ELectronics EXpress

LETTER

IEICE Electronics Express, Vol.14, No.11, 1–8

Enhancing lifetime of phase-change memory for video processor

Wooyoung Jang[a)]

Department of Electronics and Electrical Engineering, Dankook University, 152, Jukjeon-ro, Yongin-si, Gyeonggi-do, 16890, Korea

a) *wyjang@dankook.ac.kr*

Abstract: Phase-change memories (PCMs) provide the benefits of non-volatility and capacity, but they show low performance, high power consumption, and low endurance for write operations. Such PCM drawbacks are exacerbated in state-of-the-art video processors that code images block-by-block. In this letter, we propose a PCM subsystem that removes unnecessary write operations resulting from the block-by-block accesses of video processors. Experimental results show that our PCM subsystem improves the lifetime and performance of PCMs up to 716 times and 6.8 times, respectively, on average. Moreover, it makes PCMs consume 69.7 times lower power than the conventional PCM subsystem, on average.

Keywords: phase-change memory, video processor, system-on-chip

Classification: Integrated circuits

References

[1] M. Wuttig and N. Yamada: "Phase-change materials for rewriteable data storage," Nat. Mater. **6** (2007) 824 (DOI: 10.1038/nmat2009).

[2] B. C. Lee, *et al.*: "Architecting phase change memory as a scalable DRAM alternative," ACM SIGARCH Computer Architecture News **37** (2009) 2 (DOI: 10.1145/1555815.1555758).

[3] H. Yoon, *et al.*: "Efficient data mapping and buffering techniques for multilevel cell phase-change memories," ACM Trans. Archit. Code Optim. **11** (2015) 40 (DOI: 10.1145/2669365).

[4] Y. Joo, *et al.*: "Energy-and endurance-aware design of phase change memory caches," Design, Automation & Test in Europe Conference & Exhibition (2010) 136.

[5] J. Watkinson: *The MPEG Handbook: MPEG-1, MPEG-2, MPEG-4* (Taylor & Francis, 2004) 2nd ed. 204.

[6] T. Wiegand, *et al.*: "Overview of the H.264/AVC video coding standard," IEEE Trans. Circuits Syst. Video Technol. **13** (2003) 688 (DOI: 10.1109/TCSVT.2003.815168).

[7] G. J. Sullivan, *et al.*: "Overview of the high efficiency video coding (HEVC) standard," IEEE Trans. Circuits Syst. Video Technol. **22** (2012) 1649 (DOI: 10.1109/TCSVT.2012.2221191).

[8] W. Jang and D. Z. Pan: "An SDRAM-aware router for networks-on-chip," IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst. **29** (2010) 1572 (DOI: 10.1109/TCAD.2010.2061251).

[9] W. Jang and D. Z. Pan: "Application-aware NoC design for efficient SDRAM

DOI: 10.1587/elex.14.20170402
Received April 19, 2017
Accepted May 15, 2017
Publicized May 26, 2017
Copyedited June 10, 2017

accesses," IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst. **30** (2011) 1521 (DOI: 10.1109/TCAD.2011.2160176).
[10] Open core protocol specification (2013) http://www.accellera.org.

1 Introduction

As dynamic random access memories (DRAMs) are approaching the limit of process scaling, reliability, and power consumption, phase-change memories (PCMs) have recently attracted considerable attention [1]. PCMs that have a non-volatile storage mechanism amenable to process scaling with high reliability are more advanced than other non-volatile memories in practical use. Therefore, PCMs have been considered as a scalable DRAM alternative [2]. Fig. 1 shows a PCM cell and its operations. The PCM cell employs the large resistivity difference between the top electrode and the heater by the phase-change material. Let an electrical current pass through the phase-change material in Fig. 1(a). The set and reset state of a PCM cell refers to the crystalline phase and the amorphous phase, respectively. For resetting a PCM cell to the amorphous phase, an access transistor injects a large electrical current pulse into the PCM cell, and thermally induces a high-resistance state. First, a programming region is melted, and then quenched rapidly for a short time period. Next, the region of the amorphous and highly resistive material is formed. For setting a PCM cell to the crystalline phase, an access transistor injects a medium electrical current pulse into the PCM cell, and thermally induces a low-resistance state. A programming region is annealed between melting temperature and crystallization for a sufficiently long period of time to crystallize. For reading the phase of a programming region, the resistance of a PCM cell is measured by passing an electrical current small enough not to disturb the current phase. The schematic pulse shapes for writing and reading operations are summarized in Fig. 1(b).

Such a PCM write operation requires a large electrical current, and the resulting thermal stress reduces the endurance of billions of write operations per PCM cell. Because of this limitation, it may be difficult to position PCMs mainly as a DRAM

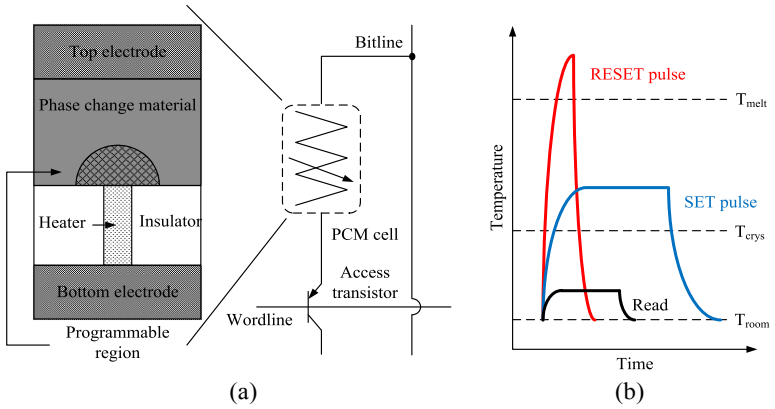

Fig. 1. PCM and its operation [2].
(a) PCM cell.
(b) Current pulse shapes for writing and reading operations.

DOI: 10.1587/elex.14.20170402
Received April 19, 2017
Accepted May 15, 2017
Publicized May 26, 2017
Copyedited June 10, 2017

replacement. Previous works have improved the latency, power consumption, and finite endurance of PCMs for general-purpose processors [2, 3, 4], but they are not effective for video processors performing the latest block-based image coding algorithms [5, 6, 7]. Therefore, PCM subsystems should consider not only general-purpose processors, but also video processors.

2 PCM write endurance in video processors

The latest video processors code images via moving picture expert group 1/2/4 [5], H.264/advanced video coding [6], and H.265/high efficiency video coding (HEVC) [7] standards that utilize block-based schemes. Now that such video processors store an image block in multiple rows of a PCM-based main memory, multiple row activation and deactivation operations are performed for writing a single image block. During the deactivation operation in a selected bank, data in the row buffer of a PCM-based main memory are written in all PCM cells on a selected row even though they all are not new. Whenever an image block is stored, thus, all PCM cells on multiple rows related to the block are accessed. As a result, the PCM cells are rapidly worn out, show low performance, and consume high power.

Fig. 2 shows one of the conventional image-to-memory mapping techniques that store ultra-high definition (UHD) 4 K images in a PCM-based main memory. The column, bank, row, rank, and channel addresses for accessing the main memory are selected from given memory address a[35:0] as shown in Fig. 2(a). In the figure, $a[x{:}y, z]$ represents bits x-y and z of memory address. The main memory that is interconnected to the conventional PCM subsystem via a 64-bit data bus consists of four banks, each bank includes 65,536 rows, and each row has 32,768 PCM cells. For ease of explanation, we assume that a single image line is one-to-one mapped to a single row of any bank as shown in Fig. 2(b). Let 16×16 block A be written in the PCM-based main memory. First, each bank is activated with the first row. That is, data stored in all PCM cells on the row are copied to a row buffer. Sublines 0A-3A are written in the row buffer of banks 0–3, respectively. Next, sublines 4A-7A should be written in the second row of banks 0–3, respectively. However, since each bank is currently activated with the first row, each bank is deactivated. While each bank is deactivated, all data in its row buffer are written in PCM cells on the first row even though the data all are not new. Then, each bank is activated with the second row, and sublines 4A-7A are written in the row buffer of banks 0–3, respectively. In order to write the rest of block A, banks 0–3 are deactivated, and thus, all data in their row buffer are written in PCM cells on their second row even though the data all are not new. In the same manner, sublines 8A-11A and 12A-15A are written in the third and fourth row of each bank, respectively. Next, let block B be written in the PCM-based main memory. Each bank is again activated with the first row, and sublines 0B-3B are written in the row buffer of banks 0–3, respectively. Then, each bank is deactivated since sublines 4B-7B should be written in the second row of banks 0–3, respectively. While banks 0–3 are deactivated, all data in their row buffer are written in PCM cells on their first row even though they all are not new. In the same manner as sublines 0B-3B, sublines 4B-7B, 8B-11B and 12B-15B are written in the second, third, and fourth

DOI: 10.1587/elex.14.20170402
Received April 19, 2017
Accepted May 15, 2017
Publicized May 26, 2017
Copyedited June 10, 2017

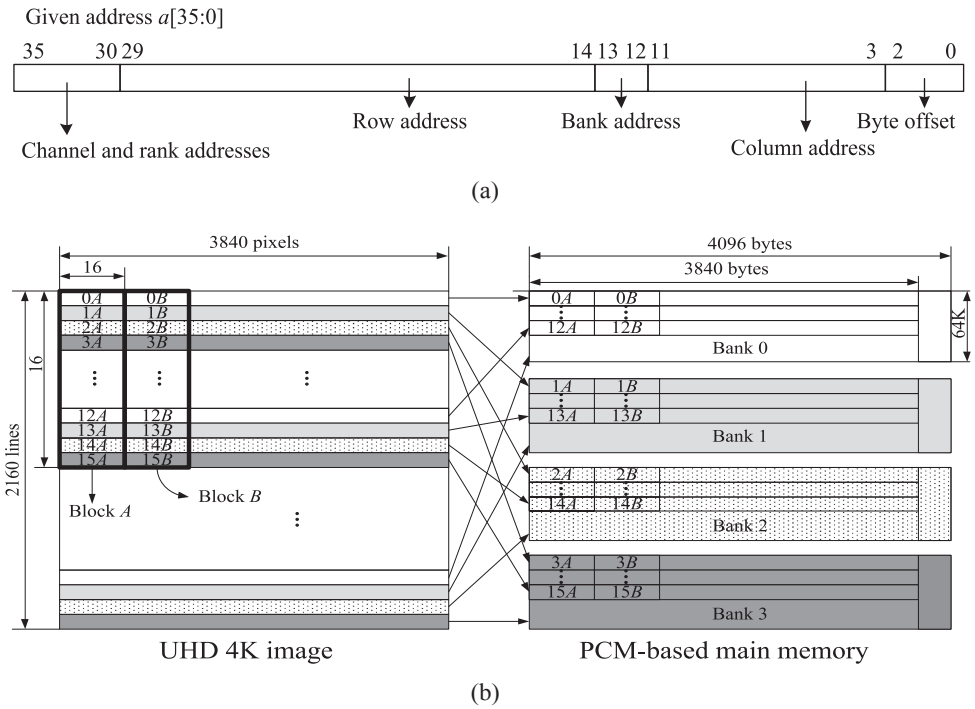

Fig. 2. Direct image-to-PCM mapping technique.
(a) Generating column, bank, and row addresses.
(b) Graphical representation of direct image-to-PCM mapping technique.

row of each bank, respectively. Such block-by-block write operations are repeated until the whole image is stored. As a result, whenever the conventional PCM subsystem applying a direct image-to-PCM mapping technique stores a block, all PCM cells on rows related to the block are accessed.

Such block-by-block write operations greatly reduce the lifetime of PCMs. The reason is that all PCM cells on a row must be updated whenever few PCM cells on the row are updated. In this example, each PCM cell is updated 960 times per image frame. In the case that frame rate is 30 Hz and a video processor has four frame buffers, each PCM cell is updated about 207-million times per day. If the mean time to failure (MTTF) of a PCM cell is 10^9 write operations [2], thus, the PCM-based main memory will be worn out within 4.8 days.

3 Enhancing PCM lifetime for video processors

In the above example, the PCM subsystem should extend the PCM lifetime by up to 227 times to guarantee a video processor for three years. Our main idea for extending the PCM lifetime involves storing the image block in a single row unlike the conventional PCM subsystem storing the image block in multiple rows. The technique minimizes the number of deactivation operations where all data in a row buffer are written in PCM cells on a row. Thus, each PCM cell can be updated once per image frame, and thus, the PCM lifetime is enhanced by up to 3.2 years. In addition, the performance of block-by-block write and read operations can be greatly improved as all sublines in an image block are accessed under the row-buffer hit condition [8]. On the contrary, our technique may be disadvantageous for

DOI: 10.1587/elex.14.20170402
Received April 19, 2017
Accepted May 15, 2017
Publicized May 26, 2017
Copyedited June 10, 2017

line-by-line read operations demanded by a display processor. Thus, we will present effective line-by-line read operations mitigating the disadvantages.

Fig. 3 shows the proposed image-to-PCM mapping technique storing UHD 4 K images in the PCM-based main memory. The column, bank, row, rank, and channel addresses for accessing the main memory are selected from given memory address a[35:0] as shown in Fig. 3(a). Whereas the bank and column addresses used in the conventional PCM subsystem are a[13:12] and a[11:3], respectively, the bank and column addresses used in the proposed PCM subsystem are a[5:4] and a[13:6,3], respectively. As shown in Fig. 3(b), such a mapping technique makes each block stored in the same row. Let the proposed PCM subsystem start storing blocks A and B. The proposed PCM subsystem activates the first row of bank 0, and then, stores all sublines of block A in the row buffer of bank 0. Next, all sublines of block B are sequentially stored in the row buffer of bank 0. Since such write operations have the row-buffer hit relation between sublines or blocks, additional deactivation and activation operations are not required [8]. When the row buffer of bank 0 is fully written with successive blocks, bank 0 is deactivated. While bank 0 is deactivated, all data in its row buffer are written to PCM cells in the first row of bank 0. Now that the data in the row buffer are all new, all PCM cells on the first row can be updated at the same time in the proposed PCM subsystem. Next, the first row of bank 1 is activated, and the next blocks are stored in the row buffer of bank 1 sequentially. Such block-by-block write operations are repeated until the whole image is stored. As a result, the proposed PCM subsystem writes data in a PCM cell once per image frame, and thus, PCM cells of which the MTTF is 10^9 write operations can run by up to 3.2 years.

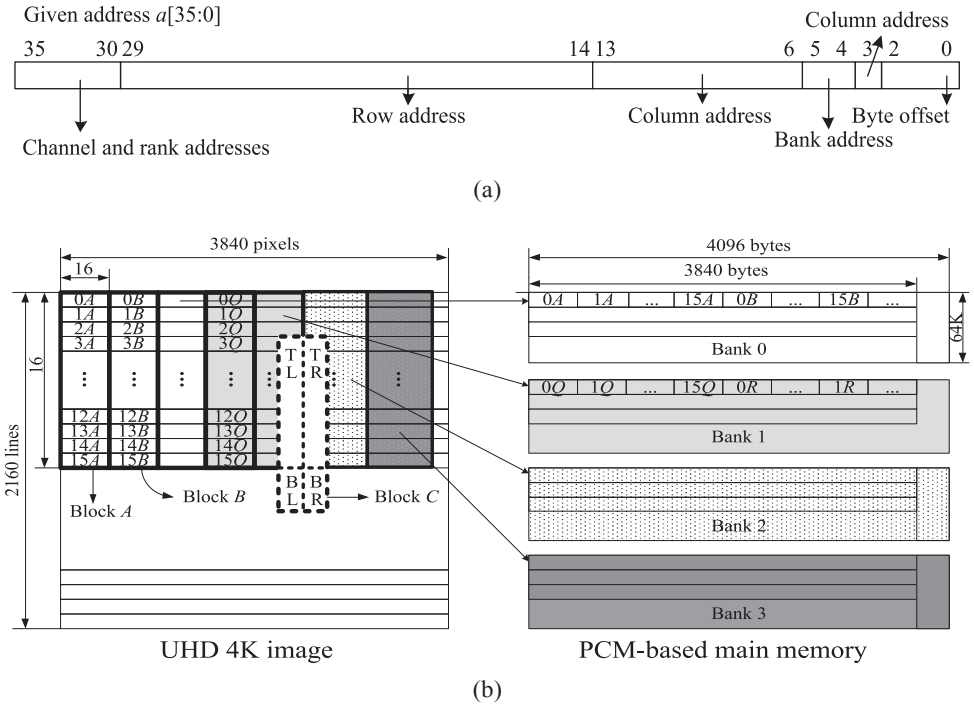

Fig. 3. Proposed image-to-PCM mapping technique.
(a) Generating column, bank, and row addresses.
(b) Graphical representation of proposed image-to-PCM mapping technique.

DOI: 10.1587/elex.14.20170402
Received April 19, 2017
Accepted May 15, 2017
Publicized May 26, 2017
Copyedited June 10, 2017

Images stored in the proposed manner are used by display and video processors. Since a display processor reads images line-by-line, but not block-by-block, the read operations of the proposed PCM subsystem can be complicated. This is because the proposed PCM subsystem should read a single image line from multiple rows and banks. In addition, the proposed PCM subsystem can show low memory efficiency in the case that it does not fully utilize data that are read. For example, the first request for reading line 0 makes the main memory output sublines, 0*A*, 1*A*, 2*A*, and 3*A*. Subline 0*A* is displayed, but sublines 1*A*, 2*A*, and 3*A* may be discarded since they are not displayed immediately. However, most display processors equip a line buffer for enhancing the quality of images and scaling images to a screen. If the line buffer stores more than three lines in this example, sublines not displayed immediately can be stored, and then used after line 0 is completely displayed. Thus, the proposed PCM subsystem does not cause any read performance losses for the display processor.

Most video processors not only write images to the PCM-based main memory block-by-block, but also read the images block-by-block for various applications such as image prediction, block matching, object recognition/detection, etc. However, such video applications can read blocks that are not aligned with blocks stored in the PCM-based main memory. For example, block *C* in Fig. 3(b) lies across four blocks, and thus it is not aligned with any blocks stored. In order to read block *C*, the conventional PCM subsystem may must perform, at the maximum, activation and deactivation operations twice as many as the number of a block line, and thus it causes high power consumption, and low memory latency. On the contrary, the proposed PCM subsystem divides block *C* into top-left (TL), bottom-left (BL), top-right (TR), and bottom-right (BR) subblocks, based on the boundary of blocks stored. The subblocks are separately read in the order of TL, TR, BL, and BR. Since sublines within each subblock are stored in the same row and bank, they can be read under the row-buffer hit condition. In addition, since subblocks are not stored each other in the different row and the same bank, read operations between the subblocks have a high possibility of being accessed under the row-buffer hit condition or the bank interleaving condition [8]. In the case that the size of blocks is 16×16 and PCMs output 64-byte data per read command, the conventional PCM subsystem, and the proposed PCM subsystem should perform, at the maximum, 32 read operations, and 10 read operations, respectively. Therefore, the proposed PCM subsystem achieves higher PCM performance and lower PCM power consumption than the conventional PCM subsystem.

4 Experimental results

Our approach is evaluated with an UHD television (TV) model that is modified from high-definition television (HDTV) model [9] implemented in Verilog hardware description language (HDL). A video processor decodes frames in an I, P, B, B, P, B, and B order via H.265/HEVC standard [7], and a display processor displays the decoded frames with a 30 Hz frame rate in an I, B, B, P, B, B, and P order. I, P, and B frames represent intra-coded, predicted, and bi-predicted frames, respectively. Our model includes several memory request generators that demand

DOI: 10.1587/elex.14.20170402
Received April 19, 2017
Accepted May 15, 2017
Publicized May 26, 2017
Copyedited June 10, 2017

Table I. PCM simulation parameters [2]

Delay and timing (ns)		
Parameter	Description	Time
t_{RCD}	The delay between PCM read and row buffer read/write commands	55
t_{CL}	The delay between a row buffer read command and the start of row buffer read transaction	12.5
t_{WL}	The delay between a row buffer write command and the start of row buffer write transaction	10
t_{CCD}	The delay between row buffer read/write commands	10
t_{WTR}	The delay between the end of row buffer write transaction and a row buffer read command	7.5
t_{WR}	The delay between row buffer write and PCM write commands	15
t_{RTP}	The delay between row buffer read and PCM write commands	7.5
t_{RP}	The delay between PCM write and following PCM read commands	150
t_{RRDact}	The delay between PCM read commands	5
t_{RRDpre}	The delay between PCM write commands	27.5

Energy (pJ/bit)				
PCM Read	PCM Write	Row Buffer Read	Row Buffer Write	Background Power
2.47	16.82	0.93	1.02	0.08

Table II. PCM lifetime comparison

Image	Row size of PCM	The number of frame buffer	Lifetime by CPS	Lifetime by PPS
FHD 1920	1-KB	4	19.3 days	3.2 years
	2-KB	5	14.4 days	4.7 years
UHD 4 K	2-KB	4	9.6 days	3.2 years
	4-KB	5	7.2 days	4.7 years
UHD 8 K	4-KB	4	4.8 days	3.2 years
	8-KB	5	3.6 days	4.7 years

memory services instead of the video processor and the display processor. The memory request generators are interconnected to a PCM subsystem via an on-chip interconnection network using an open core protocol [10]. In addition, we model a 4 GB PCM-based main memory with four banks in Verilog HDL. The major PCM simulation parameters used in our experiment are shown in Table I.

Table II shows the lifetime of PCMs accessed by the conventional PCM subsystem (called CPS), and the proposed PCM subsystem (called PPS), when the MTTF of the PCMs is 10^9 write operations. The lifetime of PCMs accessed by the CPS depends on the size of an image, the row size of a PCM, and the number of a frame buffer. On the contrary, the lifetime of PCMs accessed by the PPS depends only on the number of a frame buffer. The reason is that our PPS updates PCM cells once per frame, whereas the CPS updates PCMs cells once per block. As a result, our PPS can run PCMs up to 716 times longer than the CPS, on average.

DOI: 10.1587/elex.14.20170402
Received April 19, 2017
Accepted May 15, 2017
Publicized May 26, 2017
Copyedited June 10, 2017

Figs. 4(a) and 4(b) show the performance and power consumption of the PPS, respectively, normalized to those of the CPS. Our PPS performs 27.5 times and 2.6 times faster block-by-block write and read operations, respectively, than the CPS. On the contrary, it hardly degrades the performance of line-by-line read operations for a display processor. Therefore, the PPS achieves 6.8 times shorter decoding and displaying time than the CPS, on average. In addition, our PPS makes the PCM-based main memory consume 69.7 times lower power than the CPS, on average. The PPS reduces the PCM power consumption of block-by-block write and read operations up to 114.5 times and 7.4 times, respectively, compared to the CPS. Since the PPS stores the sublines of block in the same row and bank, row activation and deactivation operations critical to power consumption can be greatly minimized. On the contrary, the PPS performs line-by-line read operations with power consumption similar to the CPS.

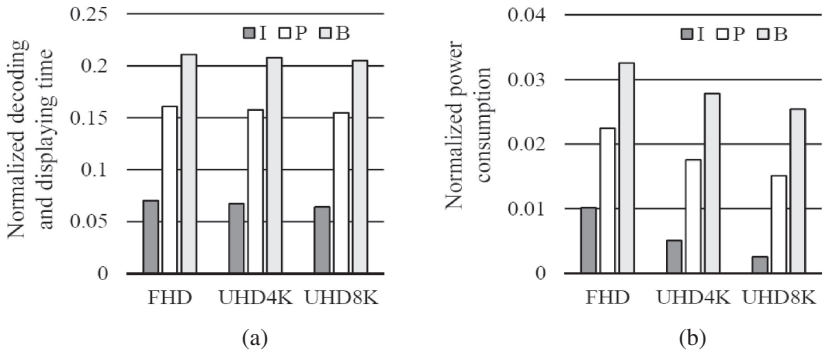

Fig. 4. Experimental results on UHD TV model.
(a) Decoding and display time of PPS normalized by those of CPS.
(b) Power consumption of PPS normalized by those of CPS.

5 Conclusion

In previous PCM-related works, most researches were carried out on general-purpose processors. However, the latest system-on-chip includes a variety of processors that exacerbate the PCM disadvantages. In particular, a video processor performing block-based algorithms rapidly increases meaningless PCM write operations. The proposed PCM subsystem reduces such unnecessary PCM write operations for the video processor, and thus, greatly enhances the lifetime, performance, and power consumption of PCMs. In conclusion, the proposed approach provides promising opportunities for video processors with the PCM-based main memory.

Acknowledgments

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning (2014R1A1A1002262).

DOI: 10.1587/elex.14.20170402
Received April 19, 2017
Accepted May 15, 2017
Publicized May 26, 2017
Copyedited June 10, 2017