black/white images," *IEEE Trans. Inf. Theory*, vol. IT-22, no. 2, pp. 190–199, Mar. 1976.

[23] J. L. McCreary and P. R. Gray, "All-MOS charge redistribution analog-to-digital conversion techniques. I," *IEEE J. Solid-State Circuits*, vol. JSSC-10, no. 6, pp. 371–379, Dec. 1975.

[24] N. Reynders and W. Dehaene, "27.3 a 210 mV 5MHz variation-resilient near-threshold JPEG encoder in 40nm CMOS," in *Proc. IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers (ISSCC)*, Feb. 2014, pp. 456–457.

[25] Y. Pu, J. P. de Gyvez, H. Corporaal, and Y. Ha, "An ultra-low-energy/frame multi-standard JPEG co-processor in 65nm CMOS with sub/near-threshold power supply," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, San Francisco, CA, USA, Feb. 2009, pp. 146–147.

[26] M. Pankaala, K. Virtanen, and A. Paasio, "An analog 2-D DCT processor," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 10, pp. 1209–1216, Oct. 2006.

[27] S. Kawahito et al., "A compressed digital output CMOS image sensor with analog 2-D DCT processors and ADC/quantizer," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 1997, pp. 184–185.

[28] T. Kuroda et al., "A 0.9-V, 150-MHz, 10-mW, 4 mm^2, 2-D discrete cosine transform core processor with variable threshold-voltage (VT) scheme," *IEEE J. Solid-State Circuits*, vol. 31, no. 11, pp. 1770–1779, Nov. 1996.

[29] A. Boukhayma and C. Enz, "A new method for kTC noise analysis in periodic passive switched-capacitor networks," in *Proc. IEEE 13th Int. New Circuits Syst. Conf. (NEWCAS)*, Jun. 2015, pp. 1–4.

[30] S. Pavan, "Simplified analysis of total integrated noise in passive switched-capacitor and N-path filters," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2019, pp. 1–5.

[31] B. Murmann. *ADC Performance Survey 1997–2024*. Accessed: Jun. 9, 2025. [Online]. Available: https://github.com/bmurmann/ADC-survey

[32] C. Weng, B. Curless, P. P. Srinivasan, J. T. Barron, and I. Kemelmacher-Shlizerman, "HumanNeRF: Free-viewpoint rendering of moving people from monocular video," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 16210–16220.

[33] D. J. McDuff, E. B. Blackford, and J. R. Estepp, "The impact of video compression on remote cardiac pulse measurement using imaging photoplethysmography," in *Proc. 12th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG)*, May 2017, pp. 63–70.

[34] D. Reddy, A. Veeraraghavan, and R. Chellappa, "P2C2: Programmable pixel compressive camera for high speed imaging," in *Proc. CVPR*, Jun. 2011, pp. 329–336.

[35] G. Barik, B. Chatterjee, G. Kumar, and S. Sen, "A 65nm 21.9pJ/Sa pixel to PWM conversion SoC with time-domain body communication for ULP body-worn video sensor nodes with distributed real-time inference," in *Proc. IEEE Custom Integr. Circuits Conf. (CICC)*, Apr. 2024, pp. 1–2.

[36] N. Thomos, N. V. Boulgouris, and M. G. Strintzis, "Optimized transmission of JPEG2000 streams over wireless channels," *IEEE Trans. Image Process.*, vol. 15, no. 1, pp. 54–67, Jan. 2006.

[37] X. Li and J. Cai, "Robust transmission of JPEG2000 encoded images over packet loss channels," in *Proc. IEEE Multimedia Expo Int. Conf.*, Jul. 2007, pp. 947–950.

Gourab Barik (Graduate Student Member, IEEE) received the bachelor's degree in electronics and telecommunication engineering from Indian Institute of Engineering Science and Technology at Shibpur, Howrah, India, in 2019. He is currently pursuing the Ph.D. degree in electrical and computer engineering with Purdue University, West Lafayette, IN, USA, working with Prof. Shreyas Sen.

Before joining Purdue University, he worked as an Analog Design Engineer at Tagore Technology, Kolkata, India, working on GaN and CMOS-based PMICs. He has interned with the Multiphase Processor Power (MPP) Team, Texas Instruments (TI), Knoxville, TN, USA, during the Summer of 2025, where he worked on designing high frequency switching DC-DC converters. His research interests include mixed-signal integrated circuit (IC) design for low-power and high-speed sensors for biomedical applications.

Mr. Barik has served as a reviewer for multiple reputed conferences, including the IEEE International Conference on VLSI Design (VLSID), IEEE EMBC, IEEE MWSCAS, and IEEE BioCAS.

Gaurav Kumar K. (Graduate Student Member, IEEE) received the bachelor's degree in electronics and telecommunication engineering from Jadavpur University, Kolkata, India, in 2017. He is currently pursuing the Ph.D. degree in electrical and computer engineering with Purdue University, West Lafayette, IN, USA.

Before joining Purdue University, he worked as a Network Experience Manager at Bharti Airtel, Delhi, India. He has interned with the Advanced Research and Development Discovery Team, Kilby Labs, Texas Instruments (TI), Dallas, TX, USA, during the Summer of 2021, where he worked on designing RX front-end circuits for high-speed electrical links. His current research interests include mixed-signal and compute-in-memory integrated circuit (IC) design for energy-efficient machine learning accelerators and low-power sensing systems.

Mr. Kumar has served as a reviewer for leading conferences and journals, including IEEE JOURNAL OF SOLID-STATE CIRCUITS (JSSC), the IEEE International Conference on VLSI Design (VLSID), and the IEEE Custom Integrated Circuits Conference (CICC).

Baibhab Chatterjee (Member, IEEE) received the Ph.D. degree from the Elmore Family School of Electrical Engineering, Purdue University, West Lafayette, IN, USA, in 2022.

His industry experience includes two years as a Digital Design Engineer/a Senior Digital Design Engineer with Intel, Bengaluru, India, and one year as a Research and Development Engineer with Tejas Networks, Bangalore, India. He was with the Quantum Hardware Design Team, IBM T. J. Watson Research Center, Yorktown Heights, NY, USA, from 2020 to 2021, where he worked on ultra-low-power quantum receiver front ends. He is currently an Assistant Professor with the Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL, USA. His research interests include low-power analog, RF, and mixed-signal circuit design for next-generation biomedical and military applications.

Dr. Chatterjee was a recipient of the NSF CAREER Award in 2025, the Eaton Award in Design Excellence in 2023, and multiple best paper/poster awards, including HOST 2018 (third), CICC 2019, HOST 2019, CICC 2021, and the RFIC/IMS 3MT Award (2020—audience choice).

Shovan Maity (Member, IEEE) received the B.E. degree from Jadavpur University, Kolkata, India, in 2012, the M.Tech. degree in electrical engineering from IIT Bombay, Mumbai, India, in 2014, and the Ph.D. degree in electrical engineering from Purdue University, West Lafayette, IN, USA, in 2019.

He worked as an Analog Design Engineer with Intel, Bangalore, India, from 2014 to 2016, and as a Senior Circuit Design Engineer with Qualcomm, San Diego, CA, USA, from 2019 to 2021. He is currently working as a Staff Electrical Engineer at Ixana, West Lafayette. His research interests lie in the area of mixed-signal circuits and systems for the Internet of Things, and biomedical and security applications.

Dr. Maity received the Institute Silver Medal from IIT Bombay in 2014, the Purdue ECE Fellowship from 2018 to 2019, the IEEE HOST Best Student Paper Award in 2017, and the CICC 2019 Best Paper Award.

Sumon Bose (Member, IEEE) received the Bachelor of Engineering degree in electronics and telecommunication engineering from Jadavpur University, Kolkata, India, in 2012, and the Ph.D. degree from Nanyang Technological University, Singapore, in 2021.

After graduation, he worked in several companies such as Cypress Semiconductor, Bangalore, India, xSi Semiconductor, Bangalore, and Texas Instruments, Bangalore, where he worked on PMIC. Currently, he is an Analog Design Engineer at Quasistatics Inc., West Lafayette, IN, USA. His research interests include low-power analog integrated circuit (IC) and machine learning-based hardware design.

Shreyas Sen (Senior Member, IEEE) is currently an Elmore Associate Professor of ECE and BME with Purdue University, West Lafayette, IN, USA. He is also the Director of the Center for Internet of Bodies (C-IoB), Purdue University. He is the inventor of the Electro-Quasistatic Human Body Communication (EQS-HBC) or Body as a Wire technology, for which he was a recipient of the MIT Technology Review top-10 Indian Inventor Worldwide under 35 (MIT TR35 India) Award in 2018 and Georgia Tech 40 Under 40 Award in 2022. To commercialize this invention, he founded Ixana, West Lafayette, and serves as the Chairperson and the CTO and led Ixana to awards such as two times CES Innovation Award 2024, EE Times Silicon 100, and Indiana Startup of the Year Mira Award 2023. His work has been covered by over 250 news releases worldwide and invited appearance on TEDx Indianapolis, NASDAQ live Trade Talks at CES 2023, Indian National Television CNBC TV18 Young Turks Program, NPR Subsidiary Lakeshore Public Radio, and the CyberWire Podcast. He has authored/co-authored three book chapters, has over 200 journal articles and conference papers and has 25 patents granted/pending. His current research interests span mixed-signal circuits/systems and electromagnetics for the Internet of Bodies (IoB) and hardware security.

Dr. Sen was a recipient of the NSF CAREER Award in 2020, the AFOSR Young Investigator Award in 2016, the NSF CISE CRII Award in 2017, the Intel Outstanding Researcher Award in 2020, the Google Faculty Research Award in 2017, the Purdue CoE Early Career Research Award in 2021, the Intel Labs Quality Award in 2012 for industrywide impact on USB-C type, the Intel Ph.D. Fellowship in 2010, the IEEE Microwave Fellowship in 2008, the GSRC Margarida Jacome Best Research Award in 2007, and nine best paper awards, including IEEE CICC 2019 and 2021 and IEEE HOST 2017–2020 for four consecutive years. His work was chosen as one of the top-10 papers in the Hardware Security field (TopPicks 2019). He serves/has served as an Associate Editor for IEEE SOLID-STATE CIRCUITS LETTERS (SSC-L), *Scientific Reports* (Nature), and *Frontiers in Electronics*; an IEEE Design & Test, Executive Committee Member of IEEE Central Indiana Section; and a Technical Program Committee Member of ISSCC, CICC, DAC, CCS, IMS, DATE, ISLPED, ICCAD, ITC, and VLSI Design.