

Received November 15, 2020, accepted December 9, 2020, date of publication December 14, 2020, date of current version December 28, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3044496

A Novel ReRAM-Based Architecture of Field Sequential Color Driver for High-Resolution LCoS Displays

YOUNGSUN HAN¹, (Member, IEEE), DONGMIN KIM¹, AND YONGTAE KIM², (Member, IEEE)

¹Department of Computer Engineering, Pukyong National University, Busan 48513, South Korea

²School of Computer Science and Engineering, Kyungpook National University, Daegu 41566, South Korea

Corresponding author: Yongtae Kim (yongtae@knu.ac.kr)

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korean Government (MSIT) under Grant 2019R1C1C1008789.

ABSTRACT Liquid crystal on-silicon (LCoS) display is one of the most representative micro-display technologies, and is widely adopted in virtual reality (VR) and augmented reality (AR) devices thanks to a relatively simple structure using a semiconductor manufacturing process to realize high-resolution displays. However, the structural complexity to handle color frames by field sequential color (FSC) scheme hinders more widespread adoptions of the LCoS displays. In this article, to resolve the problem, we propose a novel FSC driver architecture using resistive random access memory (ReRAM) that diminishes the driver's structural complexity with matrix-vector multiplications. The proposed architecture leverages fast matrix-vector multiplications with a memristor crossbar array to expedite the FSC operation that extracts the individual red, green, and blue color sub-frames from an entire image. We present the hardware performance of our architecture that is implemented using the crossbar array and peripheral circuits. Compared to the conventional static random access memory (SRAM)-based architecture, we confirm that the proposed design is much superior in terms of chip size, leakage power, and frame rate in various image resolutions. Specifically, the chip size and leakage power are reduced by up to 96% and 99%, respectively, and the frame rate is improved by up to 36%. We also analyze image quality loss caused by ReRAM read and write noise.

INDEX TERMS ReRAM (resistive random access memory), memristor crossbar, LCoS (liquid crystal-on-silicon), FSC (field sequential color), high-resolution displays.

I. INTRODUCTION

Recently, as virtual reality (VR) and augmented reality (AR) technologies become more popular, a micro-display technology for wearable glasses is being spotlighted [1]–[4]. Micro-displays can be produced from various display technologies such as organic light-emitting diode (OLED), liquid crystal on-silicon (LCoS), liquid crystal display (LCD), and micro LED [5]–[10]. Their market demand is rapidly increasing since they are widely adopted in a range of electronic display devices, including a head-mounted display (HMD), head-up display (HUD), and projector [11]–[14]. Also, the micro-display technology faces new challenges of developing an energy-efficient micro-display device with high resolution, high luminance, and low latency while simplifying

its manufacturing process. Hence, many studies have been conducted to overcome the challenges [15]–[18]. LCoS is one of the most representative micro-display technologies, and it has been popularly employed in VR and AR devices, like Google Glass, and beam projectors since it can realize high resolution with a relatively uncomplicated structure using a typical semiconductor manufacturing process. The LCoS display constructs a single red, green, and blue (RGB) color frame by using the field sequential color (FSC) scheme to overlap three individual red, green, and blue color sub-frames sequentially [19]–[22]. This approach is completely different from those of other display technologies, including OLED, LCD, and micro LED, that illuminate the RGB color lights at each pixel of the displays. Hence, the FSC driver is typically designed to split each input frame into three different R, G, and B sub-frames and store them in three separate buffers and then sequentially output to the LCoS display

The associate editor coordinating the review of this manuscript and approving it for publication was Cihun-Siyong Gong².

panel [20], [23], [24]. Although the LCoS technology has the advantages of small size, high resolution, and low power consumption, and the ease of large-scale integration, the complexity of FSC drivers restricts their widespread use.

In this article, we propose a novel FSC driver architecture using resistive random access memory (ReRAM)’s crossbar structure to reduce the structural complexity of the LCoS micro-displays. To the best of our knowledge, this is the first approach that leverages matrix-vector multiplications for the LCoS FSC method and proposes its driver architecture using ReRAM technology. We first present a new algorithm with matrix-vector multiplications for the FSC operation, and describe the internal structure of the FSC driver when implementing the algorithm. We also show the memristor crossbar organization of the FSC driver and the detailed implementation of its peripheral circuits. We introduce an optimization using a multi-level cell (MLC) ReRAM-based design of the driver to further reduce the area overhead of the basic single-level cell (SLC) ReRAM-based design.

To demonstrate the superiority of our proposed FSC driver in terms of chip size, power consumption, throughput, i.e., frame rate, and endurance, we evaluate the performance of the FSC drivers for LCoS micro-displays with standard definition (SD), high definition (HD), full high definition (FHD), and ultra high definition (UHD) resolutions. As a result, compared to the existing static random access memory (SRAM)-based FSC driver, we obtained up to 84% and 98% reductions of chip size and leakage power, respectively, and increased the frame rate by up to 36%. We also found that the chip size was diminished up to 96% further by employing the MLC ReRAM-based design instead of the SLC one, although some image quality loss occurred.

In summary, this article makes the following key contributions:

- We propose a new methodology that reduces the structural complexity of the LCoS FSC driver using matrix-vector multiplications.
- We present a ReRAM-based FSC driver architecture for LCoS micro-displays and its implementation details, including memristor crossbar organization and peripheral circuits.
- We convince that ReRAM is definitely applicable to the FSC driver with its outstanding performance, especially in terms of chip size and leakage power.

The rest of this article is organized as follows. Section II describes the background and related works. Section III presents the ReRAM-based LCoS FSC driver architecture, and Section IV describes its hardware implementation. In Section V, we perform the performance evaluation. Finally, the conclusion is made in Section VI.

II. BACKGROUND AND RELATED WORKS

A. FSC FOR LCoS DISPLAYS

Figure 1 shows the timing diagram used in FSC method to display a single color field with RGB color frames. Each color frame must be loaded prior to setting and illumination;

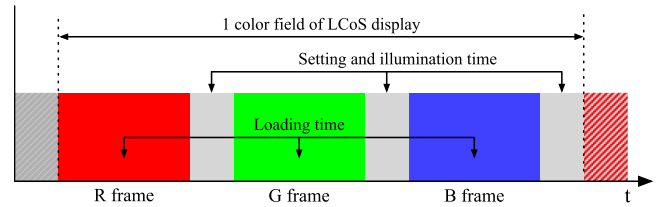


FIGURE 1. Display timing diagram for a single color field of an LCoS display. The R, G, and B frames are continuously output at identical intervals.

a short loading time is required to ensure that the refresh rate is sufficiently high, i.e., over 300 Hz [24]. Typically, each image pixel contains 24 bits in RGB format and is stored in a memory in succession to other pixels. To minimize the loading time of each color frame without latency, an FSC driver is designed to read image frames from a memory in burst and divide each of them into RGB color frames in advance before being loaded.

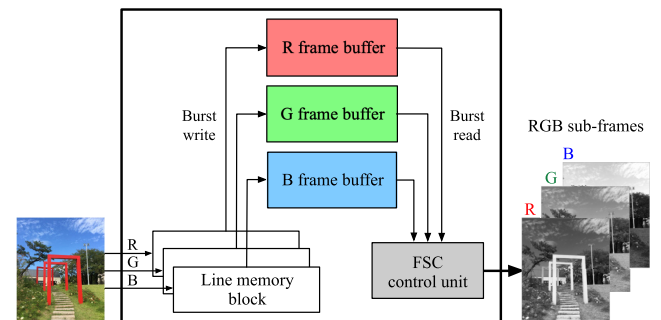


FIGURE 2. A simple structural diagram of the conventional FSC drivers. An SRAM is typically used to implement the memory blocks and RGB frame buffers, but it incurs an unavoidable resource overhead [20], [23].

Figure 2 depicts a brief architecture of a conventional FSC driver that extracts RGB color frames from an input image. The driver mainly consists of a line buffer, three different frame buffers for the three RGB colors, and a control unit. The line buffer temporarily holds image pixels read from memory; each pixel is split into the three different frame buffers for RGB colors. After the color splitting is performed on all pixels of the input image, each color frame is finally placed in its own frame buffer. Thus, the control unit can load all the color frames onto an LCoS display panel in sequence without any extra latency. SRAM is usually employed to implement the frame buffers but this incurs a considerable resource overhead [23].

B. MEMRISTOR CROSSBAR BASED ACCELERATION

Figure 3 presents a memristor crossbar array that is designed for a matrix-vector multiplication of X and V , as follows:

$$I_k = \sum_{i=1}^n X_{k,i} \times V_i \quad \text{for all } k, \text{ where } 1 \leq k \leq m \quad (1)$$

Since I_k is the sum of all the output currents from memristors $X_{k,i}$ when the input voltage V_i ranges over all i from

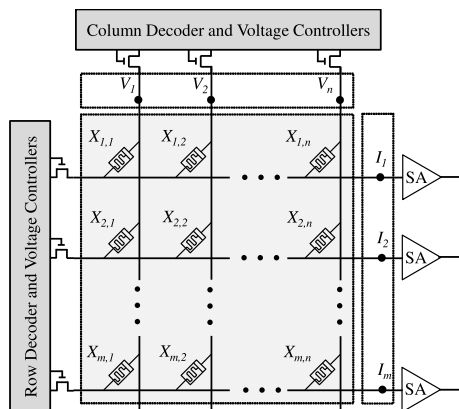


FIGURE 3. A brief structure of a memristor crossbar array for matrix-vector multiplications. V , I , and X are the input voltage, output current, and stored resistive conductance, respectively. SA stands for a sense amplifier [25].

1 to n , the matrix-vector multiplication is completed within one clock cycle by obtaining I_k for all k from 1 to m in parallel. The row and column decoders are designed to store a conductance matrix X in the crossbar array and to apply a vector V as input voltages for their multiplication.

The memristor crossbar array can be leveraged in many applications, such as image compression and neural processing, and expedites matrix multiplications [26]–[29]. A memristor crossbar based-accelerator for a lossy two dimensional discrete wavelet transform (DWT) was proposed in [26]. The accelerator comprises a computational memristor crossbar that performs the multiply add operations, an intermediate memory array that stores the row-transformed coefficients, and a final memory that holds the compressed image. The crossbar array performs the transpose matrix-vector multiplications in both storage and computing modes. The former mode stores the analog conductance values of the crossbar in the form of a coefficient matrix, and the later mode performs the multiplication operations without disturbing memristor status. The crossbar is controlled by voltage pulse generators. The accelerator achieves $10\times$ reduction in the number of operations compared to a conventional digital implementation. This reduction leads to five orders of magnitude reduction in area, approximately $11\times$ improvement in energy efficiency, and $1.28\times$ faster in computation without any significant degradation in image quality.

In addition to the image compression, the neural processing also relies heavily on matrix-vector multiplications. Kim *et al.* developed a digital neuromorphic processor using a memristor crossbar-based synapse [29]. The memristive synaptic crossbar array stores not only multibit synaptic weight values but also neural network configuration data. The crossbar array efficiently multiplies and accumulates the presynaptic weight values of each neuron for inference, and is accessible both column-wise and row-wise to expedite the synaptic weight updating during learning. The pulse width modulator (PWM) based voltage pulse generators and two types of analog-to-digital converters (ADCs) are used to read

and write the multibit memristor crossbar array. The crossbar with 64K memristor cells shows $12.8\times$ more area efficient than the conventional SRAM-based crossbar array without loss of functionality.

III. ReRAM-BASED ARCHITECTURE FOR LCoS FSC

In this section, we propose a new methodology to perform the FSC operation using ReRAM technology and present an architectural design of the FSC driver for LCoS displays.

A. PROPOSED METHODOLOGY

To reduce the resource overheads of existing SRAM-based LCoS FSC drivers by using ReRAM technology, we propose a novel FSC methodology that separates each input image into RGB color frames using the fast matrix-vector multiplications of Figure 3.

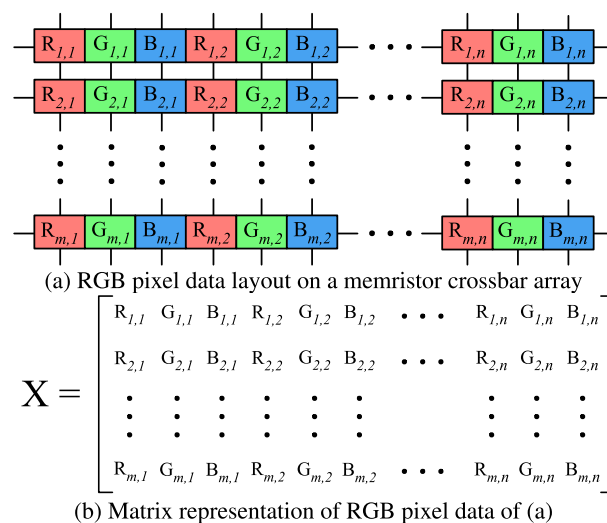


FIGURE 4. Pixel data layout on our proposed methodology.

Figure 4 (a) shows how image pixel data are deployed on the memristor crossbar array in our proposed methodology. Note that each pixel of an RGB format is composed of three bytes for red, green, and blue. All the pixels in an image row are ordered on the same row of the crossbar. $R_{i,j}$, $G_{i,j}$, and $B_{i,j}$ indicate the R, G, and B bytes of the pixel at the (i, j) position of the crossbar array, respectively. Figure 4 (b) presents the matrix representation of all RGB pixel data in the crossbar array conceptually. To obtain a specific color frame from the crossbar array, we need to perform n matrix-vector multiplications to sequentially extract n pixel columns of that color, where n is the pixel width of the input image.

Figure 5 exhibits how to perform n matrix-vector multiplications that extract a red frame from an RGB image frame, which is loaded on the crossbar array X . Each column of the second $3n \times n$ matrix contains the input voltages that return each pixel column of the red frame, i.e., the rightmost $m \times n$ matrix, on matrix-vector multiplication with X . In other words, each column of the $3n \times n$ matrix has a one-hot encoded array of size $3n$ to get a single column of X for

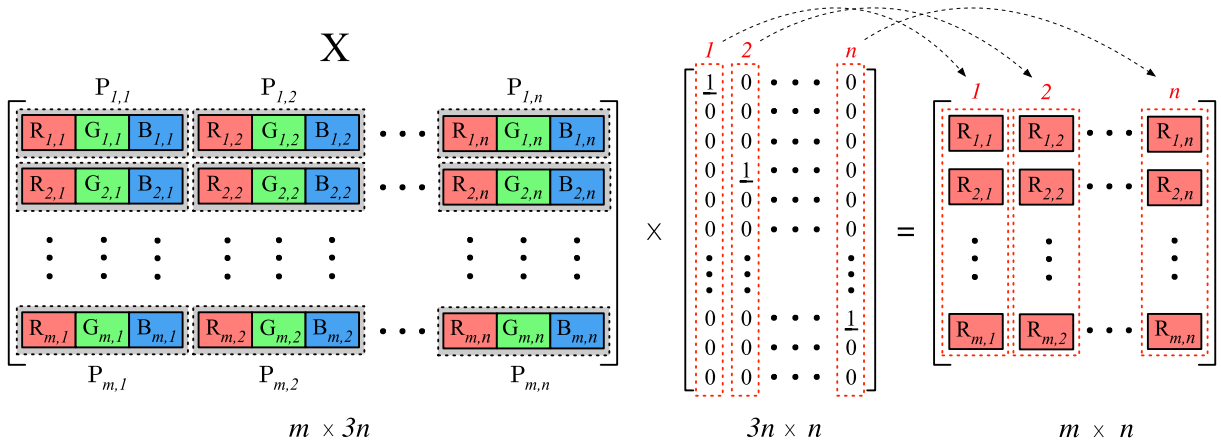


FIGURE 5. Extraction of a red frame from an RGB image frame. $P_{i,j}$ indicates the (i,j) pixel of the image frame mapped on the crossbar array X , and each pixel is assumed to contain only three bytes of R, G, and B. $R_{i,j}$, $G_{i,j}$, and $B_{i,j}$ mean the R, G, and B bytes of the $P_{i,j}$ pixel, respectively.

each matrix multiplication. Therefore, we can obtain all RGB frames of an input image sequentially, using only $3 \times n$ matrix-vector multiplications.

B. PROPOSED ARCHITECTURE

Figure 6 illustrates the architecture of our ReRAM-based LCoS FSC driver. It mainly consists of the following four components: memristor crossbar array, global decoder for column extraction, PWM array, and control unit.

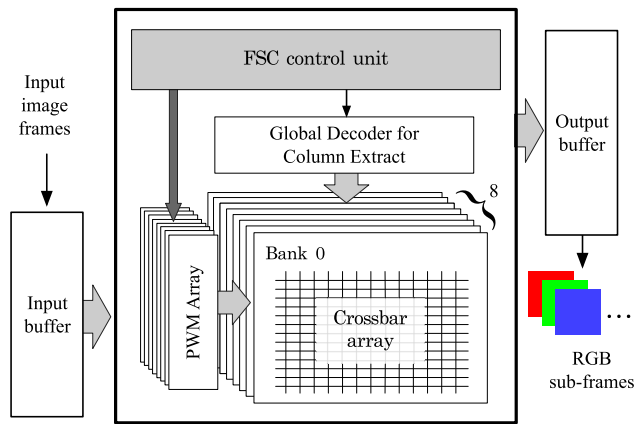


FIGURE 6. Proposed FSC architecture using ReRAM.

First of all, the crossbar array consists of eight banks that store each bit of each byte of an RGB image frame into each bank in parallel, respectively. It means that the i -th bit of each byte is written into the i -th bank. Each cross-point of the memristor crossbar can be theoretically designed to contain RGB color values from 0 to 255, as shown in Figure 4 (a). However, if we adopt such analog memristor-based crossbar as in [27], [28], [30] for the driver, we will face severe write latency and inevitable input/output errors, along with the chip area and energy overheads caused by multi-level ADCs. We thus employ the binary structure of eight banks

that store each bit at each cross-point location. In other words, an SLC ReRAM is employed for each bank. It is possible to obtain single-byte columns via matrix-vector multiplication, as in Figure 5, using the SLC structure. The data layout of the crossbar array is in the form of a two-dimensional matrix, as depicted in Figure 4 (b), except that each memristor cross-point contains a one-bit value. Hence, we store each image frame that will be subjected to FSC processing in the crossbar array and extract the RGB color frames from the image frame by applying matrix-vector multiplications consecutively.

Second, the global column decoder is designed to determine the column address in the crossbar array where all color component values of the column to be extracted are stored. The extracted color components, which is one of the R, G, and B, are transferred to the output buffer. The decoder hierarchically selects the column of the array to be extracted; it first decodes the desired column position of the image frame and then selects one of the R, G, or B line. For example, to extract all G components of the k -th column of the image frame, the decoder first determines the k -th column position of the frame and then selects the G component from the R, G, and B lines. Note that the write operation from the input buffer to the crossbar exploits the internal row and column decoders of the array; we also discuss this feature in Section IV-A.

Third, the PWM array performs both the read and write operations of the crossbar array by generating appropriate voltage pulses of various widths. For read operation, i.e., column extraction, the PWM array delivers a one-hot binary vector input for the matrix-vector multiplication to extract one specific byte column at any one time from the image frame data stored in the conductive filaments of the crossbar array. In Figure 5, each column of the $3n \times n$ matrix represents a one-hot binary vector, and the corresponding result column of the $n \times n$ matrix does the extracted byte vector, respectively. For write operation, the PWM array converts sequential RGB bytes from the input buffer into PWM signals, i.e., voltage

pulses, that vary the conductance of the memristor cells in the crossbar array. In the MLC-based design, an input value to be written is converted to a voltage pulse with an appropriate pulse duration by the PWM array. In other words, a larger input value leads to a longer pulse duration. Note that the read operation needs a voltage pulse of a single fixed width, whereas the write operation in the MLC requires that of various widths depending on the RGB values to write.

Finally, the control unit stores the RGB image frames coming through the input buffer into the memristor crossbar array, and also controls the entire process of obtaining separated R, G, and B sub-frames by combining the byte columns derived from the multiple matrix-vector multiplications.

As mentioned above, matrix-vector multiplications on the memristor crossbar array are required to extract the R, G, and B sub-frames from each input RGB image. Thus, our proposed LCoS FSC driver supports two different modes of operation, i.e., image writing and color sequencing, using the crossbar array. In the image writing mode, each input image frame is sequentially read from an external memory using 64-byte burst reads, and is relocated into eight banks of the crossbar array by interleaving bit-by-bit through the input buffer. Next, the driver sequentially extracts the R, G, and B color frames through iterative matrix-vector multiplication from the stored image frame on the crossbar array in the color sequencing mode. We can perform the FSC operations to an input image stream by executing both modes repeatedly.

IV. HARDWARE IMPLEMENTATION

In this section, we describe the organization of the memristor crossbar of our FSC driver and the peripheral circuitry.

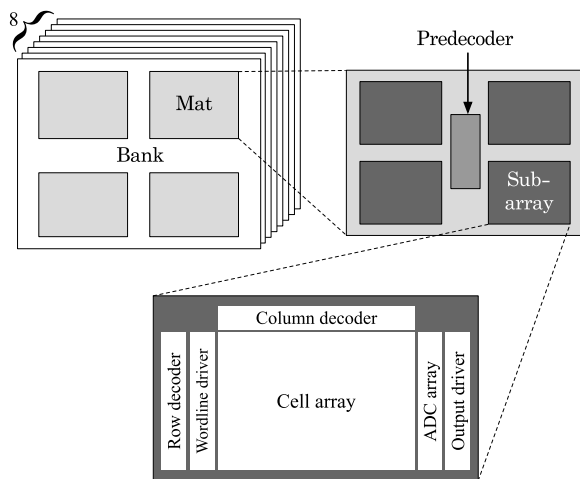


FIGURE 7. Memristor crossbar array organization.

A. MEMRISTOR CROSSBAR ORGANIZATION

Figure 7 illustrates the internal structure of the memristor crossbar array that is used to design the crossbar array of the proposed driver, which is adapted from the memory array organization of representative non-volatile memory

simulators, i.e., NVSim [31] and Destiny [32]. The crossbar array features eight banks in the SLC ReRAM-based design; each bank is hierarchically organized into mats and sub-arrays. In the case of the MLC ReRAM-based design, we only require to use four banks for the FSC driver. The detail of the MLC ReRAM-based design is presented in Section IV-B. Each bank is designed to operate independently to store each bit of R, G, and B bytes of an input image frame simultaneously by exploiting bank-level parallelism. H-tree routing scheme is used to connect the multiple mats, and to associate multiple sub-arrays with a predecoder within mats. Each sub-array has a cell array, row and column decoders, an ADC array, and wordline and output drivers. The cell array is a memristor cell, i.e., SLC or MLC, -based crossbar wherein the RGB color data are stored. The row and column decoders determine the cross-point locations of the row and column addresses used for sequential writing of the pixel data bytes of the input image. The ADC array is employed to obtain digital values from the analog voltages of each column output in the cell array. The ADC can have different designs depending on whether the cell array is implemented as SLC or MLC. In other words, 1- and 2-bit ADCs are utilized for SLC and MLC ReRAM-based designs, respectively.

B. OPTIMIZATION USING MULTI-LEVEL CELL

As the image resolution of a micro-display is upscaled from SD to UHD, the size of the memristor crossbar array increases incrementally. Thus, we present an optimization of adopting an MLC ReRAM-based design of the proposed FSC driver to diminish this area overhead. Although this complicates the ADC and PWM circuits, we halved the area of the crossbar array using the MLC ReRAM-based design that stores two bits in each memristor cell. In the MLC ReRAM-based design, the FSC driver requires only four banks of the crossbar array differently from the SLC ReRAM-based design using eight banks. Also, the MLC ReRAM-based design requires a PWM-based voltage pulse generator to deliver pulses of different durations for the write operation and a two-bit ADC rather than a sense amplifier, i.e., one-bit ADC, for the read operation. The area reduction thus obtained is exhibited in Table 3.

C. PERIPHERAL CIRCUIT DESIGN

The conductance of a memristor cell can be incrementally adjusted by modulating either the pulse width of its constant input voltage or the amplitude of the voltage input [33], [34]. Realizing a pulse amplitude modulator (PAM) requires programmable analog circuits that are difficult to integrate into a digital system, whereas the PWM is more appropriate for a digital architecture and is relatively easy to implement, as only digital logic gates are required. The PWM can be implemented with either delay lines or a digital counter [29], [35]. The delay line-based digital PWM requires many delay cells to produce different pulse widths, and a large multiplexer to select one output from these cells; the area and power overheads are considerable [35]. Seriously,

the delay cells are susceptible to process, voltage, and temperature (PVT) variations, introducing pulse width variability that may disrupt writing to the memristor cells. Therefore, we adopt the counter-based digital PWM since it can be implemented with a few digital components to produce voltage pulses with various widths and is more vulnerable to the PVT variations.

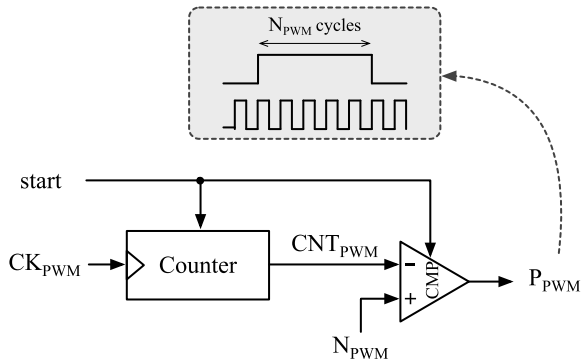


FIGURE 8. Pulse width modulator-based voltage pulse generator.

Figure 8 shows the PWM-based voltage pulse generator circuit which is comprised of a counter and a comparator. Once the start signal is asserted, the counter begins to record the number of cycles of the PWM clock signal CK_{PWM} and the comparator outputs “1”. The comparator continually checks that the counter output value CNT_{PWM} reaches the desired number of cycles N_{PWM} . When this occurs, i.e., $CNT_{PWM} = N_{PWM}$, the comparator outputs “0”. Therefore, the pulse width t_{PWM} generated by the counter-based PWM pulse generator is expressed by

$$t_{PWM} = \frac{N_{PWM}}{f_{CK_{PWM}}} \quad (2)$$

where $f_{CK_{PWM}}$ is the frequency of the clock signal CK_{PWM} . While the SLC-based design uses a fixed value of N_{PWM} , the MLC-based one requires various N_{PWM} values, which varies according to the value to be written. Note that a larger value of N_{PWM} incurs a longer voltage pulse, leading to an higher value to be written to the memristor cells in the crossbar array. As discussed in Section III-B, the entire color component values of a column of the input image are extract by the matrix-vector multiplications. Therefore, we design the decoder to select an desired column to be read in the crossbar array. Figure 9 depicts the global decoder for column extraction. It consists of an N -bit column decoder and N color decoders hierarchically, where N is the width of the input image. The column decoder is a binary $\log_2 N$ -to- N decoder that determines the address of the column position to be extracted in the image. The color decoder accepts a color index input in binary form and an enable signal, and chooses one of the R, G, and B lines or deasserts all the lines. Note that the enable signal is the output of the column decoder. In other words, the enable signal of the corresponding color decoder

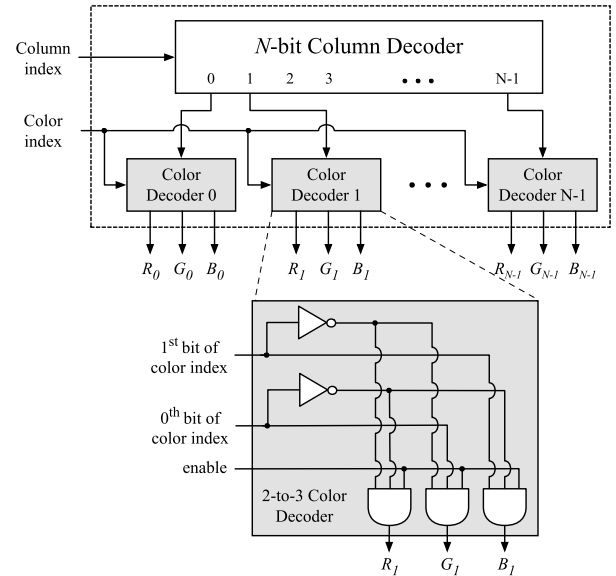


FIGURE 9. Global decoder for column extraction.

is “1” when the desired column is selected by the column decoder. When the enable signal is asserted, the desired color component, which is one of R, G, and B, of the column determined by the column decoder is selected. Otherwise, no R, G, or B line is selected and the outputs of all the lines are “0”.

Algorithm 1 FSC Extraction Algorithm

INPUT N : input image width.
OUTPUT $ColorIndex$: RGB color index, $ColumnIndex$: column index of the input image.

```

1: procedure FSC_EXTRACT( $N$ )
2:   for  $i \leftarrow 0$  to 2 do
3:     // To generate the color index
4:      $ColorIndex = i$ 
5:     for  $j \leftarrow 0$  to  $N - 1$  do
6:       // To generate the column index
7:        $ColumnIndex = j$ 
8:     end for
9:   end for
10: end procedure

```

Algorithm 1 briefly describes the FSC extraction flow using the global decoder. The input to the algorithm is the input image width N and the outputs are the color and column indices. The color index is first selected among R, G, and B and the column index is then chosen from 0 to $N - 1$ for the all three color components. Once the color and column indices are determined, the PWM array sends the readout signals to the desired RGB column of the crossbar array. Then, the M color values of the corresponding column are extracted by the ADC array and the output driver of the cell array in the crossbar, where M is the height of the input image.

V. EVALUATION

A. EXPERIMENTAL SETUP

In this section, we explain the experimental methodology to evaluate the performance results of our FSC driver compared to the conventional SRAM-based one. We also present two distinct SLC and MLC ReRAM-based designs of the proposed driver to explore the impact of area optimization described in Section IV-B.

First, to assess the hardware performance of the proposed FSC driver in terms of chip size, energy, leakage power, and read/write latency, we used Destiny V2 [32] and Synopsys Design Compiler [36] for evaluating its memristor crossbar array and peripheral circuits, respectively. Also, the peripheral circuits were designed using Verilog HDL and synthesized using a standard cell library. Note that we used 65-nm technology node to evaluate the hardware performance.

TABLE 1. ReRAM Simulation Parameters.

	SLC ReRAM	MLC ReRAM
Cell Structure	1T1R (HfO ₂)	1T1R (HfO ₂)
Cell Area (F^2)	20	20
Bits per Cell	1	2
Num. of Banks	8	4
Sensing Scheme	PSRC Current	PSRC Current
Read Scheme	Normal Read	Read and Compare
Write Scheme	Normal Write	Reset before Set

The Destiny simulator is based on NVSim [31], which is one of the representative non-volatile memory simulators, and has been extended to further simulate MLC-based memories. The simulator was configured to evaluate the memristor crossbar arrays using the characteristics presented in [37] while optimizing to minimize the write energy-delay product (EDP). Table 1 presents some simulation parameters of the SLC and MLC ReRAM. Both designs employ HfO₂-based 1T1R cells, i.e., one NMOS switch transistor and one bipolar resistive memory, and the cell area is 20 F^2 [38]. Parallel-series reference-cell (PSRC) current sensing is applied to deliver fast sensing speed. The MLC design uses a read-and-compare method that checks for faults with two successive reads and one comparison for read operations. It applies an iterative write-and-verify method of reset-before-set scheme for write operations, but the SLC design adopts normal read and write. The simulator estimates the performance of SRAM as in [39] while it is configured that each cell has 146 F^2 of area and comprises six transistors, i.e., 6T.

We estimated the performance of four different FSC driver implementations for the image resolutions, i.e., SD, HD, FHD, and UHD, shown in Table 2. The table presents the simulation parameters for each resolution, i.e., the total pixels and bytes, memory capacity, and internal bank capacity organization. Second, we calculated the frame rate, i.e., frame per second (FPS), based on the estimated read/write latency. Finally, we used CrossSim simulator [40], [41] to study the

TABLE 2. Simulation Parameters by Image Resolution.

Resolution	SD	HD	FHD	UHD
Width (W)	720	1,280	1,920	3,840
Height (H)	576	720	1,080	2,160
Pixels	414,720	921,600	2,073,600	8,294,400
Bytes	1,244,160	2,764,800	6,200,800	24,883,200
Memory Capacity (KBytes)	2,048	4,096	8,192	32,768
Bank Capacity (KBytes)	256	512	1,024	4,096
Subarray Size	256×512	256×512	256×512	256×512

image quality loss of output images regarding each combination of two ReRAM cell designs, i.e., SLC and MLC, and four image resolutions. Table 3 lists the performance results of the proposed and SRAM-based drivers.

B. PERFORMANCE ANALYSIS

1) CHIP SIZE

The proposed ReRAM-based FSC driver using both SLC and MLC shows greatly reduced chip size compared to those of the SRAM-based drivers over the all image sizes. Specifically, when comparing the area of our design using SLC with that of the SRAM-based, we can have 3.30×, 4.52×, 4.54×, and 6.19× better efficiencies when handling the SD, HD, FHD, and UHD images, respectively. As the image resolution increases, our design shows a better area efficiency because the SRAM size increases linearly while the memristor crossbar size increases a little as the number of pixels of the images increases. Also, we can obtain further area reduction of 73.56%, 73.89%, 74.68%, and 74.87%, respectively, in the SD, HD, FHD, and UHD, when using the MLC-based array. It is worth noting that the proposed peripheral circuits occupy a very little area (< 4%) of the entire chip. In short, the MLC ReRAM allows the FSC driver to reduce the chip size by more than 90% (range: 92.28% ~ 95.97%) compared to the SRAM at all the resolutions. This considerable area reduction stems from the high density provided by memristor. Therefore, the memristor-based crossbar array is certainly appealing in terms of chip size for realizing the FSC driver.

2) ENERGY AND LEAKAGE POWER

We obtained the energy consumption for the read and write operations as well as the leakage power to compare overall power efficiency of the drivers. The SRAM-based design consumes less energy than the proposed ReRAM-based one. For example, the driver with the SRAM consumes the energy of only 3.94 nJ while the energy of our design with SLC reaches 18.47 nJ in the SD resolution, which is 4.7× more energy consumption. The energy consumption of the SRAM-based design increases somewhat with the image sizes, which is not the case of our design with both SLC and MLC. In addition, the energy consumed by the peripherals in the proposed design is negligible and does not affect the total energy

TABLE 3. Performance Results.

		Area (mm^2)			Energy (nJ)			Leakage (mW)			Latency (ns)		Frame rate (FPS)
		Array	Peri.	Total	Array	Peri.	Total	Array	Peri.	Total	Read	Write	
SRAM	SD	11.30	1.24e-02	11.32	3.93	9.93e-03	3.94	3,347.59	6.71e-04	3,347.59	3.57	3.55	7,224
	HD	22.36	1.24e-02	22.37	6.62	1.00e-02	6.63	6,680.39	6.76e-04	6,680.39	3.70	3.65	3,149
	FHD	44.82	1.25e-02	44.84	9.87	1.01e-02	9.88	13,333.76	6.80e-04	13,333.76	3.75	3.68	1,384
	UHD	178.22	1.25e-02	178.23	20.76	1.01e-02	20.77	53,281.20	6.82e-04	53,281.20	4.71	4.26	286
ReRAM (SLC)	SD	3.41	2.53e-02	3.43	18.46	1.14e-02	18.47	72.54	1.44e-03	72.54	4.01	6.40	7,514
	HD	4.92	2.75e-02	4.95	19.13	1.14e-02	19.14	125.96	1.46e-03	125.96	4.05	6.43	3,409
	FHD	9.84	3.01e-02	9.87	19.57	1.15e-02	19.58	229.78	1.48e-03	229.78	4.07	6.44	1,539
	UHD	28.74	3.77e-02	28.78	24.72	1.18e-02	24.73	956.35	1.53e-03	956.35	4.23	6.48	389
ReRAM (MLC)	SD	0.87	3.48e-02	0.91	15.96	1.94e-02	15.98	19.43	2.14e-03	19.43	4.60	151.56	338
	HD	1.26	3.70e-02	1.29	16.07	1.94e-02	16.09	34.52	2.16e-03	34.53	4.63	151.58	152
	FHD	2.46	3.96e-02	2.50	16.29	1.95e-02	16.31	62.98	2.18e-03	62.98	4.68	151.63	67
	UHD	7.18	4.72e-02	7.23	17.46	1.98e-02	17.48	263.73	2.23e-03	263.73	4.81	151.64	16

significantly. Although the SRAM-based design exhibits a better energy performance than the proposed architecture, it consumes a very considerable leakage power, which reaches up to 53.28 W while the leakage of the proposed design with both SLC and MLC ReRAM never exceed 1 W in all the resolutions. Particularly, in the case of FHD resolution, the leakages consumed by the SRAM-based design are $58.03\times$ and $211.71\times$ more than our ReRAM-based designs with SLC and MLC, respectively. When considered all the resolution and the cell type, our ReRAM-based architecture can provide from 97.83% to 99.53% leakage power reduction. Hence, the ReRAM-based FSC driver is also very attractive in leakage power aspect while sacrificing the energy.

3) FRAME RATE

To confirm that the proposed FSC driver architecture satisfies the frame rate required by recent micro-displays, we evaluated the frame rates, i.e., FPS, of SLC and MLC ReRAM-based designs using the read/write latency (L_{read}/L_{write}) shown in Table 3. L_{write} was estimated under 10 ns in SLC ReRAM and about 160 ns in MLC ReRAM, respectively, and L_{read} was under 10 ns for both designs. These latency results were obtained by applying the write EDP optimization to the simulator that adopts the simulation parameters in [37]. The frame rates were calculated following Equation (3), (4), (5), and (6).

$$R_{frame} = \left\lfloor \frac{1}{T_{frame}} \right\rfloor \quad (3)$$

$$T_{frame} = T_{load} + T_{extract} \quad (4)$$

$$T_{load} = \left\lceil \frac{W \times H \times N_{pixel}}{N_{burst}} \right\rceil \times L_{write} \quad (5)$$

$$T_{extract} = (W \times N_{pixel}) \times L_{read} \quad (6)$$

The frame rate R_{frame} is defined as the floored inverse of the frame time T_{frame} that is required for the FSC operation of a single image frame. Also, the frame time consists of T_{load} and $T_{extract}$, which are the times for loading a single image

frame into the memristor crossbar array of the driver and extracting R, G, and B sub-frames from the crossbar array by performing successive matrix-vector multiplications, respectively. To determine T_{load} , we first obtained the total number of bytes in a single image frame by multiplying the image width W , the image height H , shown in Table 2, and the number of bytes per pixel N_{pixel} together. In our experiment, N_{pixel} is 3 because we assumed that each image pixel contains three bytes of RGB colors. After that, we divided the total number of bytes by the number of bytes per burst write N_{burst} , especially 64 in the experiment, and applied a ceiling function on the result. Finally, we multiplied the write latency L_{write} for each burst write access with the ceiling result to get the T_{load} of an image frame. In the case of $T_{extract}$, we could simply calculate the time by multiplying the number of matrix-vector multiplications for an FSC operation and the read latency L_{read} of each the multiplication. The number of matrix-vector multiplications was computed by multiplying the image width W and the number of bytes per pixel N_{pixel} , because the single multiplication yielded a one-byte column from all banks of the memristor crossbar array.

Thus, we confirm that most of the ReRAM-based drivers deliver frame rates faster than 60 FPS; it means that the drivers can support micro-displays of various resolutions. We also found that the SLC ReRAM-based driver is slightly better compared to the SRAM-based design in terms of frame rate. Although the MLC ReRAM design afforded only 16 FPS for UHD resolution because of the very long write latency, this is not a serious problem because most micro-displays do not currently deliver UHD. Moreover, We expect that the problem will soon be resolved by the improvements in semiconductor process technology and ReRAM write schemes.

4) IMAGE QUALITY LOSS

The analysis of the impact of non-idealities, variabilities, and physical limitations of the ReRAM technology is desirable. However, evaluating the noise of every cell of the crossbar

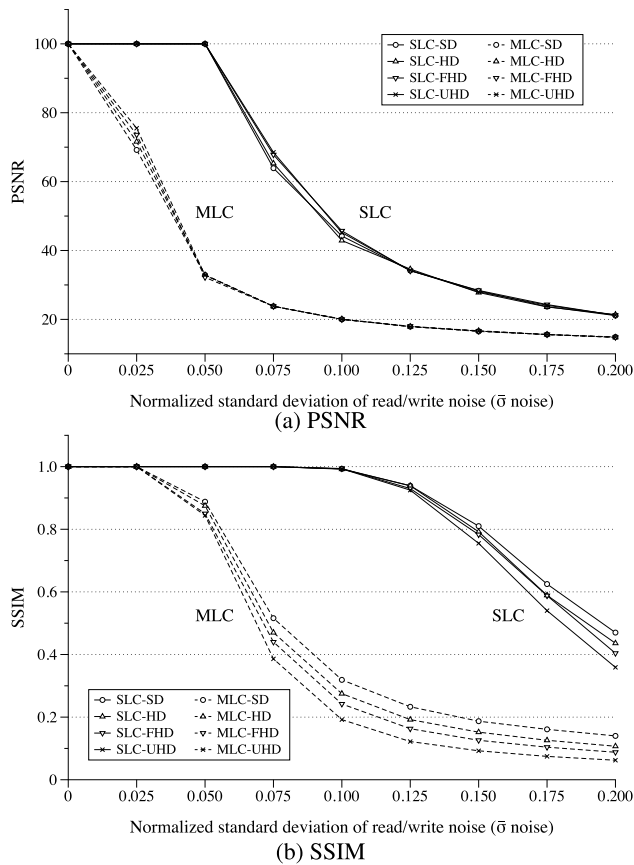
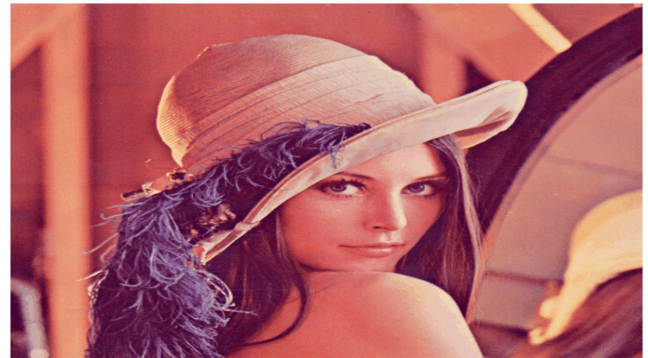


FIGURE 10. Image loss comparison in SLC and MLC ReRAM-based FSC drivers considering the read/write noise of a memristor crossbar array. PSNR and SSIM denote peak signal-to-noise ratio and structural similarity, respectively. Both indices are popularly used for image quality assessment.

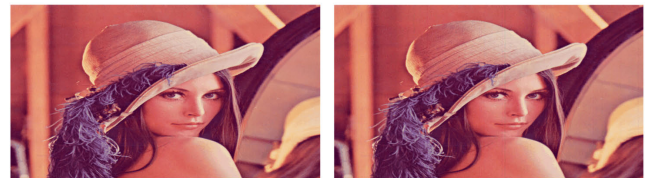
by thorough characterization of the non-idealities, variabilities, and physical limitations for the ReRAM and analyzing its impact on the entire driver is practically intractable. Therefore, we assumed that the noises induced by those uncertainties are accumulated at the read and write operations, and performed image quality loss simulations with the noises for the driver. Figure 10 shows the peak-signal-to-noise ratio (PSNR) and structural similarity index measure (SSIM) data; the output RGB sub-frames were compared to the input images. We estimated the image quality loss of our ReRAM-based FSC driver caused by its read/write noise. The FSC operations were simulated using a CrossSim simulator to obtain RGB sub-frames for SD, HD, FHD, and UHD inputs. The simulator was configured to use a numerical model for circuit approximation. Also, we employed the settings of DG_LOOKUP and G_PROPORTIONAL to model the read and write noises, respectively. The normalized standard deviation of the read/write noise ($\bar{\sigma}$ noise) was varied from 0 to 0.2 in steps of 0.025.

Figure 10 (a) shows that, for the SLC ReRAM-based design, the PSNR remains stable at 100 dB with 0 to 0.050 of $\bar{\sigma}$ noise, but decreases rapidly after 0.075. We found that the MLC ReRAM-based design severely reduces the PSNR,

even at low $\bar{\sigma}$ noise. The SSIM results in Figure 10 (b) also confirm that the proposed driver is applicable enough in terms of output image quality. The structural similarity of the output images in the SLC ReRAM-based design rarely decreases from 1.0 until at 0.1 of the $\bar{\sigma}$ noise, but declines gradually after that. The SSIM index of the MLC ReRAM-based design declines more steeply, but remains acceptable; the index is higher than 0.8 when the $\bar{\sigma}$ noise is 0.05.



(a) Original FHD image



(b) $\bar{\sigma}$ noise = 0.05 (SLC)

(c) $\bar{\sigma}$ noise = 0.05 (MLC)



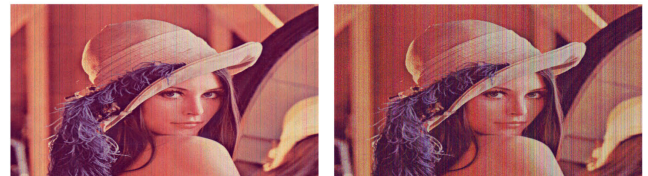
(d) $\bar{\sigma}$ noise = 0.10 (SLC)

(e) $\bar{\sigma}$ noise = 0.10 (MLC)



(f) $\bar{\sigma}$ noise = 0.15 (SLC)

(g) $\bar{\sigma}$ noise = 0.15 (MLC)



(h) $\bar{\sigma}$ noise = 0.20 (SLC)

(i) $\bar{\sigma}$ noise = 0.20 (MLC)

FIGURE 11. FSC operation results of an FHD image in SLC and MLC ReRAM-based designs for different normalized standard deviations of read/write noise ($\bar{\sigma}$ noise).

Figure 11 presents the FSC results simulated by employing the CrossSim simulator. We obtained eight output images using the original FHD “Lena” image. The bottom left four

images were produced by the SLC ReRAM-based design, and the bottom right four images were produced by the MLC design. The image quality loss in the SLC ReRAM-based design is barely noticeable, even at a $\bar{\sigma}$ noise of 0.15. However, the MLC ReRAM-based design begins to suffer from a significant quality loss at a $\bar{\sigma}$ noise of 0.10.

As mentioned above, the output image quality of the SLC ReRAM-based driver is significantly superior to that of MLC ReRAM-based one; thus we need to find the balance between the image quality loss and resource-saving by the MLC ReRAM-based design.

5) ENDURANCE

Finally, we discuss the durability of the proposed ReRAM-based FSC drivers by considering the read/write endurance of ReRAM. According to existing studies [42]–[47], ReRAM can be composed of various oxide materials such as ZnO, HfO₂, Ta/TaO_x, TaO_x/TiO₂, HfO₂/Pt, and so on; thus, it can provide various endurance between 10⁶ and 10¹² cycles. Since the proposed drivers perform write operations to all the crossbar array cells for loading each image frame, their lifetime is also strongly constrained by the number of writes required for the loading.

We evaluate the lifetime of the proposed drivers based on 60 FPS, i.e., 60 frames per second, which is the most commonly supported maximum frame rate for micro-displays. The SLC ReRAM-based driver only needs one write each time so that it can be used continuously for more than 30,000 years based on the endurance of 10¹² cycles and conservatively about 31.7 years based on 10⁹ cycles. In the case of MLC ReRAM, iterative writes may occur to each cell for a single load because we employ the write-and-verify method. However, it is definitely clear that we are still able to design the ReRAM-based FSC driver to obtain sufficient durability. We also expect that the ReRAM endurance improves more and more as technology advances.

VI. CONCLUSION

In this article, we proposed a novel FSC driver architecture based on ReRAM for high-resolution LCoS micro-displays. To the best of our knowledge, this is the first approach that applies matrix-vector multiplications for LCoS FSC and proposes a novel ReRAM-based FSC driver architecture. The proposed architecture adopts our FSC operation algorithm that exploits matrix-vector multiplications with the ReRAM's memristor crossbar array to expedite the individual R, G, and B color sub-frame extractions. We implemented the proposed driver that includes a memristor crossbar array, a global decoder for column extraction, a PWM array, and an FSC controller and demonstrated that our architecture is vastly superior to the conventional SRAM-based one. Specifically, our design reduces the chip size and leakage power by more than 96% and 99%, respectively, compared to the SRAM-based driver while supporting a frame rate faster than 60 FPS. Accordingly, our proposed ReRAM-based architecture is very appealing to realize low-cost and power-efficient

FSC drivers for high-resolution LCoS displays without a frame rate performance degradation.

REFERENCES

- [1] G. Haas, "40-2: Invited paper: Microdisplays for augmented and virtual reality," in *SID Symp. Dig. Tech. Papers*, 2018, vol. 49, no. 1, pp. 506–509.
- [2] A. Ferscha and S. Vogl, "Wearable displays for everyone!," *IEEE Pervasive Comput.*, vol. 9, no. 1, pp. 7–10, Jan. 2010.
- [3] E. Ackerman, "Google gets in your face," *IEEE Spectr.*, vol. 50, no. 1, pp. 26–29, Dec. 2013.
- [4] B. C. Kress and W. J. Cummings, "11-1: Invited paper: Towards the ultimate mixed reality experience: Hologens display architecture choices," in *SID Symp. Dig. Tech. Papers*, 2017, vol. 48, no. 1, pp. 127–131.
- [5] R. Asaki, S. Yokoyama, H. Kitagawa, S. Makimura, F. Abe, T. Yamazaki, T. Kato, M. Kanno, Y. Onoyama, E. Hasegawa, and K. Uchino, "18.1: A 0.23-in. High-resolution OLED microdisplay for wearable displays," in *SID Symp. Dig. Tech. Papers*, 2014, vol. 45, no. 1, pp. 219–222.
- [6] K.-Y. Chen, Y.-W. Li, K.-H. Fan-Chiang, H.-C. Kuo, and H.-C. Tsai, "P-181: Late-news poster: Color sequential front-lit LCOS for wearable displays," in *SID Symp. Dig. Tech. Papers*, 2015, vol. 46, no. 1, pp. 1737–1740.
- [7] N. Demoli, J. Gladić, D. Lovrić, and D. Abramović, "Digital holography using LCOS microdisplay as input three-dimensional object," *Optik*, vol. 194, Oct. 2019, Art. no. 162877.
- [8] B. Xue, H. Yang, F. Yu, X. Wang, L. Liu, Y. Pei, P. Lu, H. Xie, Q. Kong, J. Li, X. Yi, J. Wang, and J. Li, "Colour tuneable micro-display based on LED matrix," in *Optoelectronic Devices and Integration V*, vol. 9270, X. Zhang, H. Ming, and C. Yu, Eds. Bellingham, WA, USA: SPIE, 2014, pp. 240–247.
- [9] K.-H. Fan-Chiang, S.-T. Wu, and S.-H. Chen, "Fringing-field effects on high-resolution liquid crystal microdisplays," *J. Display Technol.*, vol. 1, no. 2, p. 304, Dec. 2005.
- [10] W. C. Chong, K. M. Wong, Z. J. Liu, and K. M. Lau, "60.4: A novel full-color 3LED projection system using R-G-B light emitting diodes on silicon (LEDoS) micro-displays," in *SID Symp. Dig. Tech. Papers*, 2013, vol. 44, no. 1, pp. 838–841.
- [11] P. Wartenberg, B. Richter, S. Brenner, M. Rolle, G. Bunk, S. Ulbricht, J. Baumgarten, C. Schmidt, M. Schober, and U. Vogel, "52-2: Invited paper: A new 0.64" 720p OLED microdisplay for application in industrial see-through AR HMD," in *SID Symp. Dig. Tech. Papers*, 2019, vol. 50, no. 1, pp. 717–720.
- [12] U. Vogel, D. Kreye, B. Richter, G. Bunk, S. Reckziegel, R. Herold, M. Scholles, M. Törker, C. Grillberger, J. Amelung, S.-T. Graupner, S. Pannasch, M. Heubner, and B. M. Velichkovsky, "8.2: Bi-directional OLED microdisplay for interactive HMD," in *SID Symp. Dig. Tech. Papers*, 2008, vol. 39, no. 1, pp. 81–84.
- [13] B. Mullins, P. Greenhalgh, and J. Christmas, "59-5: Invited paper: The holographic future of head up displays," in *SID Symp. Dig. Tech. Papers*, 2017, vol. 48, no. 1, pp. 886–889.
- [14] Z. J. Liu, W. C. Chong, K. M. Wong, K. H. Tam, and K. M. Lau, "A novel BLU-free full-color LED projector using LED on silicon micro-displays," *IEEE Photon. Technol. Lett.*, vol. 25, no. 23, pp. 2267–2270, Dec. 2013.
- [15] T. Fujii, C. Kon, Y. Motoyama, K. Shimizu, T. Shimayama, T. Yamazaki, T. Kato, S. Sakai, K. Hashikaki, K. Tanaka, and Y. Nakano, "4032 ppi high-resolution OLED microdisplay," *J. Soc. Inf. Display*, vol. 26, no. 3, pp. 178–186, 2018.
- [16] X. Zhang, P. Li, X. Zou, J. Jiang, S. H. Yuen, C. W. Tang, and K. M. Lau, "Active matrix monolithic LED micro-display using GaN-on-Si epilayers," *IEEE Photon. Technol. Lett.*, vol. 31, no. 11, pp. 865–868, Jun. 1, 2019.
- [17] B.-C. Kwak and O.-K. Kwon, "A 2822-ppi resolution pixel circuit with high luminance uniformity for OLED microdisplays," *J. Display Technol.*, vol. 12, no. 10, pp. 1083–1088, Oct. 2016.
- [18] U. Vogel, B. Beyer, M. Schober, P. Wartenberg, S. Brenner, G. Bunk, S. Ulbricht, P. König, and B. Richter, "77-1: Invited paper: Ultra-low power OLED microdisplay for extended battery life in NTE displays," in *SID Symp. Dig. Tech. Papers*, 2017, vol. 48, no. 1, pp. 1125–1128.
- [19] K. Fung, C. Waller, E. Eisenbrandt, H. L. Ong, T. Rost, and D. Wong, "32.1: Invited paper: Q-view: A compression technology for UHD resolution, low power, and low cost LCOS panels," in *SID Symp. Dig. Tech. Papers*, 2019, vol. 50, no. S1, pp. 342–344.

- [20] J. Lee, Y. Chung, and C.-G. Oh, "ASIC design of color control driver for LCOS (liquid crystal on silicon) micro display," *IEEE Trans. Consum. Electron.*, vol. 47, no. 3, pp. 278–282, Aug. 2001.
- [21] D. Vettese, "Liquid crystal on silicon," *Nature Photon.*, vol. 4, no. 11, pp. 752–754, 2010.
- [22] J.-P. Yang, H.-M. P. Chen, Y. Huang, Y.-C. Chang, F.-W. Lai, S.-T. Wu, C. Hsu, R. Tsai, and R. Hsu, "66-3: Submillisecond-response 10-megapixel 4k2k LCOS for microdisplay and spatial light modulator," in *SID Symp. Dig. Tech. Papers*, 2019, vol. 50, no. 1, pp. 933–936.
- [23] S.-W. Eo, J. G. Lee, M.-S. Kim, and Y.-C. Ko, "High performance and low power timing controller design for LCoS microdisplay system," in *Proc. Int. SoC Design Conf. (ISOCC)*, Nov. 2017, pp. 71–72.
- [24] M. Mori, T. Hatada, K. Ishikawa, T. Saishouji, O. Wada, J. Nakamura, and N. Terashima, "Mechanism of color breakup on field-sequential color projectors," in *SID Symp. Dig. Tech. Papers*, 1999, vol. 30, no. 1, pp. 350–353.
- [25] J. Yang, D. Strukov, and D. Stewart, "Memristive devices for computing," *Nature Nanotechnol.*, vol. 8, no. 1, p. 13, Jan. 2013.
- [26] Y. Halawani, B. Mohammad, M. Al-Qutayri, and S. F. Al-Sarawi, "Memristor-based hardware accelerator for image compression," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 26, no. 12, pp. 2749–2758, Dec. 2018.
- [27] C. Li, M. Hu, Y. Li, H. Jiang, N. Ge, E. Montgomery, J. Zhang, W. Song, N. Dávila, C. E. Graves, Z. Li, J. P. Strachan, P. Lin, Z. Wang, M. Barnell, Q. Wu, R. S. Williams, J. J. Yang, and Q. Xia, "Analogue signal and image processing with large memristor crossbars," *Nature Electron.*, vol. 1, no. 1, pp. 52–59, Jan. 2018.
- [28] M. Hu, C. E. Graves, C. Li, Y. Li, N. Ge, E. Montgomery, N. Davila, H. Jiang, S. R. Williams, J. J. Yang, Q. Xia, and J. Strachan, "Memristor-based analog computation and neural network classification with a dot product engine," *Adv. Mater.*, vol. 30, no. 9, 2018, Art. no. 1705914.
- [29] Y. Kim, Y. Zhang, and P. Li, "A reconfigurable digital neuromorphic processor with memristive synaptic crossbar for cognitive computing," *ACM J. Emerg. Technol. Comput. Syst.*, vol. 11, no. 4, pp. 1–25, Apr. 2015.
- [30] M. R. Mahmoodi, A. F. Vincent, H. Nili, and D. B. Strukov, "Intrinsic bounds for computing precision in memristor-based vector-by-matrix multipliers," *IEEE Trans. Nanotechnol.*, vol. 19, pp. 429–435, 2020.
- [31] X. Dong, C. Xu, Y. Xie, and N. P. Jouppi, "NVSIM: A circuit-level performance, energy, and area model for emerging nonvolatile memory," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 31, no. 7, pp. 994–1007, Jul. 2012.
- [32] S. Mittal, R. Wang, and J. Vetter, "DESTINY: A comprehensive tool with 3D and multi-level cell memory modeling capability," *J. Low Power Electron. Appl.*, vol. 7, no. 3, p. 23, Sep. 2017.
- [33] S. H. Jo, T. Chang, I. Ebong, B. B. Bhadviya, P. Mazumder, and W. Lu, "Nanoscale memristor device as synapse in neuromorphic systems," *Nano Lett.*, vol. 10, no. 4, pp. 1297–1301, Apr. 2010.
- [34] Y. Ho, G. M. Huang, and P. Li, "Dynamical properties and design analysis for nonvolatile memristor memories," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 58, no. 4, pp. 724–736, Apr. 2011.
- [35] A. Syed, E. Ahmed, D. Maksimovic, and E. Alarcon, "Digital pulse width modulator architectures," in *Proc. IEEE 35th Annu. Power Electron. Spec. Conf.*, vol. 6, Jun. 2004, pp. 4689–4695.
- [36] Synopsys. (2010). *Design Compiler User Guide*. [Online]. Available: http://acsweb.ucsd.edu/~coz004/DC_user_guide.pdf
- [37] S.-S. Sheu et al., "A 4Mb embedded SLC resistive-RAM macro with 7.2 ns read-write random-access time and 160 ns MLC-access capability," in *Proc. IEEE Int. Solid-State Circuits Conf.*, vol. 1, Feb. 2011, pp. 200–202.
- [38] M. Mao, Y. Cao, S. Yu, and C. Chakrabarti, "Optimizing latency, energy, and reliability of 1T1R ReRAM through cross-layer techniques," *IEEE J. Emerg. Sel. Topics Circuits Syst.*, vol. 6, no. 3, pp. 352–363, Sep. 2016.
- [39] R. J. Evans and P. D. Franzon, "Energy consumption modeling and optimization for SRAM's," *IEEE J. Solid-State Circuits*, vol. 30, no. 5, pp. 571–579, May 1995.
- [40] R. B. Jacobs-Gedrim, D. R. Hughart, S. Agarwal, G. Vizkelethy, E. S. Bielejec, B. L. Vaandrager, S. E. Swanson, K. E. Knisely, J. L. Taggart, H. J. Barnaby, and M. J. Marinella, "Training a neural network on analog TaOx ReRAM devices irradiated with heavy ions: Effects on classification accuracy demonstrated with CrossSim," *IEEE Trans. Nucl. Sci.*, vol. 66, no. 1, pp. 54–60, Jan. 2019.
- [41] S. Agarwal, S. J. Plimpton, I. Richter, A. H. Hsia, and D. R. Hughart, Sandia National Laboratories. *Crosssim: Crossbar Simulator*. Accessed: May 21, 2020. [Online]. Available: <http://cross-sim.sandia.gov>
- [42] F.-C. Chiu, P.-W. Li, and W.-Y. Chang, "Reliability characteristics and conduction mechanisms in resistive switching memory devices using ZnO thin films," *Nanos. Res. Lett.*, vol. 7, no. 1, p. 178, Dec. 2012.
- [43] U. Chand, C.-Y. Huang, J.-H. Jieng, W.-Y. Jang, C.-H. Lin, and T.-Y. Tseng, "Suppression of endurance degradation by utilizing oxygen plasma treatment in HfO₂ resistive switching memory," *Appl. Phys. Lett.*, vol. 106, no. 15, Apr. 2015, Art. no. 153502.
- [44] A. Prakash, J. Park, J. Song, J. Woo, E.-J. Cha, and H. Hwang, "Demonstration of low power 3-bit multilevel cell characteristics in a TaO_x-based RRAM by stack engineering," *IEEE Electron Device Lett.*, vol. 36, no. 1, pp. 32–34, Jan. 2015.
- [45] C. Hsu, I. Wang, C. Lo, M. Chiang, W. Jang, C. Lin, and T. Hou, "Self-rectifying bipolar TaO_x/TiO₂ RRAM with superior endurance over 10¹² cycles for 3D high-density storage-class memory," in *Proc. Symp. VLSI Technol.*, 2013, pp. T166–T167.
- [46] H. Jiang, L. Han, P. Lin, Z. Wang, M. H. Jang, Q. Wu, M. Barnell, J. J. Yang, H. L. Xin, and Q. Xia, "Sub-10 nm ta channel responsible for superior performance of a HfO₂ memristor," *Sci. Rep.*, vol. 6, no. 1, p. 28525, Jun. 2016.
- [47] F. Zahoor, T. Z. A. Zulkifli, and F. A. Khanday, "Resistive random access memory (RRAM): An overview of materials, switching mechanism, performance, multilevel cell (MLC) storage, modeling, and applications," *Nanos. Res. Lett.*, vol. 15, no. 1, p. 90, Dec. 2020.



YOUNGSUN HAN (Member, IEEE) received the B.E. and Ph.D. degrees in electrical and computer engineering from Korea University, Seoul, South Korea, in 2003 and 2009, respectively. From 2009 to 2011, he was a Senior Engineer with System LSI, Samsung Electronics, Suwon, South Korea. From 2011 to 2019, he was an Assistant/Associate Professor with the Department of Electronic Engineering, Kyungil University, Gyeongsan, South Korea. Since 2019, he has been an Associate Professor with the Department of Computer Engineering, Pukyong National University, Pusan, South Korea. His research interests include high-performance computing, emerging memory systems, compiler construction, and SoC design.



DONGMIN KIM is currently pursuing the B.E. degree with the Department of Computer Engineering, Pukyong National University, Pusan, South Korea. His research interests include high-performance computing, memory systems, and quantum computing, particularly, quantum machine learning.



YONGTAE KIM (Member, IEEE) received the B.S. and M.S. degrees in electrical engineering from Korea University, Seoul, South Korea, in 2007 and 2009, respectively, and the Ph.D. degree from the Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX, USA, in 2013. From 2013 to 2018, he was a Software Engineer with Intel Corporation, Santa Clara, CA, USA. Since 2018, he has been with the School of Computer Science and Engineering, Kyungpook National University, Daegu, South Korea, where he is currently an Assistant Professor. His research interests include energy efficient integrated circuits and systems, particularly, neuromorphic computing and approximate computing, and new memory devices and architectures.

...