

A Hybrid Display Frame Buffer Architecture for Energy Efficient Display Subsystems

Kyungtae Han, Alexander W. Min, Nithyananda S. Jeganathan and Paul S. Diefenbaugh
Circuits and Systems Research, Intel Labs
2111 N.E. 25th Avenue, Hillsboro, OR 97124
Email: {kyungtae.han, alexander.w.min, nithyananda.s.jeganathan, paul.s.diefenbaugh}@intel.com

Abstract—Our principal motivation is to reduce the energy consumption of display subsystems in mobile devices by introducing a *hybrid* frame buffer architecture into the platform. We observed that display contents on a screen are quite static for certain mobile workloads, such as web browsing. As a result, data reading from the display frame is much more frequent than the writing of new data onto the frame buffer, a state we refer to as *read dominance*. Based on this observation, we propose a hybrid frame buffer architecture that exploits the display contents' read-dominant property to improve the energy efficiency of display subsystems. Specifically, we employ two memory types: DRAM and Phase-Change Memory (PCM), in the display frame buffer to exploit their different read/write energy characteristics. We also present an analysis of the energy efficiency of the hybrid frame buffer based on our display content and energy consumption models. Our evaluation results show that the proposed hybrid frame buffer reduces frame buffer energy consumption by up to 43%, compared to the conventional DRAM-only frame buffer.

I. INTRODUCTION

Display subsystems are power hungry components and their energy efficient design is critical for battery operated mobile devices, such as smartphones and tablets. The display subsystem delivers computer output to humans through visual media, e.g., display panels such as LCD or LED, typically consuming more than 30% (including panel and processing logics) of the total mobile device power [6]. Even worse, the display subsystem is expected to consume more energy as the size and resolution of display panels continue to increase, generating a larger amount of data access to/from the frame buffer. Therefore, the energy efficient design of display subsystems remains an important problem.

Depending on the contents to be displayed, the read access to the frame buffer can be much more frequent than the write access, which we refer to as *read dominance*. For instance, in an office application (e.g., text editor) a large portion of the displayed image may remain static for a long period of time. In this case, the contents in the frame buffer may not be updated frequently, but the display engine will nevertheless periodically scan/read the same image from the frame buffer to project them onto the display. Because of the constant read from the frame buffer, a memory that consumes less power on read access than on write access will improve the energy efficiency of the display subsystem, especially for static contents, such as web browsing, e-mail and messaging applications.

While there exist many techniques for improving energy efficiency of display subsystems [6], [8], [13], [14], [23], none of them have considered the read-dominant property of the frame buffer. For example, Shim *et al.* [23] introduced a compressed frame buffer to reduce the amount of data delivered to the panel. Choi *et al.* [8] developed a dynamic-colour-depth-based

TABLE I
DRAM AND PCM ENERGY [7]

	DRAM	PCM
Read energy (pJ/bit)	4.4	2.47
Write energy (pJ/bit)	5.5	14.03 (set) 19.73 (reset)

frame buffer access scheme at the expense of image quality. Hollevoet *et al.* [13] proposed to add an extra frame buffer on the screen to decrease the frequency of memory access. Kim *et al.* [14] devised a scheme that dynamically adjusts the display refresh rate to minimize display power consumption. However, our scheme is unique in that it exploits the asymmetric frame buffer access patterns.

In this paper, we propose a hybrid frame buffer architecture that significantly improves energy efficiency of display subsystems by exploiting the read-dominant property of display contents. There are several papers that propose a hybrid memory using DRAM and Phase-Change Memory (PCM) [15]. Dhiman *et al.* [10] proposed a hybrid main memory, which exposes DRAM and PCM addressabilities to the OS. Qureshi *et al.* [20] adopted PCM and used DRAM as a conventional cache to PCM. Liu *et al.* [17] proposed a hybrid memory for DSP systems. To demonstrate the benefits of the hybrid frame buffer, we employ two memory types, DRAM and PCM, to store images in the frame buffer. DRAM consumes relatively less power on write than PCM, whereas certain PCM is more energy efficient on read than DRAM [5], [7], as shown in Table I. In the hybrid frame buffer, we exploit such heterogenous power characteristics of DRAM and PCM for read-intensive, low-activity contents, such as web browsing and office applications. Our evaluation results show that the hybrid frame buffer with DRAM/PCM reduces power by up to 43% compared to the conventional DRAM-only frame buffer. We also evaluate the suitability of PCM as a hybrid frame buffer because PCM is known to have certain limitations, e.g., lifetime and speed [11]. We discuss several potential solutions to mitigate these practical constraints. To the best of our knowledge, our work is the first to propose hybrid frame buffer architecture to improve the display subsystem energy efficiency.

This paper is organized as follows. Section II reviews the display subsystem, display contents and PCM. Section III introduces the hybrid frame buffer architecture with PCM. Section IV analyzes the benefits of the proposed architecture compared to the DRAM-based system. Section V evaluates the performance of the proposed hybrid frame buffer archi-

texture for various workloads, and discusses more advanced techniques for display subsystems.

II. BACKGROUND

In this section, we briefly describe the operation of display subsystems, study the static behavior of display contents, and then introduce PCM as a potential technology for hybrid frame buffers.

A. Display Subsystem

Figure 1 illustrates a conventional display subsystem, which consists of the application processor, display frame buffer, and display panel. The application processor generates frame images and stores them in the display frame buffer. The display frame buffer is associated with a display panel, such as LCD or LED display, which reads the frame image from the frame buffer to display it on the screen. In a conventional display frame buffer, there is a single (homogeneous) type of memory, e.g., DRAM, to write and read an image.

B. Computer Display Contents

Typically computer-generated display contents have large static regions, e.g., a desktop background image does not change frequently without user interaction or notification. To study the static behavior of display contents, we have developed a custom display capture board with FPGA and studied the characteristics of the display contents. The display capture board reports the percentage of pixels changed in each frame by continuously storing display frames and comparing their pixels with those of the previous frame. Table II shows the statistics for typical mobile workloads and benchmarks, including 3D animation and MobileMark2007 [1]. The results show that the display contents seldom change (e.g., < 13%) for most of the tested workloads. This makes frequent periodic reads from the frame buffer for screen refresh unnecessary, and significantly wasteful of energy. Therefore, we can conclude that certain mobile workloads are read-dominant in displaying their contents, which motivates us to propose the hybrid frame buffer architecture that exploits this asymmetry in read/write energy consumption.

C. Phase-Change Memory (PCM)

Of the several energy efficient memory and storage techniques recently developed [9], PCM is known as one of the most energy efficient, non-volatile memories [5]. One interesting feature of PCM is its *asymmetric* power consumption characteristics—i.e., write energy is much higher than read energy—, as shown in Table I. This asymmetry in read/write power consumption is mainly due to the high power required for changing its structure for write, regardless of the set/reset of the state. This power asymmetry is quite different

TABLE II
DISPLAY CONTENTS STATISTICS

	Changed pixel (%)	Duration (sec)
3D animation	12.73	656
Web suite	0.30	99
MobileMark2007	0.08	7244
SysMark productivity	0.84	555
Random mouse move	0.06	48

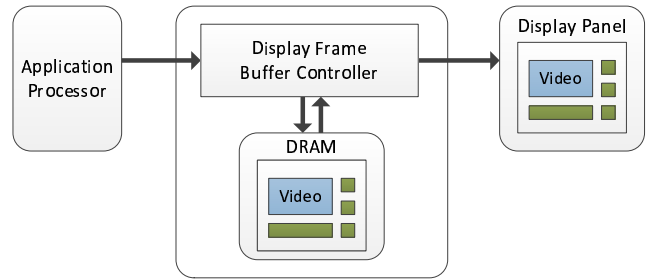


Fig. 1. Conventional display subsystem.

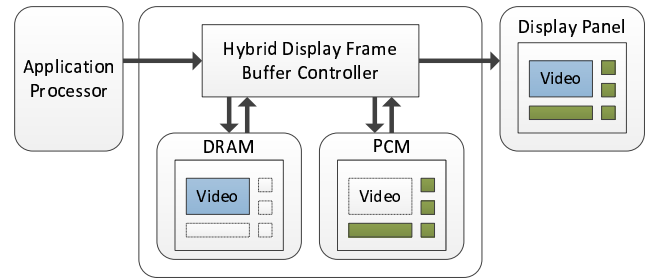


Fig. 2. Data flow in hybrid approach: PCM stores static contents and DRAM stores dynamic contents.

from conventional memory types, e.g., DRAM, in which read and write power consumption is similar. Therefore, PCM’s read-power-efficient property makes it a good candidate for applications in read-dominated scenarios. Table I shows the DRAM and PCM energy numbers that we use in this paper.

III. THE PROPOSED ARCHITECTURE

As we mentioned, display contents are quite static for many mobile workloads, and thus the contents can be categorized into high and low activity areas. For example, while streaming a YouTube video from the Internet, the video playback area can be considered a high activity area as it continuously displays new data, and the rest of the screen can be considered a low activity area.

Based on this practical observation, we present the proposed hybrid frame buffer architecture that improves the energy efficiency of the display subsystem by employing heterogeneous memory types with different read/write power characteristics, e.g., DRAM and PCM. For instance, in the hybrid frame buffer, the high activity area (i.e., video content) is stored in a write-efficient memory (e.g., DRAM), whereas the static area is stored in a read-efficient memory (e.g., PCM). This hybrid architecture is quite different from the traditional architecture with a homogeneous frame buffer, in which the read/write power consumption is comparable (e.g., DRAM as shown in Table I).

A. Hybrid Frame Buffer Architecture

Figure 2 shows the proposed display subsystem consisting of five components: (i) an application processor, (ii) a DRAM, (iii) a PCM, (iv) a hybrid display frame buffer controller, and (v) a display panel. The hybrid architecture introduces PCM to store read-dominant contents, and introduces a hybrid display frame buffer controller that identifies the high/low-activity areas. Once it identifies or predicts the content characteristics,

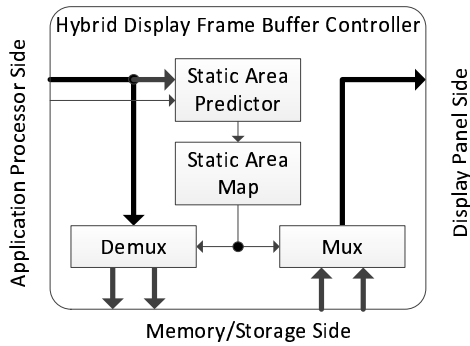


Fig. 3. Data flow in hybrid frame buffer controller.

it stores the contents for high-activity regions to the DRAM, and the remaining low-activity regions to the PCM. Next, we will elaborate on the detailed operation of the hybrid frame buffer controller.

Figure 3 illustrates the data flow in the hybrid display frame buffer controller that consists of four main components: (i) a static area predictor, (ii) a static area map, (iii) a Demux, and (iv) a Mux. Initially, the controller stores the display contents to DRAM, similar to a conventional display controller. For example, the application processor stores images in the DRAM frame buffer through Demux, and the display panel reads them from DRAM through Mux, as shown in Figure 3. Meanwhile, it monitors the content characteristics based on recent history, e.g., update frequency. Once it identifies a static (read-dominant) region of current contents, the controller updates the static area map and directs the updated contents for the static region to the PCM for energy efficient read operations. The static area information in the static region map will be used by the Demux to split and distribute the updated images to the PCM/DRAM. When the display panel requests a frame image, the Mux combines the images according to the static area maps.

B. Display Content Prediction

To maximize energy savings, it is important for the static region predictor to accurately identify the static regions of the displayed images. While the prediction of static/dynamic region is outside the scope of this paper, we describe several practical methods that can be used to identify content characteristics, including OS-guided monitoring and pixel-by-pixel comparison, as we describe below.¹

First, in the OS guided monitoring method, the OS provides the changed area information in the current frame, and the static area predictor can use this information to predict the static area for future frames. One example of the OS guided monitoring is XDamage extension software [2] that provides the dynamic region in a frame buffer. This feature is widely used for remote display applications, e.g., X11VNC [3], to reduce the amount of data to be transmitted to a remote display by sending only images in changed areas. Once the OS guided monitoring provides the dynamic region to the static area predictor, the detector keeps track of the dynamic region to calculate the inactive region. In general, the OS

¹Note that we assume the content prediction can be done with high accuracy throughout the paper.

TABLE III
GENERAL TERMS

Terms	Descriptions
N	Total number of coherent content frames
f	Number of changed frames in high-activity area
d	Portion of dynamic contents in a frame
s	Portion of static contents in a frame
B	Number of bits to represent a frame
E_{DRAM}^w	Write energy per bit in DRAM
E_{DRAM}^r	Read energy per bit in DRAM
E_{PCM}^w	Write energy per bit in PCM
E_{PCM}^r	Read energy per bit in PCM

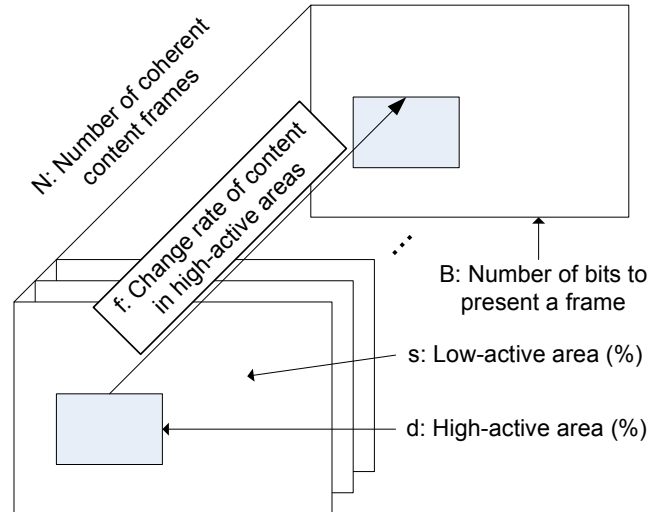


Fig. 4. Display contents model.

guided methods have negligible impact on CPU utilization. For example, one of the OS guided detection methods called Dynamic Image Detection Scheme (DIDS) [24] shows only 1% increase in CPU utilization.

Another approach is a pixel-by-pixel comparison, which compares each pixel in the current frame to one in the previous frame. This approach provides the greatest accuracy and finest granularity of the static region detection. However, computation overhead for the pixel comparison grows as the screen resolution increases.

IV. ANALYSIS

In this section, we first introduce the display content memory and frame buffer energy models, and then we analyze the energy, life time and speed of the hybrid frame buffer architecture to demonstrate its benefits. Table III defines the terms that we use in our analysis.

A. Display Content Model

The amount of read and write access depends on the characteristics of the displayed contents. For example, high motion contents incur a relatively large amount of write to the frame buffer, whereas office applications (e.g., text editor or email) incur relatively low write activity. Fig. 4 illustrates the display content model where we identify the key parameters that quantify the content characteristics: (i) the number of changed frames in high-activity areas, f , within given coherent content N frames, and (ii) the portion of content changed d .

TABLE IV
AN EXAMPLE OF CONTENT TYPES WITH $N = 60$ FRAMES PER SECOND

Region \ Frame	Frame	
	Low (0-30 frames)	High (30-60 frames)
Low	Web browsing	Cursor movement
High	Video playback	3D game

For example, $f = 60$ and $N = 60$ means that every frame changes its context, and thus new data will be written to the frame buffer for each frame update. With these key parameters (i.e., f , N and d), display contents can be classified into four categories, as shown in Table IV.

B. Memory Energy Model

We employ a simplified version of the DRAM energy model in [19] in order to highlight the impact of read and write access characteristics on energy consumption. The DRAM energy model consists of read, write, and background energy from different power modes, including refresh, idle, and retention modes [19]. Typically, the background energy varies depending on memory size, power management policy and workloads [18]. In order to understand the energy consumption behavior with different read and write rates, we use the following DRAM energy model:

$$E_{DRAM}(x, y) = x \cdot E_{DRAM}^w + y \cdot E_{DRAM}^r \quad (1)$$

where x and y are the amounts of write and read bits, respectively, and E_{DRAM}^w and E_{DRAM}^r are DRAM's write and read energy per bit, respectively.

We consider PCM for read-energy-efficiency memory, and use the PCM energy model introduced in [22]. The energy consumption of PCM can be expressed as:

$$E_{PCM}(x, y) = x \cdot \frac{E_{PCM}^{preset} + E_{PCM}^{set}}{2} + y \cdot E_{PCM}^r, \quad (2)$$

where E_{PCM}^{preset} , E_{PCM}^{set} and E_{PCM}^r denote PCM's energy consumption for preset, set, and read, respectively.

Note that in Eq. (2), to simplify the analysis, we assume that the probability of write for one (set) or zero (preset) in a bit is equiprobable within an image. We focus on the energy of DRAM and PCM, and also assume that the energy overhead introduced by the hybrid controller is negligible. However, our energy model is generic and can easily be modified to accommodate different ratios between set and reset operations in the PCM.

C. Frame Buffer Energy Model

In the conventional frame buffer with homogeneous memory, the frame buffer energy consumption is mainly determined by the amount of write and read activity. First, the energy spent on write activity consists of dynamic and static parts, i.e., $(d \cdot f + s) \cdot B$, where f is the number of changed frames in high-activity areas, B denotes the number of bits to represent a frame, and d and s denote the portion of dynamic and static contents in a frame, respectively. Second, the total energy spent on read activity, which transfers the data of N frames in the frame buffer to a panel, is $N \cdot B$. The write activity is the function of the number of changed frames, f . However the read activity is not a function of total frames since the display subsystem consumes energy continuously on read, and

TABLE V
WRITE AND READ AMOUNTS IN HOMOGENEOUS AND HYBRID FRAME BUFFERS

		Write	Read
Homogeneous		$(d \cdot f + s) \cdot B$	$N \cdot B$
Hybrid	DRAM	$d \cdot f \cdot B$	$d \cdot N \cdot B$
	PCM	$s \cdot B$	$s \cdot N \cdot B$

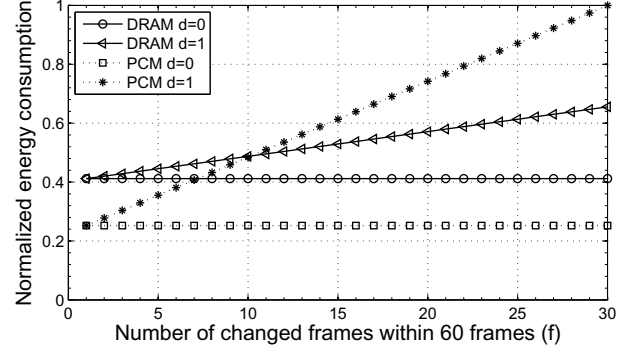


Fig. 5. Energy comparison between DRAM and PCM frame buffers.

the panel refreshes every frame regardless of changes in the displayed contents.

In the hybrid frame buffer, DRAM stores pixels in the foreground (dynamic) contents, whereas PCM stores pixels in the background (static) contents. The number of pixels to write the foreground contents to DRAM and read is $d \cdot f \cdot B$ and $d \cdot N \cdot B$, respectively. The changed region is re-written only for f frames, but it is read every N frames. The number of pixels to write the background content to PCM and read is $s \cdot B$, and $s \cdot N \cdot B$, respectively. Background is written one time, while it is read every N frames. Table V summarizes the model of write and read amounts for homogeneous (i.e., DRAM) and hybrid frame buffers (i.e., DRAM+PCM).

D. Numerical Evaluation

Here we first evaluate the energy consumption characteristics of DRAM and PCM, and identify the display content scenarios under which PCM outperforms DRAM in terms of energy efficiency. We then evaluate the potential energy saving benefits of our hybrid approach.

Fig. 5 compares the normalized energy consumption of the DRAM-only and PCM-only frame buffers for two extreme cases, i.e., $d = 0$ and 1. The figure shows that PCM is more energy efficient for relatively static contents (i.e., small f), whereas DRAM is more energy efficient for dynamic contents (i.e., large f). The existence and locations of the cross points are determined mainly by the two parameters, d and f , and the crossing point, if it exists, can be calculated from Eqs. (1) and (2), and Table V. For a given d , the cross point of f is:

$$f = \frac{(E_{PCM}^w \cdot s - E_{DRAM}^w) + (E_{PCM}^r - E_{DRAM}^r) \cdot N}{(E_{DRAM}^w - E_{PCM}^w) \cdot d}. \quad (3)$$

Figure 6 plots the cross points derived from Eq. (3), and the curve represents the break-even points between DRAM-only and PCM-only frame buffers in terms of energy consumption. DRAM outperforms PCM above the line and vice versa. The low active display contents, such as web browsing and office

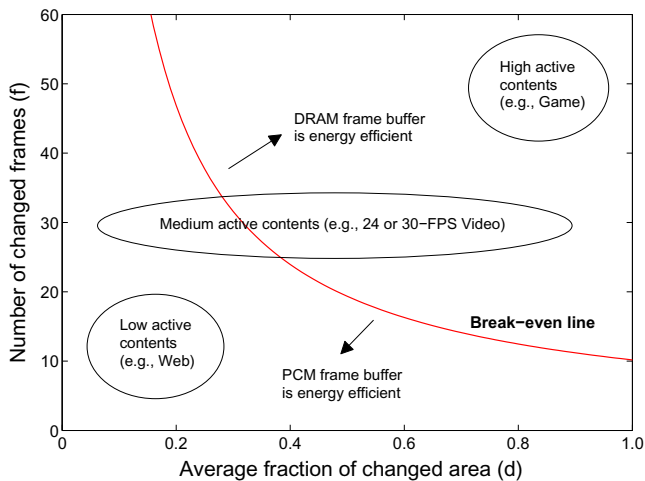


Fig. 6. Break-even line between DRAM and PCM frame buffers.

applications, benefit from the PCM frame buffer, whereas the high activity display contents, such as video games, benefit from the use of DRAM over PCM. For instance, video playback applications, which normally have 24 or 30 frames per second, benefit depending on the size of the changed area d . Contents of small screen changes with a high number of frames per second, such as YouTube video streaming, would benefit from the use of PCM. The DRAM frame buffer would be better for full screen video playback.

V. EVALUATION

In this section, we demonstrate the efficacy of the proposed hybrid frame buffer architecture via in-depth evaluations.

A. Evaluation Setup

We evaluate the efficacy of the hybrid frame buffer by comparing the energy consumption of the following three testing schemes: (i) DRAM-only, (ii) PCM-only, and (iii) Hybrid frame buffer with DRAM and PCM. We evaluate the schemes for three different levels of granularities, i.e., Pixel (P), Macroblock (MB), and Scanline (SL), in identifying the static/dynamic region of the display contents, and writing/reading them to/from the frame buffers. This also defines the minimum unit for content mapping information in the Static Area Map in the Hybrid frame buffer controller (see Figure 3). The pixel level is the finest granularity to separate and combine the frame buffer. Macroblock represents a tile of 16-by-16 pixels [21]. A scan line is a single line in the frame, which is usually a horizontal line in the display.

We evaluate the schemes for the scenarios listed in Table VI, based on the experiments conducted in [12]. In the evaluation, we use the DRAM and PCM read/write power consumption in Table I.

B. Energy Benefit

Figure 7 compares the energy consumption of the three tested frame buffer methods. The figure shows that the proposed hybrid frame buffer outperforms the homogeneous, i.e., DRAM- and PCM-only, schemes for all the tested scenarios, thanks to its ability to exploit the read-dominant property of the displayed contents and leverage the asymmetry in

TABLE VI
SCENARIO CONFIGURATION (f_{ps} : FRAMES PER SECOND, PORTION OF CHANGED PIXELS IN A CHANGED FRAME)

	fps	SL (%)	MB (%)	P (%)
MobileMark2007	5.4	5.73	1.96	0.92
3D animation (VLC)	24.51	41.09	33.25	21.11
3D animation (QT)	39.1	41.33	33.59	19.54
SysMark 3D	14.28	4.41	0.31	0.17
SysMark E-Learning	15.24	4.89	1.11	0.67
SysMark productivity	14.13	17.61	7.67	3.55
SysMark video creation	12.07	9.44	3.66	2.65
YouTube	30.07	32	17.02	9.24
Web suite	12.4	6.81	2.78	1.44
Random mouse move	55.98	4.13	0.22	0.06

read/write power consumption in DRAM and PCM. The results show that the hybrid frame buffer saves up to 43% of the energy compared to the conventional DRAM-only frame buffer; in general, the lower the portion of active contents, the greater the power savings.

One interesting observation is that as the content writing methods become less sophisticated (e.g., SL), average energy consumption increases due to the increased energy overhead in writing the contents to the frame buffer. For highly active contents, e.g., 3D animation with QuickTime player, the energy consumption of the PCM-only frame buffer becomes even higher than that of the DRAM-only frame buffer because of the high write-power consumption of the PCM.

C. Discussion

In this section, we evaluate the feasibility of using PCM for display frame buffers from other aspects, such as memory access speed and memory lifetime expectations. We also discuss several potential solution approaches and techniques that can mitigate such constraints.

Speed: While PCM provides a great energy saving benefit for read-intensive display contents, it is relatively slow compared to DRAM. For example, write bandwidth of DRAM is in the order of GB/s per die, whereas PCM achieves 50-100MB/s per die [16]. This might not be sufficient to support FullHD ($1920 \times 1080 \times 24 \text{ bpp} \times 60 \text{ fps}$), which requires 373 MByte/s for bandwidth. However, this bandwidth problem can be overcome via a parallel use of PCM. For example, Micron recently announced a PCM with bandwidth of 400 MByte/s [4], which can provide bandwidth for servicing FullHD display contents.

Lifetime: It is known that PCM has a short lifetime expectancy with at most 10^7 writes per bit cell before failure. However, the PCM lifetime can be extended significantly with known techniques, such as *wear leveling*, *read-before-write*, and *frame buffer compression*. The wear leveling technique spreads PCM cell use over the available cells, ideally equalizing the use of all PCM cells. Figure 8 shows an example of expected lifetime of 2 GB PCM with the wear leveling technique for various applications and usages. For example, PCM can last more than 10 and 100 years for video and web browsing applications, respectively.

VI. CONCLUSION

Energy efficiency is a critical design consideration for mobile devices such as smartphones and tablets. In this

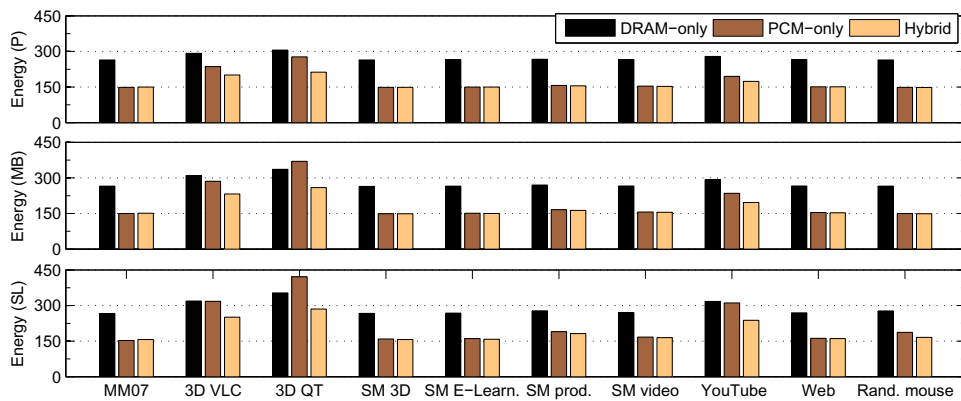


Fig. 7. Energy comparison among DRAM-only, PCM-only and hybrid frame buffers.

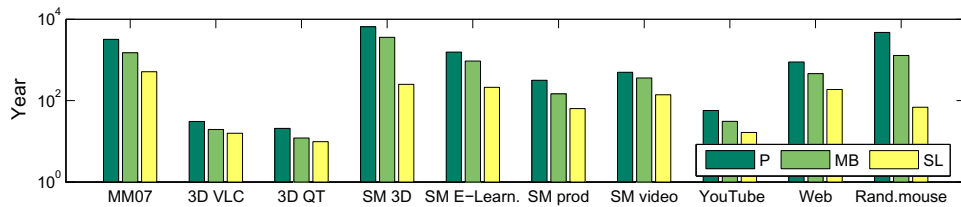


Fig. 8. Lifetime comparison among Pixel (P), Macroblock (MB), and Scanline (SL) map with wear leveling techniques in 2 GB PCM.

paper, we proposed a novel hybrid frame buffer architecture for display subsystems that improve energy efficiency by employing heterogeneous frame buffers with asymmetric read/write power characteristics. Specifically, we (i) observed that certain mobile display contents exhibit a read-intensive property, (ii) proposed a novel hybrid architecture for display subsystem frame buffers to take full advantage of the read-intensive access pattern of display contents, (iii) compared the read/write energy consumption behavior of DRAM and PCM, (iv) characterized the scenarios under which PCM outperforms DRAM thanks to its ability to exploit the read dominance of the displayed contents, and (v) comparatively evaluated the efficacy of the proposed hybrid frame buffer architecture. Our evaluation results showed that the proposed hybrid frame buffer architecture can reduce the energy consumption of display subsystems by up to 43 % for highly active contents.

REFERENCES

- [1] BAPCO, MobileMark 2007, <http://www.bapco.com/products/mobilemark2007/>.
- [2] Keith Packard, Software/XDamage, <http://freedesktop.org/wiki/Software/XDamage>.
- [3] x11vnc: a VNC server for real X displays, <http://www.karlrunde.com/x11vnc/>.
- [4] Micron first to market with phase-change memory modules for portable devices, <http://www.engadget.com/2012/07/18/micron-first-phase-change-memory/>.
- [5] R. Bheda, J. Poovey, J. Beu, and T. Conte. Energy efficient Phase Change Memory based main memory for future high performance systems. In *IGCC*, July 2011.
- [6] A. Bhowmik and R. Brennan. System-Level Display Power Reduction Technologies for Portable Computing and Communications Devices. In *IEEE Portable*, May 2007.
- [7] J. Chen, R. C. Chiang, H. H. Huang, and G. Venkataramani. Energy-aware writes to non-volatile main memory. In *ACM HotPower*, October 2011.
- [8] I. Choi, H. Shim, and N. Chang. Low-power color TFT LCD display for hand-held embedded systems. In *ACM ISLPED*, August 2002.
- [9] N. Derhacopian, S. Hollmer, N. Gilbert, and M. Kozicki. Power and Energy Perspectives of Nonvolatile Memory Technologies. *Proceedings of the IEEE*, 98(2):283–298, February 2010.
- [10] G. Dhiman, R. Ayoub, and T. Rosing. PDRAM: A hybrid PRAM and DRAM main memory system. In *DAC*, July 2009.
- [11] A. P. Ferreira, M. Zhou, S. Bock, B. Childers, R. Melhem, and D. Mossé. Increasing PCM main memory lifetime. In *ACM DATE*, March 2010.
- [12] K. Han, N. S. Jeganathan, and P. S. Diefenbaugh. Parametric Evaluation of Computer Displays for Optimized Display Data Transmission. *Symposium Digest of Technical Papers*, 42(1), June 2011.
- [13] L. Hollevoet, A. Dewilde, K. Denolf, F. Catthoor, and F. Louagie. A power optimized display memory organization for handheld user terminals. In *ACM DATE*, February 2004.
- [14] H. Kim, H. Cha, and R. Ha. Dynamic refresh-rate scaling via frame buffer monitoring for power-aware LCD management. *Softw. Pract. Exper.*, 37(2):193–206, February 2007.
- [15] B. Lee, P. Zhou, J. Yang, Y. Zhang, B. Zhao, E. Ipek, O. Mutlu, and D. Burger. Phase-Change Technology and the Future of Main Memory. *IEEE Micro*, 30(1):131–141, January/February 2010.
- [16] B. C. Lee, E. Ipek, O. Mutlu, and D. Burger. Phase change memory architecture and the quest for scalability. *Commun. ACM*, 53(7):99–106, July 2010.
- [17] T. Liu, Y. Zhao, C. Xue, and M. Li. Power-aware variable partitioning for DSPs with hybrid PRAM and DRAM main memory. In *DAC*, June 2011.
- [18] Micron. TN-47-04: Calculating Memory System Power for DDR2 Introduction. Technical report, Micron Technology, 2005.
- [19] V. Moshnyaga, H. Vo, G. Reinman, and M. Potkonjak. Handheld System Energy Reduction by OS-Driven Refresh. In *Integrated Circuit and System Design. Power and Timing Modeling, Optimization and Simulation*, volume 4148, pages 24–35. 2006.
- [20] M. K. Qureshi, V. Srinivasan, and J. A. Rivers. Scalable high performance main memory system using phase-change memory technology. In *ACM/IEEE ISCA*, June 2009.
- [21] I. Richardson. *H.264 and MPEG-4 Video Compression*. John Wiley and Sons, 2004.
- [22] Z. Shao, Y. Liu, Y. Chen, and T. Li. Utilizing PCM for Energy Optimization in Embedded Systems. In *IEEE ISVLSI*, August 2012.
- [23] H. Shim, N. Chang, and M. Pedram. A compressed frame buffer to reduce display power consumption in mobile systems. In *ASP-DAC*, January 2004.
- [24] K.-J. Tan, J.-W. Gong, B.-T. Wu, D.-C. Chang, H.-Y. Li, Y.-M. Hsiao, Y.-C. Chen, S.-W. Lo, Y.-S. Chu, and J.-I. Guo. A remote thin client system for real time multimedia streaming over VNC. In *IEEE ICME*, July 2010.