Optimal Selection of SRAM Bit-Cell Size for Power Reduction in Video Compression

Hyun Kim, *Member, IEEE,* Ik Joon Chang, *Member, IEEE,* and Hyuk-Jae Lee, *Member, IEEE*

***Abstract*— In mobile multimedia devices with video compression capability, a large amount of power consumption is incurred by storing video data in static random-access memories (SRAMs). The power consumption by SRAM access is reduced by decreasing the supply voltage but the decreased supply voltage may cause data loss stored in the SRAM. To reduce the probability of data loss, previous research attempts to increase the SRAM cell size, which incurs a large area overhead. To minimize the area penalty by increasing the SRAM cell size, this paper proposes a novel method to adaptively select the cell sizes of SRAMs according to their sensitivities to the quality degradation in video compression while maintaining the total SRAM area. In order to decide the optimal size for embedded SRAMs, optimization based on Lagrange multipliers and Karush–Kuhn–Tucker conditions is proposed. The proposed optimization significantly reduces the number of simulations necessary to determine the optimal combination of SRAM bit-cell sizes. By applying the proposed heterogeneous SRAM cell sizing scheme along with the proposed optimization scheme, the best Bjontegaard delta peak signal-to-noise ratio (BDPSNR) improvement is achieved. Simulation results show that the proposed approach remarkably improves the video quality by up to 3.72 dB in BDPSNR compared with the conventional SRAM with an identical cell size. These results imply that the proposed heterogeneous SRAM allows a reduction of the supply voltage while maintaining the video quality.**

***Index Terms*— Adaptive static random-access memory (SRAM) bit-cell sizing, energy-quality scalable circuits and systems, low-power SRAM, video compression, video quality optimization.**

I. Introduction

THE demand for portable multimedia devices with video compression capability such as smart-phones and video cameras is increasing. Video compression requires high computational complexity and excessive memory accesses, thereby incurring significant power consumption [1]. Thus, power reduction in video compression is indispensable to prolonging battery lifetime. The video compression standard such as H.264/AVC or high efficiency video coding (HEVC) is widely used in multimedia devices due to its high compression efficiency [2]–[4]; consequently, extensive research efforts have been undertaken to reduce the computational complexity and power consumption in video compression [5]–[8]. These previous efforts focus on power consumption in the logic circuits; however, the power consumption by the accesses to embedded static random access memories (SRAMs) accounts for a significant portion of total power consumption [9]. Therefore, power consumption in embedded SRAMs should also be considered in low-power complementary metal-oxide semiconductor (CMOS) design.

In video compression, CMOS design is still dominant because of its price merit and supply voltage scaling is widely used for reducing the power consumption of embedded memory accesses [10]. With supply voltage scaling for CMOS design, the failure probability of SRAM bit-cells significantly increases [11], [12]. This increase of failure probability in embedded SRAMs results in a significant degradation of video quality in video compression [13]. To avoid such degradation, an error correction code (ECC) is utilized [14]. In [15], a voltage-scaled SRAM for error-tolerant applications with dynamic energy-quality management that uses two resilient techniques, write assist and ECC, is proposed. An ECC compensates effectively for the increase in failure probability of SRAM cells. However, this is accompanied by an increase in area overhead for the addition of a logic circuit for the ECC. In [16]–[18], SRAM structures using extra transistors to reduce the failure probabilities in the embedded SRAMs are proposed. Those approaches require large area penalties although read failures are remarkably reduced. To overcome these area penalties, a priority-based 6T/8T hybrid SRAM structure is proposed in [19]. In [20], heterogeneous SRAM cell sizing is proposed to avoid a video quality degradation with a small area penalty. In [21], an application-specific SRAM design that is suitable for applications with highly correlated data such as video and imaging applications is presented. In [22], in order to further avoid the area penalty, a new FinFET-based SRAM design that exploits the asymmetry of cell-level characteristics with respect to data storage is proposed, which improves the power saving while guaranteeing lower SRAM failure with the negligible area overhead. However, despite these advantages of the new FinFET technology, there is still a high demand for CMOS-based circuits that do not use the FinFET technology due to their cost competitiveness.

These existing efforts have failed in the derivation of the optimal trade-off between energy and performance. In order to design the optimal energy-quality scalable circuits and

Manuscript received December 14, 2017; revised February 17, 2018 and April 12, 2018; accepted April 14, 2018. Date of publication April 18, 2018; date of current version September 11, 2018. This work was supported by the Samsung Research Funding Center of Samsung Electronics under Project SRFC-IT1602-03. This paper was recommended by Guest Editor V. De. *(Corresponding author: Hyuk-Jae Lee.)*

H. Kim and H.-J. Lee are with the Inter-University Semiconductor Research Center, Department of Electrical and Computer Engineering, Seoul National University, Seoul 08826, South Korea (e-mail: snusbkh0@capp.snu.ac.kr; hyuk_jae_lee@capp.snu.ac.kr).

I. J. Chang is with the Department of Electronics and Radio Engineering, Kyung Hee University, Seoul 17104, South Korea (e-mail: ichang@khu.ac.kr).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JETCAS.2018.2828129

2156-3357 © 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

systems for video compression, this paper proposes a novel methodology to achieve video quality enhancement in most CMOS-based video compression circuits, except for specific structures such as FinFET technology, by adjusting the cell size of the embedded SRAMs in video compression. The proposed scheme selects the optimal bit-cell size by considering the influence on video quality by each embedded SRAM while maintaining the area resources. This allows aggressive scaling of the SRAM supply voltage because the proposed scheme selects a relatively large SRAM bit-cell size for sensitive data in order to avoid the quality degradation by supply voltage scaling. In contrast, a relatively small SRAM bit-cell size is chosen to store relatively insensitive data that do not significantly degrade video quality even though the supply voltage is scaled down significantly. It should be noted that utilizing various SRAM cell sizes is much easier than using various supply voltages when implementing a unitary system-on-chip (SoC) design. The proposed design does not require additional voltage sources for SRAM, which results in low design complexity, and provides the most efficient solution at a given low supply voltage. For fast and accurate selection of the SRAM bit-cell size, formulation and optimization schemes based on Lagrange multipliers [23] and Karush-Kuhn-Tucker (KKT) conditions [24] are proposed. By applying the proposed scheme to the SoC design for video compression, the failure probabilities in the embedded SRAMs for sensitive data are remarkably reduced, and the video quality degradation resulting from supply voltage scaling is considerably mitigated without any area and complexity overheads. In other words, the proposed scheme can achieve appreciable power saving without video quality loss compared to a conventional system that utilizes identical SRAM bit-cell sizing. Simulation results show that the proposed scheme significantly improves the Bjontegaard delta peak signal-to-noise ratio (BDPSNR) [25] by up to 3.72 dB while maintaining the total SRAM area.

The rest of this paper is organized as follows. Section II introduces the background for the SRAM architecture and the SRAM bit-cell failure probability model. Section III presents the proposed approach for adaptive selection of the SRAM bit-cell size depending on the functionality of SRAMs in a video encoder. Furthermore, the formulation and optimization schemes are also explained. In Section IV, simulation results are shown, and finally conclusions are presented in Section V.

II. Background

A. Various Bit-Cell Widths on 6T SRAM Architectures

In this subsection, brief discussions on the structure of the 6T SRAM and various bit-cell widths on the 6T SRAM are presented. A six-transistor cell makes a basic structure of SRAM [26]–[28]. Using extra transistors such as 8T and 10T transistors can improve read stabilities and their accompanying quality improvements; however, it causes large area penalties in peripheral circuits [26]. In the 6T SRAM, most of SRAM failures mainly arise from the random transistor threshold voltage (V_{th}) variations which are caused by the random dopant fluctuation (RDF) [26]. The RDF based V_{th} variation in a simple reverse-quadratic model is expressed as follows:

$$\sigma_{Vth} = \sigma_{Vth0} \cdot \sqrt{(\frac{L_{min}}{L}) \cdot (\frac{W_{min}}{W})} \tag{1}$$

where σ_{Vth0} is the standard deviation of the variations for a minimum sized transistor with a channel length L_{min} and a channel width W_{min}. From the equation above, since the RDF effect is the dominant source of permanent defects in digital circuit designs that can be alleviated by increasing conventional transistor sizing, the SRAM failures can be reduced by using larger bit-cell width (W) or length (L) on the 6T SRAM architecture [20], [27]. Therefore, extensive research efforts have been undertaken to minimize the failure probabilities due to the process variations by considering the cell widths and cell lengths of the six transistors as design parameters [20], [27], [28]. The sizes of the transistors are selected such that the area constraint is not violated. It should be noted that the bit-cell can have different types of layout [26]. In order to meet the specifications for all the bit-cell design metrics in the limited possible area, the widths and lengths of the bit-cell transistors must be chosen optimally. Among the cell widths and lengths, the cell widths are commonly utilized for SRAM bit-cell sizing [20], [27]. When various bit-cell widths are utilized, each bit-cell must have the same height; whereas they are composed with a variety of widths. However, there is a limitation on the maximum size for the bit-cell width, 1.8 times larger than the minimum size bit-cell (1.0× bit-cell size), because the reduction of the SRAM failure probability by increasing the transistor size is almost saturated beyond the 1.8× bit-cell size in this paper. From the minimum size of the SRAM bit-cell, the SRAM bit-cell is extended by minimum steps. The minimum step size is decided by 0.05× because this is determined by the minimum grid size in a given process and the minimum permissible grid size is 5nm in case of the semiconductor technology used in this paper [20]. As a result, one of the 17 different sized bit-cell widths from 1.0× to 1.8× can be chosen variously.

B. SRAM Bit-Cell Failure Model

As mentioned in the introduction, failure probabilities of SRAM bit-cells significantly increase as supply voltage is scaled down [12]. It should be noted that an increase of failures in embedded SRAMs within video compression modules such as H.264/AVC and HEVC can result in significant video quality degradation. Fig. 1 summarizes the failure probabilities of SRAM bit-cells for different supply voltages and bit-cell sizes. The horizontal axis represents the normalized SRAM bit-cell sizes, whereas the vertical axis represents the SRAM failure probabilities. The results in Fig. 1 are estimated by utilizing two previous researches, [20] and [29]. In [20], 100,000 samples of read operations in an embedded SRAM within an H.264/AVC encoder are simulated by the Monte-Carlo method along with local intra-die threshold voltage variations at the "Fast-nMOS and Slow-pMOS" corner which results in the worst SRAM failure probability. In [29], the SRAM failure probability is presented according to various supply voltage from 900 mV to 500 mV. In Fig. 1, it is

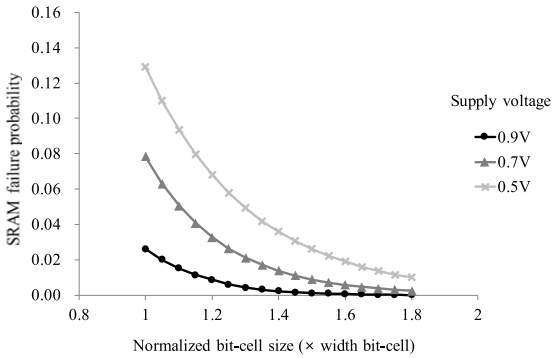


Fig. 1. SRAM failure probabilities for various supply voltages and cell sizes.

assumed that the overall width of all transistors in the 6T SRAM is increased or decreased proportionally. This reduces process variations and consequently improves both read and write stabilities. Considering each transistor separately, it is obvious that more optimized sizing is possible and thus energy efficiency can be further improved, but it is difficult to implement in peripheral circuits because it results in several design challenges for peripheral circuitries. The results represented in Fig. 1 indicate that SRAM failure probabilities gradually increase as supply voltage and bit-cell size decrease. It should be noted that this tendency is based on (1) and thus, it appears similar in other technology libraries. These results mean that an increase in SRAM failures due to supply voltage scaling can be mitigated by using a larger transistor size. The failure probability model represented by Fig. 1 is employed for the analyses reported in this paper.

III. Adaptive Selection of SRAM Bit-Cell Sizes

A. Overall Flow for Adaptive Selection of SRAM Bit-Cell Sizes

As mentioned in Section II, an increase of failure probabilities in embedded SRAMs during video compression results in a significant video quality degradation. However, the degree of video quality degradation varies with the failure locations in the embedded SRAMs. In order to achieve the best video quality within a limited level of power consumption, the SRAM bit-cell size should be adaptively allocated according to the video quality sensitivity of each embedded SRAM.

Fig. 2 shows an overall flow to select the optimal SRAM bit-cell size. At the first step, three simulations with three inputs are performed. First, Monte-Carlo simulations (M.C. SIM) with two inputs, supply voltage and technology model, should be performed to acquire a failure probability model. These simulations require a large number of samples, but there are previous attempts to efficiently reduce the required number of simulations [12]. Furthermore, as this study utilizes the failure probability models from [20] and [29] in Fig. 1, the failure probability models presented previously can be used if the supply voltages and technology models are the same. The second simulation is the hardware post-layout simulation (HW post SIM) to analyze the embedded SRAMs in video compression. As a case study, a hardware-based H.264/AVC encoder [30] is synthesized by using a Synopsys

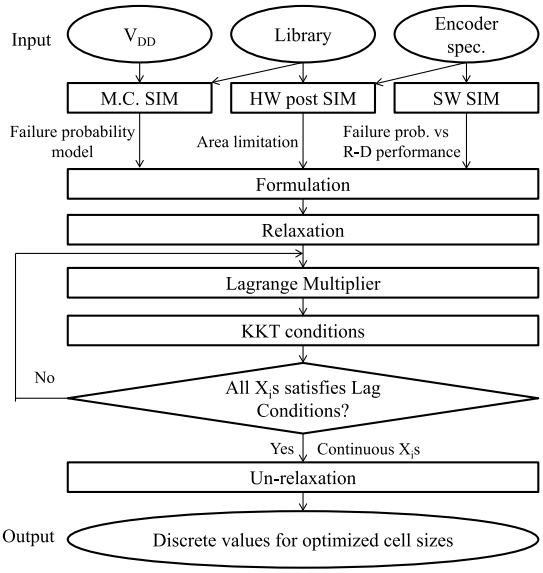


Fig. 2. Overall flow for adaptive selection of an SRAM bit-cell size.

Design Compiler with two inputs, the technology model and the encoding configuration of the H.264/AVC encoder. These data facilitate making the decision on area limitation, which is essential for retaining the total available area. Third, in order to obtain the video quality sensitivity for each embedded SRAM, software simulations (SW SIM) are performed. In this paper, H.264/AVC encoding simulations are performed by using JM 13.2 reference software [31] with the encoding configuration of the hardware-based H.264/AVC encoder. After completing these essential simulations, a mathematical formulation related to optimization is developed to clarify the given problem. Subsequently, a relaxation scheme is applied to each formulation for transforming discrete problems into continuous problems. In order to solve the given optimization problem efficiently, two general mathematical schemes, Lagrange multipliers [23] and KKT conditions [24], are utilized. These optimizations are conducted until all solutions satisfy the Lagrange conditions. The continuous solutions that satisfy the Lagrange conditions are then recovered by an un-relaxation process to derive discrete solutions. Through these steps, the optimal SRAM bit-cell sizes are determined.

In summary, the proposed overall flow can be easily extended to a variety of video compression applications if only three simulations (*i.e.*, Monte-Carlo, hardware post-layout, and software simulations) are performed. Another advantage of the proposed methodology is that it can be combined with a variety of schemes to reduce the SRAM failure probability, such as using less rows per bitline or using a hierarchical bitline [32]. After obtaining the new Fig. 1 with the reduced SRAM error probability by these various schemes, and then applying the proposed overall flow shown in Fig. 2, it is possible to derive the optimal bit-cell sizes for each SRAM block on the given SRAM failure probability.

B. Analysis of an Embedded SRAM in the Video Encoder

1) The Effect of SRAM Bit-Cell Failure on Video Quality Degradation: In this subsection, in order to obtain the video

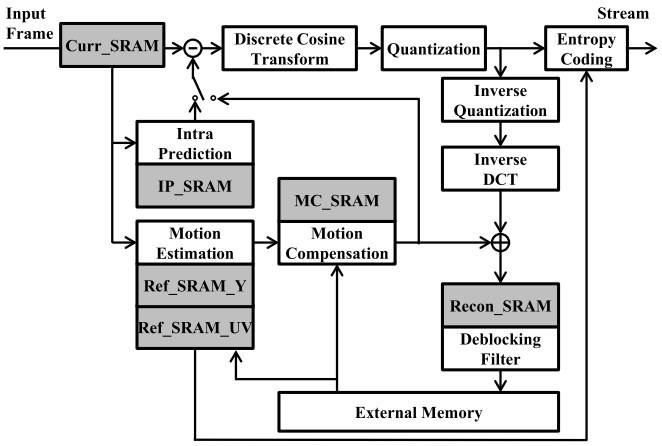


Fig. 3. Basic block diagram of video standard including embedded SRAMs modified from [20] and [30].

TABLE I
ANALYSIS OF THE EMBEDDED SRAMS IN THE VIDEO ENCODER [30]

SRAM	Size (Kbytes)	Area Portion (%)
Ref_SRAM_Y	61.44	46.9
Ref_SRAM_UV	38.4	29.3
Curr_SRAM	14.4	11
Recon_SRAM	7.2	5.5
MC_SRAM	5.76	4.4
IP_SRAM	3.84	2.9

quality sensitivity for each embedded SRAM, the effect of SRAM bit-cell failure positions on video quality degradation is analyzed for an H.264/AVC encoder which is selected for the case study in this paper. The hardware architecture of the H.264/AVC encoder from [30] is shown in Fig. 3. In this figure, the hardware modules and embedded SRAMs are presented with white and gray blocks, respectively. In the video encoder, there are six mandatory SRAMs which are typically used [20], [33], [34], and all of these SRAMs are utilized in this study for determining the optimal SRAM bit-cell size. Table I shows the size of these embedded SRAMs and the corresponding area portion of each of the SRAMs, and a total of 131.04 Kbytes is embedded in the video encoder. In SoC designs, one bit of SRAM can be replaced with about 0.5 gate [35]. Therefore, 131.04 Kbytes of SRAM can be replaced with 524.16 Kgates. Since the gate count of the video encoder used for the case study is 677 Kgates [30], the proportion of the SRAM in the SoC design for video compression is quite remarkable, about 44%.

To evaluate video quality degradation caused by failures of each SRAM within the video encoder, simulation estimates of rate-distortion (R-D) performance are derived according to SRAM failure location. Table II summarizes the conditions of simulations using six HD video sequences with 90 frames in each sequence (listed in top row of Table II). The encoding configurations of the video encoder are presented in the bottom row of Table II. In order to ensure the universality that the proposed method can be applied to a variety of video encoders,

TABLE II
SIMULATION CONDITIONS FOR R-D PERFORMANCE ESTIMATION

Test Video Sequences	HD (1280×720) 90 Frames	Blue_Sky, Factory, Pedestrian_Area Rush_Hour, Sunflower, Tractor
H.264/AVC Encoding Configuration	Profile QP Group of Pictures	Baseline 16,20,24,28,32 IPPP....(only first frame is I)

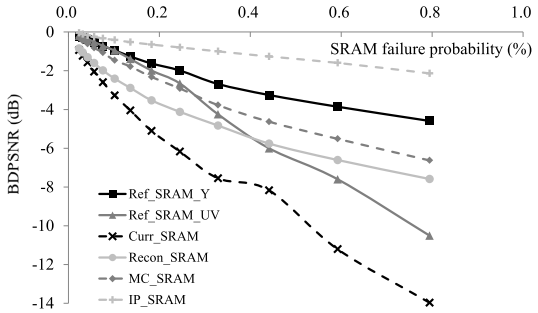


Fig. 4. R-D performance according to failure probabilities in six embedded SRAMs in the video encoder.

simulations are conducted on a baseline profile containing only the basic functions supported by all video encoders.

For each of the six assessed SRAMs, the relationships between SRAM failure probability and video quality degradation are plotted in Fig. 4. The horizontal axis represents the SRAM failure probabilities, whereas the vertical axis represents the BDPSNR with four quantization parameter (QP) values. Among various quality evaluation methods, the BDPSNR is selected because it can express the R-D performance (*i.e.*, both the bit rate and PSNR) by a single variable. This advantage is very helpful when comparing the R-D performance with other factors such as the failure probability. The BDPSNR degradation resulting from bit-cell sizes smaller than 1.2× is too large when operated at a supply voltage below 900 mV; thus, they are not considered hereafter in this paper. Recently, with the rapid development of the semiconductor process technology, it is common to assign iso-area condition bit-cell widths larger than 1.2× to each transistor [20]. As the R-D performance varies according to the location where a SRAM failure occurs, Fig. 4 presents individual relationship graphs for each of the assessed SRAMs. A graph with a steep slope indicates a greater sensitivity to quality loss than that associated with a moderately sloping graph.

In general, during video encoding, inter-frame prediction (*i.e.*, motion estimation) and intra prediction are performed in parallel and one of them showing better results is selected as the best mode. Therefore, "Ref_SRAM_Y" which stores the reference frame of the Y component for inter-frame prediction and "IP_SRAM" for intra prediction are relatively insensitive to quality loss from SRAM failures because failures related to each prediction can be mitigated by the results from the other prediction. On the other hand, the current macroblock SRAM

(denoted by "Curr_SRAM") stores the current macroblock prior to performing inter/intra-block predictions and consequently, failures in the current macroblock are more likely to be sensitive to video quality loss because they affect the entire encoding process. The reference frame SRAMs for the U and V components (denoted by "Ref_SRAM_UV") are also relatively sensitive to video quality loss because the motion vector is only decided by the Y component, and failures in the U and V components cannot be mitigated. SRAM failures in the reconstructed frame SRAM (denoted by "Recon_SRAM") and the SRAM for performing the motion compensation (denoted by "MC_SRAM") will lead to a mismatch problem between the encoder and the decoder. Therefore, in general, they are relatively more sensitive to video quality loss than "Ref_SRAM_Y" and "IP_SRAM." It should be noted that all of the above six SRAMs are mandatory in various H.264 and HEVC architectures [20], [33], [34] and thus, the functionality of the embedded SRAMs in most video encoders is similar. Therefore, even if the number of the reference frame and the supported resolution are changed, the role of each SRAM remains unchanged, and the analysis of the embedded SRAMs in Table I and the correlation between the R-D performance and SRAM failure probabilities in Fig. 4 can always be obtained from all video encoders.

2) Application of Various Bit-Cell Sizes: As mentioned, the key concept of this study is that the degree of the video quality degradation depends on the type of SRAM. Therefore, this paper proposes a novel scheme to use various SRAM bit-cell sizes in video compression according to the type of SRAM. The proposed scheme can be easily applied to most CMOS-based video compression circuits, except for specific structures such as FinFET technology [22], where the number of fins must be changed for width control. Especially, the proposed scheme applies the SRAM sizing to each type of SRAM at block level and as a consequence, it can resolve design problems of the bit level application in peripheral circuits which can occur in the previous researches, [19] and [20]. It should be noted that in SRAM array, peripheral circuits such as sense amplifier and column decoders are drawn with the same size as the width of columns to share these peripheral circuits. Therefore, for different sizing SRAM cells in the bit-level, the layout sizes of peripheral circuits even inside an SRAM array should be suitably adjusted. This is not so practical in terms of design efficiency. Under such environment, per each column transistor sizes of sense amplifiers are not same and hence, their offsets become different. As the transistor sizes become smaller, the offset is inclined to become larger due to process variation [36]. Therefore, previous bit-level different sizing SRAM cells in [19] and [20] have to consider the worst case offset, which further degrades design efficiency, but the proposed scheme in the block-level addresses this problem.

On the other hand, although the proposed scheme also requires additional periphery development and verification effort for various SRAM bit-cell sizes, it is not difficult to design various SRAM bit-cell sizes for the proposed methodology because the SRAM can be configured as a library and designed on the post-layout simulator. Therefore, the proposed method can be easily designed with only 15 libraries from $1.2\times$ bit-cell width to $1.8\times$ bit-cell width. It should be noted that if the bit-cell size is larger than the basic size (*i.e.*, $1.0\times$), the stability of the cell would be improved and thus, it is much easier to design and optimize the peripheral circuit compared to the basic bit-cell size. Therefore, it is possible to make an SRAM library with an increased bit-width without any difficulty. As a result, the layout of the proposed SRAM is neat and straightforward compared to the previous researches, [19] and [20], and the proposed scheme offers simple and efficient SRAM architectures, where the conventional 6T SRAM array structure is not changed.

As described in Section III-B1, six embedded SRAMs are considered for resizing the SRAM bit-cells. In this study, two factors analyzed in Section III-B1, area portion and sensitivity, affect the selection of an appropriate SRAM bit-cell size. In order to minimize the quality degradation, a large SRAM bit-cell size is selected for a sensitive SRAM, whereas a small SRAM bit-cell size is selected for an insensitive SRAM. Thus, SRAM bit-cell size is selected adaptively according to its sensitivity (Fig. 4). However, when a SRAM bit-cell size is selected, the total SRAM area should be maintained, thereby satisfying the following constraint:

$$\sum_{i=1}^{n} A_i \cdot x_i \leq x_{av} \tag{2}$$

where A_i and x_i represent the area portion in Table I and the selected bit-cell size for each SRAM, respectively. Term x_{av} denotes the identical SRAM cell size utilized in a conventional system. This bit-cell size can vary depending on the design environment, but is typically set to a value between $1.3\times$ and $1.6\times$ [20]. Therefore, the selected size of x_{av} differs for each simulation presented in Section IV in order to verify the performance of the proposed scheme over differing environments. Then, each SRAM bit-cell size is decided to the end that a larger bit-cell size can be applied to a more sensitive SRAM while the SRAM area constraint is satisfied. However, although two major factors, area portion and sensitivity, are considered, the best video quality cannot be guaranteed in the adaptive SRAM sizing scheme without an optimization scheme accompanied by logical reasons. To this end, a novel optimization scheme to select the most appropriate SRAM bit-cell size for the best R-D performance is described in Sections III-C and III-D.

C. Formulation

In this section, development of the optimal solution for selecting the most appropriate SRAM bit-cell size for each embedded SRAM is presented. To this end, the relationships among SRAM bit-cell size, SRAM failure probability, and video quality due to the SRAM failure are analyzed and summarized as a single equation.

1) Analysis of the Correlation and Formulation of the Equation: As shown in Fig. 1, SRAM failure probabilities have a direct relation to SRAM bit-cell sizes. The correlation between SRAM bit-cell sizes and SRAM failure probabilities can be expressed by the exponential function shown

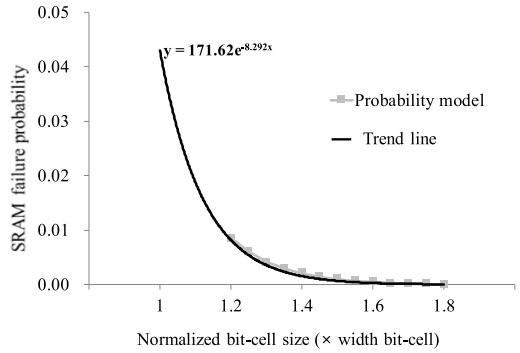


Fig. 5. Exponential function between the normalized SRAM bit-cell sizes and SRAM failure probabilities.

in Fig. 5. The horizontal axis of Fig. 5 presents normalized SRAM bit-cell sizes, whereas the vertical axis presents SRAM failure probabilities on a linear scale. The gray curve labeled "Probability model" is plotted by connecting the points obtained from Monte-Carlo simulations, whereas the black curve labeled "Trend line" denotes an exponential function that is most similar to the "Probability model" curve. The maximum difference between the "Probability model" and the "Trend line" is 0.063% at a 1.3× bit-cell size, a negligible difference. Therefore, the curve that presents the correlation between SRAM bit-cell size and SRAM failure probability is approximately formulated as follows:

$$f_i = a \cdot e^{-b \cdot x_i} \tag{3}$$

where f_i denotes the SRAM failure probability, and a and b are constants determined by Monte-Carlo simulations related to CMOS technology. Although the SRAM failure probability varies according to the SRAM capacity [37], this change in the SRAM failure probability does not affect the proposed optimal selection scheme. This is because the change in the SRAM failure probability caused by the SRAM capacity results in a change in the y-intercept of the exponential graph in Fig. 5 and the variable for the y-intercept is added to the right side of (3), but this variable disappears in the process of calculating the optimal SRAM bit-cell sizes.

In contrast, Fig. 4 shows that BDPSNR degradations have a direct relationship with SRAM failure probabilities. Correlations between SRAM failure probabilities and BDPSNR degradations for each embedded SRAM can be expressed as linear function as shown in Fig. 6. The horizontal axis presents the SRAM failure probabilities and the vertical axis presents the BDPSNR degradations. The gray curves labeled "Measured" denote the simulation results from Fig. 4, whereas the black curves labeled "Trend line" denote the linear functions that are most similar to the "Measured" curves. The maximum difference between the "Measured" and "Trend line" is 10.4% at a 1.35× bit-cell size in the "Curr_SRAM"; this difference can be regarded as tolerable. Therefore, the correlation between SRAM failure probabilities and BDPSNR degradations for each of the assessed embedded SRAMs is approximately formulated as follows:

$$p_i = g_i \cdot f_i + c_i \tag{4}$$

TABLE III
PROBABILITIES OF SRAM FAILURES WITH VARIOUS SRAMS

# of failures	Probabilities
0	$(1\text{-}8\cdot f_{av})^n$
1	${}^nC_1\cdot(8\cdot f_{av})^1\cdot(1\text{-}8\cdot f_{av})^{n-1}$
2	${}^nC_2\cdot(8\cdot f_{av})^2\cdot(1\text{-}8\cdot f_{av})^{n-2}$
⋮	⋮
n-1	${}^nC_{n-1}\cdot(8\cdot f_{av})^{n-1}\cdot(1\text{-}8\cdot f_{av})^1$
n	${}^nC_n\cdot(8\cdot f_{av})^n$

where p_i and g_i denote the BDPSNR degradation and gradient of the linear function, respectively, and c_i is a constant denoting the y-axis intercept. Based on this formulation, g_i can be used to express the sensitivity of each of the embedded SRAMs. In other words, a larger g_i implies that the SRAM is more sensitive to quality loss when SRAM failures occur.

The BDPSNR degradation associated with failures in a single embedded SRAM can be quickly and easily measured by software simulation. However, to obtain the total BDPSNR degradation of video compression consisting of numerous embedded SRAMs is time consuming because the number of all possible combinations of embedded SRAM sizes significantly increases as the number of the embedded SRAMs increases. To save time, the total BDPSNR degradation is not measured by using software simulation. Instead, an estimation model is used to derive the total BDPSNR degradation associated with various combinations of failures in several embedded SRAMs. Although the total BDPSNR degradation can be simply estimated by summation of the individual BDPSNR degradations from each SRAM, this approach may cause considerable errors because overlapping failures from various SRAMs that occur in a single sample (*i.e.*, pixel consisting of eight bits) are not considered. Therefore, the probabilities that various SRAM failures overlap in a single pixel should be included in the estimation model in order to achieve more accurate estimates. Table III shows the probabilities of SRAM failures in video compression according to the number of failures in a single pixel when n SRAMs are considered. The first column presents the number of SRAM failures in a single pixel, while the second column shows the occurrence probabilities of each case. The parameter f_{av} which denotes an identical failure probability on a single bit can be derived from x_{av} by (3). As a single pixel is represented by eight bits, the probability that SRAM failures occur in a single pixel is eight times f_{av}. When SRAM failures occur, all failures, except for a single failure, mean overlapping failures in a single pixel; thus, the probability of overlapping failures denoted by f_{ov} is formulated as follows:

$$f_{ov} = 1 - \frac{{}^nC_1 \cdot (8 \cdot f_{av})^1 \cdot (1 - 8 \cdot f_{av})^{n-1}}{1 - (1 - 8 \cdot f_{av})^n}. \tag{5}$$

In (5), f_{ov} is a constant which can be derived from f_{av}. Therefore, the total BDPSNR degradation in video compression,

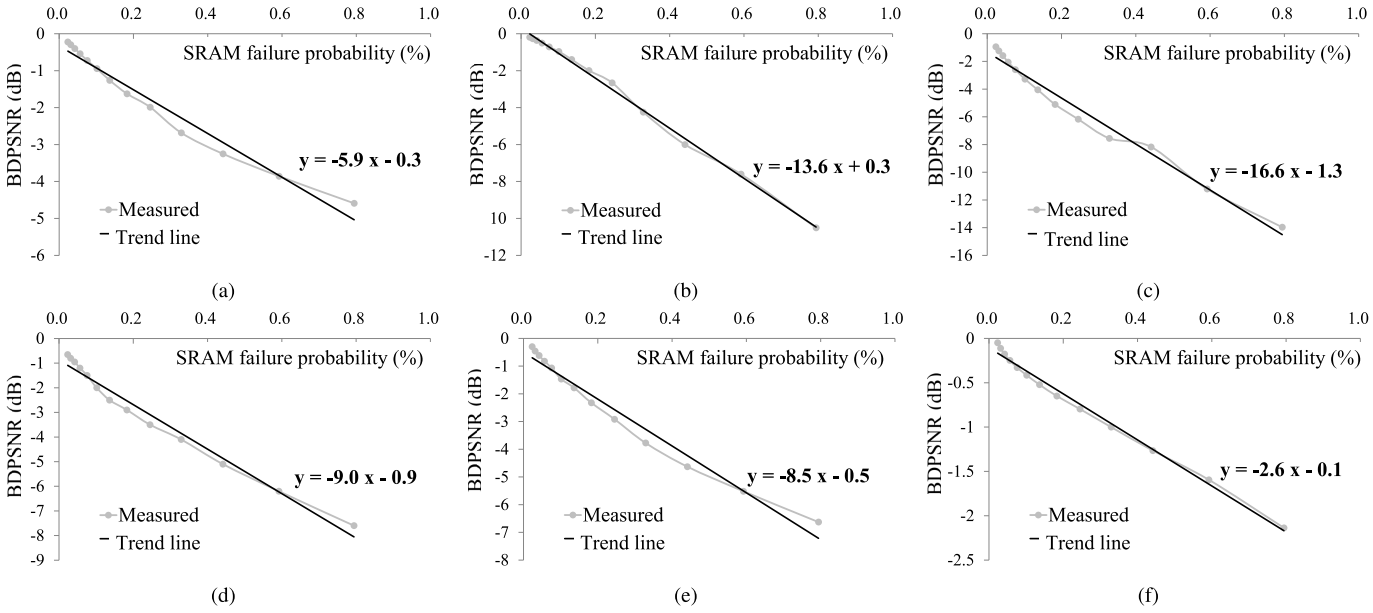


Fig. 6. Linear functions between SRAM failure probabilities and BDPSNRs in each assessed SRAM. (a) Ref_SRAM_Y. (b) Ref_SRAM_UV. (c) Curr_SRAM. (d) Recon_SRAM. (e) MC_SRAM. (f) IP_SRAM.

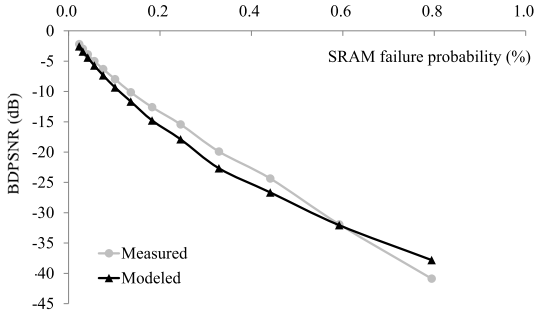


Fig. 7. Comparison of measured and modeled BDPSNR results.

denoted by p_t, can be approximately formulated as follows:

$$p_t = \sum_{i=1}^{n} p_i \cdot (1 - f_{ov}). \tag{6}$$

Fig. 7 compares two graphs; the black graph denotes the estimated total BDPSNR degradation using the proposed model and the gray graph denotes the measured total BDPSNR degradation by software simulation. Comparison indicates that the proposed model estimates are similar to the simulation results. The average difference between the "Measured" and "Modeled" is approximately 8.1%. It should be noted that the trends of two graphs are noticeably similar and the magnitude of the differences between the two graphs is also relatively consistent. These results imply that the proposed model can be reasonably utilized during video compression for roughly estimating the total BDPSNR degradation through assessment of individual BDPSNR degradation values.

2) Video Quality Problem Definition: The problem can be stated as follows: Within a given target supply voltage and an SRAM area constraint, determine each of the SRAM bit-cell sizes to various embedded SRAMs in the video encoder so that video quality is maximized. This problem can be resolved by finding the best combination of SRAM bit-cell sizes for each of the embedded SRAMs that minimizes p_t. By using (3), (4), and (6), this problem can be expressed as follows:

$$\begin{aligned} \min p_t &= \min \sum_{i=1}^{n} p_i \cdot (1 - f_{ov}) \\ &= \min \sum_{i=1}^{n} (1 - f_{ov}) \cdot (a \cdot g_i \cdot e^{-b \cdot x_i} + c_i). \end{aligned} \tag{7}$$

In the last equation in (7), only x_i needs to be decided in order to minimize BDPSNR degradation because a, b, g_i, and c_i are constants that have already been determined by prior simulations. Term x_i has a range that the embedded SRAMs can select, and it has a limitation such that only a discrete value in an interval unit of a 0.05× normalized bit-cell size can be selected. These conditions are expressed as follows:

$$R_{min} \le x_i \le R_{max} \tag{8}$$

where R_{min} and R_{max} denote the minimum and maximum values, respectively, of the available range. It should be noted that (2) should also be considered in order to maintain the total SRAM area when the best combination of SRAM bit-cell sizes is determined.

D. Optimization

1) Lagrange Multipliers: In mathematical optimization, the Lagrange multiplier method, which has a duality characteristic, is widely utilized for determining local maxima and minima of a function subject to constraints [23]. In order to solve (7), there are three constraints derived from (2) and (8), and these are presented as follows:

$$-x_i + R_{min} \le 0. \tag{9}$$

$$x_i - R_{max} \le 0. \tag{10}$$

$$\left(\sum_{i=1}^{n} A_i \cdot x_i\right) - x_{av} \le 0. \tag{11}$$

To utilize Lagrange multipliers, the constraints should be defined in a continuous space; thus, a relaxation process, which is a technique for transforming discrete problems into continuous problems, is applied to (8). As a consequence, (9) and (10) are formulated. These constraints should satisfy the need for primal feasibility and thus, they have inequality. By combining (7), (9), (10), and (11), a weighted sum of objective and constraint functions is formulated with the Lagrange function as follows:

$$L = \sum_{i=1}^{n}(1-f_{ov})\cdot(a\cdot g_i\cdot e^{-b\cdot x_i}+c_i)+\sum_{i=1}^{n}\alpha_i\cdot(-x_i+R_{min}) + \sum_{i=1}^{n}\beta_i\cdot(x_i-R_{max})+\gamma\cdot[(\sum_{i=1}^{n}A_i\cdot x_i)-x_{av}] \quad (12)$$

where α_i, β_i, and γ are Lagrange multipliers associated with (9), (10), and (11), respectively. These Lagrange multipliers are equal to or larger than zero in order to satisfy dual feasibility. The most important point is that (12) is always equal to or less than the objective function (*i.e.*, the first term of the right side) because all of the second, third, and fourth terms of the right side are negative values and thus, it can be expressed as follows:

$$L \le \sum_{i=1}^{n}(1-f_{ov})\cdot(a\cdot g_i\cdot e^{-b\cdot x_i}+c_i). \quad (13)$$

Therefore, the minimal value of the objective function in the right side of (13) is formulated by the maximum value of the Lagrange function in the left side of (13); consequently, those solutions that satisfy the maximum value of the Lagrange function in (12) become the optimal solutions of the given problem.

2) KKT Conditions: In mathematical optimization, KKT conditions are first-order necessary conditions for obtaining an optimal solution in nonlinear programming, provided that some regularity conditions are satisfied [24]. The KKT approach to nonlinear programming is widely combined with the Lagrange multiplier method to meet inequality constraints. Many optimization algorithms can be interpreted as methods for numerically solving the KKT system of equations. In order to satisfy the complementary slackness conditions, all components of the right side in (12), except the first term of the right side in (12), should become zero. Thus, the following equations should be satisfied when the objective function has an optimal solution.

$$x_i = R_{min} \quad or \ \alpha_i = 0. \quad (14)$$

$$x_i = R_{max} \quad or \ \beta_i = 0. \quad (15)$$

$$\sum_{i=1}^{n}A_i\cdot x_i = x_{av} \quad or \ \gamma = 0. \quad (16)$$

From (14) to (16), each equation has two conditions and at least one condition should be satisfied. With primal constraints, dual constraints, and complementary slackness, a gradient of Lagrange function in (12) with respect to x_i should vanish in order to determine the optimal condition for the objective function and it is expressed as follows:

$$\frac{\partial L}{\partial x_i} = -(1-f_{ov})\cdot a\cdot b\cdot g_i\cdot e^{-b\cdot x_i}-\alpha_i+\beta_i+\gamma\cdot A_i = 0. \quad (17)$$

Finally, the optimal solution that satisfies (17) can be solved as follows:

$$x_i = -\frac{1}{b}\cdot\ln\frac{-\alpha_i+\beta_i+\gamma\cdot A_i}{(1-f_{ov})\cdot a\cdot b\cdot g_i}. \quad (18)$$

It should be noted that a larger x_i results in smaller BDPSNR degradation. Therefore, in order to minimize (7), all x_i values cannot be a minimum value, R_{min}, and consequently, α_i should be zero in order to satisfy (14). On the other hand, all x_i values cannot be R_{max} due to the area limitation in (2); thus, β_i should be zero in order to satisfy (15). In (16), the left condition should be satisfied because the available SRAM area is utilized as much as possible to produce a better video quality; subsequently, all x_i values from (18) are inserted to the left equation in (16), and as a consequence, the corresponding γ value is determined because all constants except the γ value are previously determined. With the determined γ value, x_i in (18) can be calculated.

If all solutions satisfy (9) and (10), the set of x_i values denotes the optimal solution for the objective function that satisfies (7). However, if any solution cannot satisfy (9), x_i is adjusted to R_{min} and new α_i value is selected so that x_i and R_{min} have equal values. All x_i values including the updated x_i which is equal to R_{min} are inserted again to the left equation in (16) and the corresponding γ value is updated. This leveraging process with the updated γ value is repeated until all suitable x_i values satisfy (9). On the other hand, if any solution cannot satisfy (10), x_i is adjusted to R_{max} and new β_i value is determined so that x_i and R_{max} are the same. As in the previous case, the leveraging process is repeated with a renewed value from the new x_i, which is updated to R_{max}, until all x_i values satisfy (10).

In summary, the proposed optimization scheme reduces the number of variables required for determining the optimal solutions from N to one, where N is the number of embedded SRAMs considered for SRAM bit-cell sizing. As a consequence, only γ variable leveraging is needed to acquire a set of optimal SRAM cell sizes. The γ variable leveraging is easily performed by using the CVX Stephen Boyd function in MATLAB [38]. When viewed from a broad perspective, compared to the brute-force search algorithm [39] which computes all of the possible bit-cell combinations, the proposed formulation and optimization schemes markedly reduce the number of simulations required to determine the optimized combination of normalized SRAM bit-cell sizes from O($M!$) to O(M) complexity where M is the number of available normalized SRAM bit-cell sizes that can be selected for each SRAM. This scheme is more effective when large numbers of SRAMs (*i.e.*, N) and available normalized SRAM bit-cell sizes (*i.e.*, M) are used.

3) Selection of the Optimized SRAM Cell Size: In this subsection, the values of the case study which are obtained from Monte-Carlo simulations, post-layout simulations by

the Synopsys Design Compiler, and JM reference software simulations are inserted into the proposed scheme in Fig. 2, and actual solutions are calculated. It should be noted that these are limited results for this case study in this paper. The correlation between the R-D performance and SRAM failures depends on the various applications of the video encoder and thus, the optimized result varies according to the video encoder and its application. The proposed scheme in Fig. 2 can support all video encoders since the distribution of the embedded SRAMs in Table I and the correlation between the R-D performance and SRAM failure probabilities in Fig. 4 can be obtained from all video encoders. As mentioned, a and b are constants determined by Monte-Carlo simulations according to the supply voltage and CMOS technology. The R_{min} and R_{max} are also determined based on CMOS technology conditions, whereas g_i is determined by software simulations and A_i is decided based on the hardware design of the video encoder. The identical bit-cell size, x_{av}, is assumed to be a 1.4× iso-area condition. Based on these constants, the solution of the given problem is expressed as follows:

$$\min(5.9 \cdot e^{-8.3 \cdot x_1} + 13.6 \cdot e^{-8.3 \cdot x_2} + 16.6 \cdot e^{-8.3 \cdot x_3} + 9 \cdot e^{-8.3 \cdot x_4} + 8.5 \cdot e^{-8.3 \cdot x_5} + 2.6 \cdot e^{-8.3 \cdot x_6}). \quad (19)$$

In addition, the appropriate constraints are expressed as follows:

$$1.2\times \leq x_i \leq 1.8 \times . \quad (20)$$

$$46.9 \cdot x_1 + 29.3 \cdot x_2 + 11 \cdot x_3 + 5.5 \cdot x_4 + 4.4 \cdot x_5 + 2.9 \cdot x_6 \leq 140. \quad (21)$$

The x_1, x_2, x_3, x_4, x_5, and x_6 terms are the bit-cell sizes allocated to Ref_SRAM_Y, Ref_SRAM_UV, Curr_SRAM, Recon_SRAM, MC_SRAM, and IP_SRAM, respectively. Those sizes are expressed by applying (18) as follows:

$$x_1 = -\frac{1}{8.3} \cdot \ln \frac{46.9 \cdot \gamma}{0.959 \cdot 17162 \cdot 8.3 \cdot 5.9}. \quad (22)$$

$$x_2 = -\frac{1}{8.3} \cdot \ln \frac{29.3 \cdot \gamma}{0.959 \cdot 17162 \cdot 8.3 \cdot 13.6}. \quad (23)$$

$$x_3 = -\frac{1}{8.3} \cdot \ln \frac{11 \cdot \gamma}{0.959 \cdot 17162 \cdot 8.3 \cdot 16.6}. \quad (24)$$

$$x_4 = -\frac{1}{8.3} \cdot \ln \frac{5.5 \cdot \gamma}{0.959 \cdot 17162 \cdot 8.3 \cdot 9}. \quad (25)$$

$$x_5 = -\frac{1}{8.3} \cdot \ln \frac{4.4 \cdot \gamma}{0.959 \cdot 17162 \cdot 8.3 \cdot 8.5}. \quad (26)$$

$$x_6 = -\frac{1}{8.3} \cdot \ln \frac{2.9 \cdot \gamma}{0.959 \cdot 17162 \cdot 8.3 \cdot 2.6}. \quad (27)$$

By changing the inequality in (21) to equality, inserting these six equations into (21) with equality, and then solving this equation, a γ value of approximately 0.409 is obtained, and x_1, x_2, x_3, x_4, x_5, and x_6 are calculated to be 1.283, 1.44, 1.582, 1.592, 1.612, and 1.519, respectively. All of these solutions satisfy (20), and thus, a leveraging process is unnecessary.

To recover the continuous values to the discrete values, the continuous values should be scaled down or up. Up-scaling naturally results in better performance, but if all normalized SRAM bit-cell sizes are scaled up, the total area will exceed the area limitation; thus, the most appropriate discrete values should be selected within the area limitation in (21). First, in order to guarantee remaining within the area limitation, all normalized SRAM bit-cells are scaled down. Then, an area margin is generated and an order of priority for up-scaling should be decided. To decide the order of priority for up-scaling and to find the best discrete set of x_i values, the elements that affect video quality are analyzed. Video quality can be easily measured by determining the BDPSNR change, Δp_i, but the area portion, A_i, also affects video quality because a smaller area portion is better for applying a discrete up-scaling scheme when smaller and larger area portions show the same BDPSNR change; thus, BDPSNR change per area portion should be considered for up-scaling. In addition to these two elements, the difference between the continuous optimal value and the discrete scaled-down value, Δd_i, is also an important factor for up-scaling because the amount of this difference is directly related to the amount of quality degradation. As a result, a new indicator (U_i) to select the appropriate SRAM bit-cell for up-scaling is defined and expressed as follows:

$$U_i = \frac{\Delta p_i \cdot \Delta d_i}{A_i} = \frac{g_i \cdot \Delta f_i \cdot \Delta d_i}{A_i} \quad (28)$$

where Δf_i is the change in failure probabilities when up-scaling is applied. According to this indicator, SRAM bit-cells are scaled up in descending order from high to low indicator level until the area limitation is reached. In this process, if the SRAM bit-cell with a larger U_i cannot be scaled up because it violates the area limitation, the SRAM bit-cell with the next order of U_i can be scaled up while remaining within the area limitation. In this case study, U_1, U_2, U_3, U_4, U_5, and U_6 are calculated as 0.747, 1.133, 0.966, 1.375, 0.44, and 0.528, respectively, by (28).

Finally, Table IV shows both the optimal SRAM bit-cell size and the corresponding SRAM failure probability with 1.4× iso-area condition under a 900 mV supply voltage after applying the up-scaling scheme. The corresponding SRAM failure probabilities are obtained from Fig. 1. Note that although U_1 shows the higher indicator than U_5 and U_6, all SRAMs except Ref_SRAM_Y are scaled up because to apply up-scaling to U_1 violates the area limitation. The results show that, in order to minimize the quality degradation, both the sensitivity and the area portion affect the optimized selection of SRAM cell size. Because the area constraint should be satisfied, to compare the sensitivity within a unit area is very important. It should be noted that the sensitivity is proportional to the quality degradation; whereas the area portion is inversely proportional to the quality degradation. This correlation, CR_i, determines the priority for allocating the area and it can be expressed by an equation as follows:

$$CR_i = \frac{g_i}{A_i}. \quad (29)$$

For application of new indicators, each value for Ref_SRAM_Y, Ref_SRAM_UV, Curr_SRAM, Recon_SRAM, MC_SRAM, and IP_SRAM is calculated as 0.126, 0.464, 1.509, 1.636, 1.932, and 0.897, respectively. The order of these parameters is exactly in concordance with the order of the allocated SRAM bit-cell size in Table IV. It should be noted that the correlation between SRAM bit-cell

TABLE IV
OPTIMAL SRAM BIT-CELL SIZES AND THEIR CORRESPONDING FAILURE PROBABILITIES (WITH THE 1.4× ISO-AREA CONDITION)

SRAM	SRAM Bit-Cell Size	Failure Probability
Ref_SRAM_Y	1.25×	0.6%
Ref_SRAM_UV	1.45×	0.145%
Curr_SRAM	1.6×	0.05%
Recon_SRAM	1.6×	0.05%
MC_SRAM	1.65×	0.031%
IP_SRAM	1.55×	0.07%

sizes and SRAM failure probabilities is expressed by the exponential function in Fig. 5 and thus, the SRAM failure probability changes slowly when the bit-cell size is large enough. Therefore, after allocating sufficient bit-cell size to the SRAM which has the largest CR_i parameter, it achieves better video quality to allocate remaining bit-cell size to the following SRAMs according to the order of the CR_i parameter.

The number of simulations needed to determine the optimal solution when using the proposed scheme is 13 (*i.e.*, ♯ of candidate bit-cell sizes) × 6 (*i.e.*, ♯ of candidate SRAMs) × 5 (*i.e.*, ♯ of QPs) × 6 (*i.e.*, ♯ of test sequences) = 2,340. In contrast, the number of simulations required for determining the optimal solution with the brute-force search algorithm [39] is $13^6 \times 5 \times 6 = 144,804,270$. Therefore, by using the proposed schemes, the number of simulations needed to decide on the optimal solution is significantly reduced by 99.99%. In addition, a new comparison is performed with [20] which optimizes the SRAM bit-cell sizes by commonly known dynamic programming technique. The time complexity of the dynamic programming is known as $O(N^3)$ and thus, the number of simulations is $13^3 \times 5 \times 6 = 65,910$. Therefore, by using the proposed schemes, the number of simulations is remarkably reduced by 97.63% compared to the dynamic programming technique.

IV. SIMULATION RESULTS

This section evaluates the proposed SRAM bit-cell sizing scheme in terms of the video quality and power saving with various supply voltages. The proposed SRAM bit-cell sizing scheme is compared with the conventional SRAM bit-cell sizing scheme that has the same bit-cell size and thereby having the same failure probability. The proposed scheme is also compared with a hybrid SRAM sizing scheme [19], where a minimum sized 8T SRAM is allocated to the upper 3-bits and a minimum size 6T SRAM is allocated to the lower 5-bits. For fair comparison, based on the size ratio of 6T SRAM and 8T SRAM [19], the SRAM structure of [19] is implemented in the SoC design used for this paper while maintaining the same amount of the total SRAM area with the conventional and proposed SRAM schemes.

The simulation results indicate that the overall video quality is significantly improved by the proposed scheme, even under a low-voltage operation. In order to show the video quality

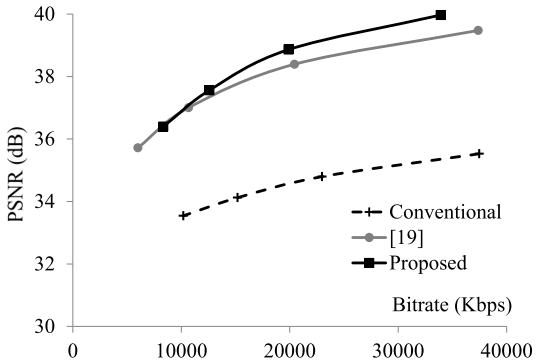
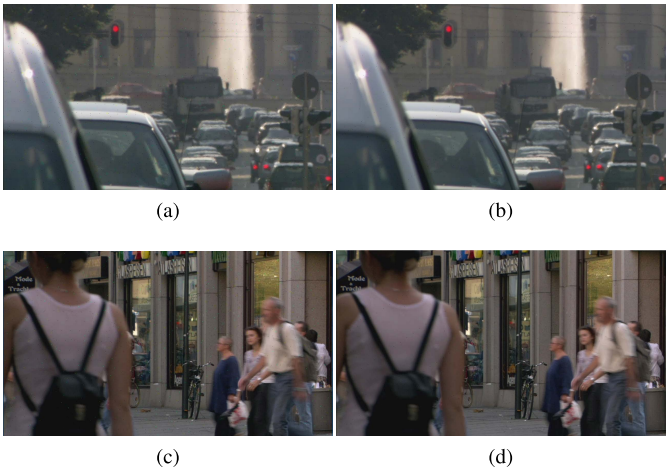


Fig. 8. The R-D curves of the proposed scheme with the 1.4× iso-area condition under 900 mV supply voltage.

changes for various output bitrates during a low-voltage operation, R-D curves are presented in Fig. 8. The horizontal and vertical axes represent the bitrates and PSNRs, respectively. The output bitrates from 5,000 kbps to 40,000 kbps are tested with the six video sequences noted in Table II and the average R-D curves are presented. The dashed curve labeled "Conventional" presents the results from the conventional system with identical SRAM bit-cell sizing. The gray curve labeled "[19]" shows the results obtained by applying the hybrid scheme in [19]. The black curve labeled "Proposed" presents the results obtained by applying the proposed heterogeneous SRAM bit-cell sizing along with the optimization scheme. It should be noted that these three designs use the same amount of SRAM area. The "Proposed" R-D curve shows a significantly higher level of performance than that in the "Conventional" R-D curve, regardless of the bitrate and PSNR level under the 1.4× iso-area condition. The maximum difference in the PSNR between the "Proposed" and "Conventional" curves is approximately 4.78 dB at 30,000 kbps. The "[19]" presents better video quality than the "Conventional," but still worse than the "Proposed." The maximum difference in the PSNR between the proposed scheme and [19] is approximately 0.8 dB at 30,000 kbps.

For subjective comparison, Fig. 9 shows still images from the reconstructed videos and the corresponding PSNRs for the two schemes in order to compare the efficiency of the proposed scheme and that of the conventional scheme. The "Rush hour" and "Pedestrian area" video sequences in Table II are selected for this evaluation. A QP of 20 is used in both the conventional and proposed approaches. The video quality degradation by color dots is clearly visible in the conventional approach, whereas this quality degradation is almost invisible in the proposed approach. Under the 1.6× iso-area conditions of two SRAM structures, the proposed approach shows 2.56 dB and 1.86 dB improvements over that from the conventional approach for the "Rush hour" and "Pedestrian area" videos, respectively.

In Table V, the average BDPSNRs of the proposed scheme and hybrid scheme in [19] are presented under various SRAM iso-area conditions. The results show that the proposed scheme achieves a significant BDPSNR improvement, approximately



(a) (b) (c) (d)

Fig. 9. Still images and PSNR comparisons between the conventional and proposed approaches under the 1.6× iso-area condition. (a) Conventional approach in "Rush hour." (b) Proposed approach in "Rush hour." (c) Conventional approach in "Pedestrian area." (d) Proposed approach in "Pedestrian area."

TABLE V
BDPSNR IN DIFFERENT AREA CONSTRAINTS

Area Constraint	$BDPSNR_{Pro}$ **(dB)**	$BDPSNR_{[19]}$ **(dB)**	**Difference (dB)**
1.3×	3.1	2.8	0.3
1.4×	2.88	2.46	0.42
1.5×	2.5	1.89	0.61
1.6×	2.11	1.2	0.91

TABLE VI
BDPSNR IN VARIOUS LOW-VOLTAGE OPERATIONS

Voltage (mV)	$BDPSNR_{Pro}$ **(dB)**	$BDPSNR_{[19]}$ **(dB)**	**Difference (dB)**
900	2.11	1.2	0.91
700	2.96	2.39	0.57
500	3.72	3.36	0.36

3.1 dB, under the 1.3× iso-area condition. Furthermore, the proposed scheme offers a much better BDPSNR than the hybrid scheme in [19], regardless of the area constraints. As the size of the iso-area conditions increases, the amount of the BDPSNR improvement over the conventional scheme decreases, but the difference in BDPSNR between the proposed scheme and [19] increases. The maximum difference in BDPSNR between the proposed scheme and [19] is approximately 0.91 dB under the 1.6× iso-area condition.

To compare the relationships of the energy consumption, R-D performance, and supply voltage in each scheme is very important. It should be noted that the lower supply voltage results in the greater energy saving, but also more significant BDPSNR drop. In order to compare the difference in BDPSNRs when the supply voltage and energy saving are the same, Table VI shows the average BDPSNRs of the proposed scheme and hybrid scheme in [19] under various low-voltage operations. The results show that the proposed

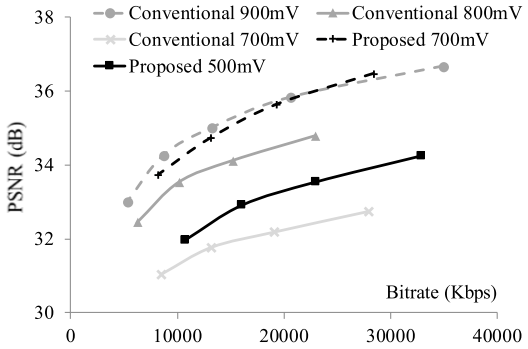


Fig. 10. Comparison of R-D performance curves for supply voltage gain from the proposed scheme with the 1.45× iso-area condition.

approach achieves a positive BDPSNR compared with the conventional design and a notably better BDPSNR than [19], regardless of the supply voltage. Furthermore, the differences between the proposed scheme and other schemes increase as the supply voltage is aggressively scaled down. This means that the proposed scheme can achieve much higher efficiency at very low voltages.

On the other hand, in order to compare the difference in energy savings by the difference in supply voltages when the R-D performance is almost the same, Fig. 10 shows R-D curves obtained via the conventional and proposed schemes at different supply voltages. It should be noted that thanks to the positive BDPSNR gain, the proposed heterogeneous SRAM can operate with a lower supply voltage compared to the conventional SRAM while maintaining similar R-D performances. The results show that the proposed SRAM approach with a 700 mV supply voltage (*i.e.*, black dashed curve) presents approximately similar R-D performances with the conventional SRAM design operating with a 900 mV supply voltage (*i.e.*, gray dashed curve). Furthermore, the proposed SRAM approach with a 500 mV supply voltage (*i.e.*, black curve) presents approximately similar R-D performances with the conventional SRAM design operating with a 750 mV supply voltage because the R-D curve for the proposed approach with a 500 mV supply voltage is placed between the R-D curve for the conventional SRAM with a 700 mV supply voltage (*i.e.*, light gray curve) and that with a 800 mV supply voltage (*i.e.*, dark gray curve). Therefore, the proposed SRAM approach reduces the supply voltage requirement by up to 250 mV without a quality degradation, indicating a power gain of up to 55.56% over that in the conventional SRAM approach.

Not only is there video quality enhancement, but the proposed scheme also has the advantage of allowing implementation of an integrated circuit. As mentioned above, the hybrid SRAM scheme in [19] can be applied to the embedded SRAM with bit level, but that is very difficult to implement in peripheral circuits due to different cell pitches. However, the proposed scheme avoids this problem.

V. CONCLUSION

In video compression, supply voltage scaling is used widely for reducing the power consumption related to the embedded

SRAM accesses. Unfortunately, supply voltage scaling causes SRAM failures, and as a consequence, results in a significant video quality degradation. In order to overcome these problems, this paper proposes a heterogeneous SRAM cell sizing methodology for video compression. This methodology mitigates a video quality degradation by adjusting cell sizes of the embedded SRAMs according to the functionality of these SRAMs without area and complexity overheads. To determine the optimal sizing of each SRAM bit-cell, an optimization scheme based on the application of Lagrange multipliers and KKT conditions is proposed. The number of simulations required to determine the optimal combination of the normalized SRAM bit-cell size is remarkably reduced thanks to the proposed optimization scheme. By applying the proposed heterogeneous embedded SRAM cell sizing scheme along with the proposed optimization scheme, significant BDPSNR improvements by up to 3.72 dB, are achieved compared to those from the conventional SRAM cell sizing. These results show that the proposed SRAM sizing scheme achieves a large amount of power savings in video compression curcuits without a loss of video quality compared to the conventional system. Therefore, the proposed SRAM sizing approach can greatly contribute to the low-power circuit design of video compression in most of the areas excluding the FinFET technology and it has high possibility to be extended to a design structure suitable for FinFETs in the future.

REFERENCES

[1] T.-C. Chen *et al.*, "Analysis and architecture design of an HDTV720p 30 frames/s H.264/AVC encoder," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 6, pp. 673–688, Jun. 2006.

[2] H. Kim, C. E. Rhee, and H.-J. Lee, "A low-power video recording system with multiple operation modes for H.264 and light-weight compression," *IEEE Trans. Multimedia*, vol. 18, no. 4, pp. 603–613, Apr. 2016.

[3] Q. Chen and D. Wu, "Delay-rate-distortion model for real-time video communication," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 8, pp. 1376–1394, Aug. 2015.

[4] Y.-H. Chen, T.-C. Chen, C.-Y. Tsai, S.-F. Tsai, and L.-G. Chen, "Algorithm and architecture design of power-oriented H.264/AVC baseline profile encoder for portable devices," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 8, pp. 1118–1128, Apr. 2009.

[5] H. Kim, C. E. Rhee, and H.-J. Lee, "An effective combination of power scaling for H.264/AVC compression," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 23, no. 11, pp. 2685–2689, Nov. 2015.

[6] C.-E. Rhee, J.-S. Kim, and H.-J. Lee, "Cascaded direction filtering for fast multidirectional inter-prediction in H.264/AVC main and high profile compression," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 3, pp. 403–413, Mar. 2012.

[7] C.-E. Rhee, T. S. Kim, and H.-J. Lee, "An H.264 high-profile intra-prediction with adaptive selection between the parallel and pipelined executions of prediction modes," *IEEE Trans. Multimedia*, vol. 16, no. 4, pp. 947–959, Jun. 2014.

[8] O. Ndili and T. Ogunfunmi, "Algorithm and architecture co-design of hardware-oriented, modified diamond search for fast motion estimation in H.264/AVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 9, pp. 1214–1227, Sep. 2011.

[9] C.-P. Lin *et al.*, "A 5 mW MPEG4 SP encoder with 2D bandwidth-sharing motion estimation for mobile applications," in *IEEE ISSCC Dig. Tech. Papers*, Feb. 2006, pp. 1626–1635.

[10] A. P. Chandrakasan, S. Sheng, and R. W. Brodersen, "Low-power CMOS digital design," *IEEE J. Solid State Circuits*, vol. 6, no. 5, pp. 473–484, Apr. 1992.

[11] A. Raychowdhury *et al.*, "Error detection and correction in microprocessor core and memory due to fast dynamic voltage droops," *IEEE J. Emerg. Sel. Topics Circuits Syst.*, vol. 1, no. 3, pp. 208–217, Sep. 2011.

[12] I. J. Chang, J. Park, K. Kang, and K. Roy, "Fast and accurate estimation of SRAM read and hold failure probability using critical point sampling," *IET Circuits, Devices Syst.*, vol. 4, no. 6, pp. 469–478, Nov. 2010.

[13] F. J. Kurdahi, A. Eltawil, K. Yi, S. Cheng, and A. Khajeh, "Low-power multimedia system design by aggressive voltage scaling," *IEEE Trans. VLSI Syst.*, vol. 18, no. 5, pp. 852–856, May 2010.

[14] J. Park, J. Park, and S. Bhunia, "VL-ECC: Variable data-length error correction code for embedded memory in DSP applications," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 61, no. 2, pp. 120–124, Feb. 2014.

[15] F. Frustaci, M. Khayatzadeh, D. Blaauw, D. Sylvester, and M. Alioto, "SRAM for error-tolerant applications with dynamic energy-quality management in 28 nm CMOS," *IEEE J. Solid-State Circuits*, vol. 50, no. 5, pp. 1310–1323, May 2015.

[16] L. Chang *et al.*, "A 5.3 GHz 8T-SRAM with operation down to 0.41 V in 65 nm CMOS," in *Symp. VLSI Circuits Dig.*, Jun. 2007, pp. 252–253.

[17] I. J. Chang *et al.*, "A 32 kb 10T sub-threshold SRAM array with bit-interleaving and differential read scheme in 90 nm CMOS," *IEEE J. Solid State Circuits*, vol. 44, no. 2, pp. 650–658, Feb. 2009.

[18] D. Anh-Tuan, J. Y. S. Low, J. Y. L. Low, Z.-H. Kong, X. Tan, and K.-S. Yeo, "An 8T differential SRAM with improved noise margin for bit-interleaving in 65 nm CMOS," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 58, no. 6, pp. 1252–1263, Jun. 2011.

[19] I. J. Chang, D. Mohapatra, and K. Roy, "A priority-based 6T/8T hybrid SRAM architecture for aggressive voltage scaling in video applications," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 2, pp. 101–112, Feb. 2011.

[20] J. Kwon, I. J. Chang, I. Lee, H. Park, and J. Park, "Heterogeneous SRAM cell sizing for low-power H.264 applications," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 59, no. 10, pp. 2275–2284, Oct. 2012.

[21] M. E. Sinangil and A. P. Chandrakasan, "application-specific SRAM design using output prediction to reduce bit-line switching activity and statistically gated sense amplifiers for up to 1.9 × lower energy/access," *IEEE J. Solid-State Circuits*, vol. 49, no. 1, pp. 107–117, Jan. 2014.

[22] M. Imani, S. Patil, and T. S. Rosing, "Hierarchical design of robust and low data dependent FinFET based SRAM array," in *Proc. IEEE/ACM Int. Symp. Nanoscale Archit. (NANOARCH)*, Jul. 2015, pp. 63–68.

[23] R. Bellman, "Dynamic programming and Lagrange multipliers," *Proc. Nat. Acad. Sci. USA*, vol. 42, no. 10, pp. 767–769, 1956.

[24] H. W. Kuhn and A. W. Tucker, "Nonlinear programming," in *Proc. 2th Berkeley Symp. Math. Statist. Probab.*, Berkeley, CA, USA, 1951, pp. 481–492.

[25] G. Bjontegaard, *Calculation of Average PSNR Differences Between RD Curves*, document VCEG-M33 of ITU-T Q6/16, Austin TX, USA, Apr. 2001.

[26] S. Mukhopadhyay, H. Mahmoodi, and K. Roy, "Modeling of failure probability and statistical design of SRAM array for yield enhancement in nanoscaled CMOS," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 24, no. 12, pp. 1859–1880, Dec. 2005.

[27] M. Alioto, "Ultra-low power VLSI circuit design demystified and explained: A tutorial," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 59, no. 1, pp. 3–29, Jan. 2012.

[28] R. Sharma and S. S. Chopade, "Performance and area scaling of 6T SRAM using SOI MOSFET at 32 nm node," in *Proc. IEEE Int. Conf. Commun., Inf. Comput. Technol.*, Jan. 2015, pp. 1–5.

[29] S.-T. Zhou, S. Katariya, H. Ghasemi, S. Draper, and N. S. Kim, "Minimizing total area of low-voltage SRAM arrays through joint optimization of cell size, redundancy, and ECC," in *Proc. IEEE Int. Conf. Comput. Design*, Oct. 2010, pp. 112–117.

[30] C. E. Rhee, J.-S. Jung, and H.-J. Lee, "A real-time H.264/AVC encoder with complexity-aware time allocation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 12, pp. 1848–1862, Dec. 2010.

[31] Joint Video Team,*Reference Software JM13.2*. [Online]. Available: http://iphome.hhi.de/suehring/tml

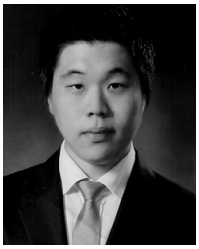
[32] M.-H. Chang, Y.-T. Chiu, and W. Hwang, "Design and iso-area $V_{\min}$ analysis of 9T subthreshold SRAM with bit-interleaving scheme in 65-nm CMOS," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 59, no. 7, pp. 429–433, Jul. 2012.

[33] J. S. Wang, P. Y. Chang, T. S. Tang, J. W. Chen, and J. I. Guo, "Design of subthreshold SRAMs for energy-efficient quality-scalable video applications," *IEEE Trans. Emerg. Sel. Topics Circuits Syst.*, vol. 1, no. 2, pp. 183–192, Jun. 2011.

[34] C.-C. Ju *et al.*, "A 0.5 nJ/Pixel 4 K H.265/HEVC codec LSI for multi-format smartphone applications," *IEEE J. Solid-State Circuits*, vol. 51, no. 1, pp. 56–67, Jan. 2016.

[35] S.-Y. Wu *et al.*, "A highly manufacturable 28 nm CMOS low power platform technology with fully functional 64 Mb SRAM using dual/tripe gate oxide process," in *Proc. Symp. VLSI Technol.*, Jun. 2009, pp. 210–211.

[36] N. Verma and A. P. Chandrakasan, "A 256 kb 65 nm 8T subthreshold SRAM employing sense-amplifier redundancy," *IEEE J. Solid-State Circuits*, vol. 43, no. 1, pp. 141–149, Jan. 2008.
[37] P. Sharma, A. K. Gundu, and M. S. Hashmi, "Modeling and yield estimation of SRAM sub-system for different capacities subjected to parametric variations," in *Proc. Int. Symp. VLSI Design Test*, May 2016, pp. 1–6.
[38] M. Grant and S. Boyd. *CVX: MATLAB Software for Disciplined Convex Programming, Version 1.21 2010*. [Online]. Available: http://cvxr.com/cvx
[39] T. Anantharaman, M. S. Campbell, and F.-H. Hsu, "Singular extensions: Adding selectivity to brute-force searching," *Artif. Intell.*, vol. 43, no. 1, pp. 99–109, Apr. 1990.



Hyun Kim received the B.S., M.S., and Ph.D. degrees in electrical engineering and computer science from Seoul National University, Seoul, South Korea, in 2009, 2011, and 2015, respectively. In 2015, he joined the BK21 Creative Research Engineer Development for IT, Seoul National University, where he is currently an Assistant Professor. His research interests are the areas of algorithm, computer architecture, and SoC design for low-complexity multimedia applications.

Ik Joon Chang received the B.S. degree *(summa cum laude)* in electrical engineering from Seoul National University, Seoul, South Korea, and the M.S. and Ph.D. degrees from the School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN, USA, in 2005 and 2009, respectively.

After graduation, he was with the Samsung Electronics NAND Flash Design Team for two years. He is currently an Assistant Professor with Kyung Hee University, South Korea.

Dr. Chang was awarded by Samsung Scholarship Foundation in 2005.

Hyuk-Jae Lee received the B.S. and M.S. degrees in electronics engineering from Seoul National University, South Korea, in 1987 and 1989, respectively, and the Ph.D. degree in electrical and computer engineering from Purdue University, West Lafayette, IN, USA, in 1996.

From 1998 to 2001, he was with the Server and Workstation Chipset Division, Intel Corporation, Hillsboro, OR, USA, as a Senior Component Design Engineer. From 1996 to 1998, he was on the Faculty of the Department of Computer Science, Louisiana Tech University, Ruston, LA, USA. In 2001, he joined the School of Electrical Engineering and Computer Science, Seoul National University, South Korea, where he is currently a Professor. He is a Founder of Mamurian Design, Inc., a fabless SoC design house for multimedia applications. His research interests are in the areas of computer architecture and SoC design for multimedia applications.