A Partial Memory Protection Scheme for Higher Effective Yield of Embedded Memory for Video Data

Kang Yi[1], Shih-Yang Cheng[2], Fadi Kurdahi[2], and Ahmed Eltawil[2]

[1] School of Computer Sci. and Electrical Eng., Handong Global University, Pohang, Korea
yk@handong.edu
[2]Department of EECS, University of California, Irvine, CA 92697-265
{shihyanc, kurdahi, aeltawil}@uci.edu

Abstract

With the emerging SoC era the on-chip embedded memory will occupy most of the silicon real estate. As the technology proceeds into very deep submicron, the yield of SoCs will drop sharply mainly because of the on-chip memory failure. Therefore, the embedded memory is becoming the crucial part for achieving higher chip yield. In this paper, we propose an error-resilient video data memory system architecture design. The proposed scheme employs partial memory protection scheme rather than traditional whole memory protection. Our approach is based on the fact that video data memory need not to be error-free because multimedia data has built-in redundancies by their own nature and allows partial data loss without serious quality degradation. With our approach we can achieve 100% data memory yield while incurring a small power overhead. We demonstrate the efficiency of our approach with H.264 application up to 2.0% memory bit error.

1. Introduction

The recent trend in the microelectronic system design is to integrate as many IPs as possible into a single silicon die. Because most popular applications usually require large size data memory arrays, it makes sense to consider embedding that memory on-chip. Multimedia applications are perfect examples that can benefit from having large embedded memory on-chip. Integrating large data memory has several benefits:

(1) it lowers power consumption
(2) it lower cost because the system parts count is reduced
(3) smaller PCB size with a smaller number of parts
(4) higher performance due to the elimination of pad-to-pad delays
(5) more reliable operation due to single packaging

It is reported most of SOC will be moving from computation-bound domain to memory-bound domain. It is expected to be 94% of total area of a chip will be occupied by memory arrays by year 2014 [1] as shown in Figure 1.

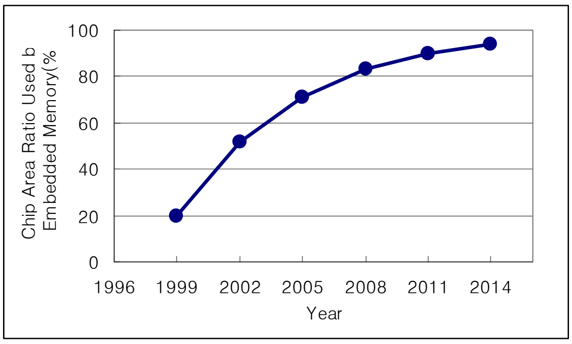


Figure 1. The increasing ratio of embedded memory area in a single chip

However, integrating embedded memory blocks into a chip results in a very low manufacturing yield because: (1) the die area is significantly increased and (2) memories are laid out using aggressive design rules allowing a denser packing than random logic, this making memory cells more susceptible to failure due to manufacturing defects. What makes things worse is that as the process technology goes into very deep submicron, the dominant reason of permanent defects is no longer topology changes as is the case in current technolgies. Instead, the random placement of chemical dopants so called RDF (Random Dopant Fluctuation) will be the dominant source of permanent defects

978-1-4244-2683-6/08/$25.00 ©2008 IEEE

during manufacturing process. These random fluctuations lead to inter-circuit transistor mismatches that can have detrimental effects on performance. Furthermore, these effects are strongly dependent on the operating conditions (voltage, frequency, temperature etc.) [2].

In order to combat the increasingly detrimental impact of RDF and similar process variations on memories, we have to devise for recovering data from defective memory cells after chip fabrication in order to achieve a reasonable effective memory yield.

There are a multitude of techniques that have been extensively studied in literature to counteract memory failures, such as redundant rows/columns technique and the use of error correction codes (ECC). The redundant row/column techniques require a lot of area overhead under the error rates expected in the current technology. According to [3] the area overhead will drastically increase for the next generation nano-scale devices. An alternative approach is ECC which is employed in memory architectures to correct for transient faults (soft errors). This dynamic technique can be adapted to the parameter variation change due to RDF. However, the drawback is that most ECC systems can correct only a single error (without significant overhead). Therefore, ECC is not effective in handling the high volume of errors expected in nanometer scale designs [3,4].

Traditionally, system designers assume that the underlying memory system is 100% defect-free. Based on this assumption the existing memory recovery algorithms for permanent defect were developed. Existing memory repair techniques are becoming less effective with rising defects, especially RDF and other process variations. Thus, we cannot have reasonable yield if we require 100% perfect memory. Therefore, 100% error free requirement for on-chip memory is becoming unrealistic and impractical especially for the near future nano-scale systems.

In our previous work [5,6,7], the authors proposed a new paradigm on the memory defect to handle the memory yield problem with the nano-scale technology. Instead of searching ways to make error-free memory architecture, our paradigm allows memories to have errors and try to find the way to live with errors by mitigating the impacts of error as much as possible without too much cost overhead.

In this paper we propose a new paradigm in video applications by partial memory protection. We protect the embedded SRAM data memory blocks with more important data while allowing errors in the memory blocks with less important data. Video frame data in multimedia applications like H.264 is error resilient by nature. Also, in order to achieve high compression rates, video data is usually compressed by lossy algorithms rather than lossless algorithm, which means video data is very error-resilient. Based on this observation we develop a partial memory protection scheme for building error-resilient memory system as means of achieving higher effective yield.

This paper is organized as follows. Section 2 describes the idea and implementation of our partial memory protection technique with H.264. Section 3 demonstrates our idea with video data example. Section 4 concludes our work and suggests future directions.

2. Partial Memory Protection Scheme for H.264 Video Decoder Application

2.1. Embedded Memory of H.264 Video Decoder System

We choose the H.264 decoder as an example application to apply our idea because it is one of the most promising applications nowadays.

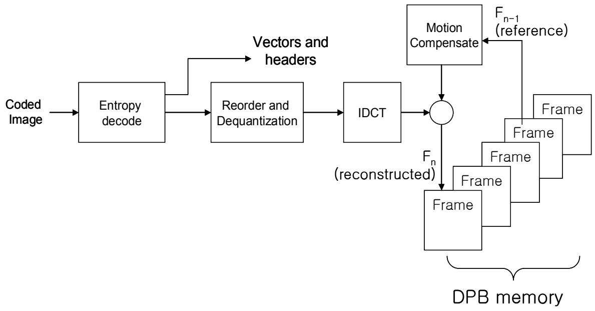


Figure 2. H.264 Video Decoder System Overview

Figure 2 shows the H.264 decoder system block diagram. The steps involved in decoding each frame are: entropy decoding, dequantization, inverse DCT, and motion compensation (for B and P slices). The DPB (Decoded Picture Buffer) in the Figure 2 is the storage for previously decoded picture image. In the motion compensation step the next frame is derived by reading the pixel data from the reference frame stored in the DPB. If the target application of this decoder is HDTV with 1920 x 1080 frame size, the DPB (Decoded Picture Buffer) memory need to be as large as 24Mb/frame. Since H.264 compression for HDTV systems usually needs 2 to 5 frame buffers, it requires at least 48Mbits memory. As mentioned in Section 1, there are many pressures from markets to integrate the large memory blocks with the core compression engine block into a single chip. Actually, it is reported embedded memory saves power dissipation by 22%

compared to a multi chip solution with external memory arrays [7]. However, as mentioned before, the large on-chip memory makes the die very susceptible to manufacturing defects resulting in lower chip yield.

2.2. Partial Protection Memory Overview

Our proposed scheme is to protect the frame buffer memory partially rather than trying to make the whole memory array as 100% error-free. Regarding each byte as a pixel data the MSB bits should contain more significant value of the pixel information (Chroma or Luma component) as shown in Figure 3. Since the MSB parts carry more important data compared to LSB parts for every pixel data in an integer representation, we protect the MSB bits for every pixel word. If we keep *n* MSB bits we can expect less distortion of visual image or more acceptable image quality. Generally, the more redundancy there is in the data, the smaller *n* needs to be.

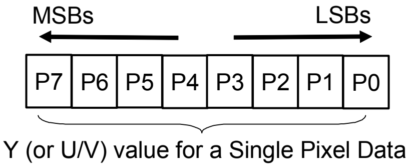


Figure 3. . MSBs and LSBs in a data word

We build the memory architecture as shown in Figure 4. With this partial memory protection scheme we combine protected bits and unprotected bits to get a pixel data for each DPB read operation. Here, we consider how many bits should be protected and how to protect memory bits. The choice of *n* may depend on the bit error rates in memory and the required image quality.

2.3. Memory Protection Method

Note that the most memory errors at near future will be from parameter fluctuations in quantities such as MOS device threshold. The random dopant fluctuation (RDF) has resulted in SRAM circuits which are topologically correct, but that fail to meet performance metrics. In other words, memory cell failure occurs because of different access time for read or write operation that is resulted from different cell-to-cell threshold values. According to our previous work [5] with Berkeley 70nm model, the memory error rates and memory supply voltage has the relation at a range of access time condition (delay) as shown in Figure 5 and Figure 6 .

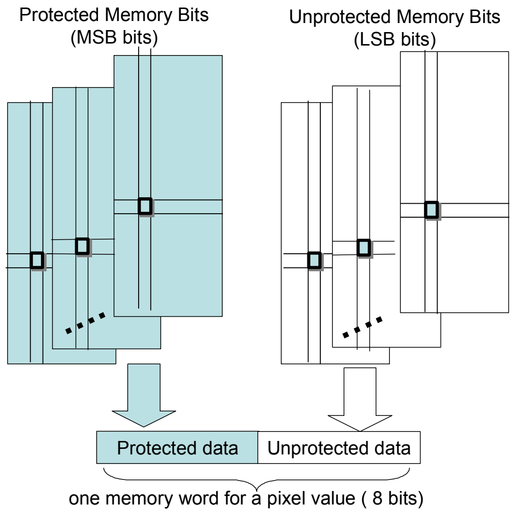


Figure 4. Our Partial Memory Protection Scheme

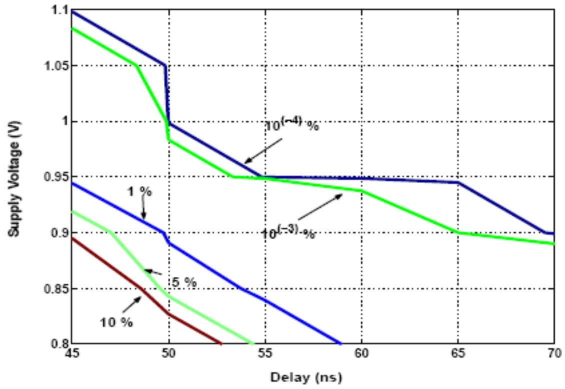


Figure 5. Equi-probable contour for Error Rates of Memory Write Operation

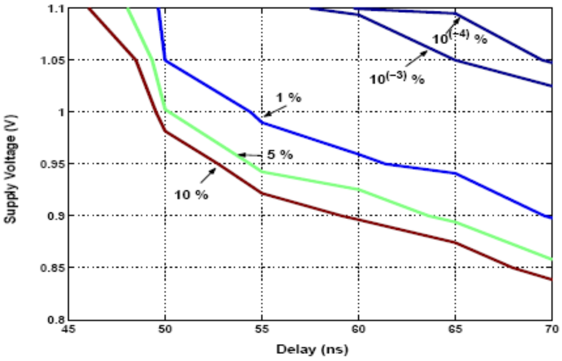


Figure 6. Equi-probable Contours for Error Rates of Read Operation

The contour lines in the Figure 5 and Figure 6 depict the following cell failure probabilities: 10^{-4}, 10^{-3}, 1, 5, and 10%. From the graphs in Figure 6, it becomes clear

clear that read operations are more susceptible to failures under the lower memory supply voltage. If memory operates at 0.95 volt the error rate of memory is about 1% with the condition of delay 65 ps. While if the same memory operates at 1.1 volt the error rate of memory is about 10^{-4} %.

At the low memory error rates like 10^{-4} % the video quality is the same as that of uncorrupted video image. It means that if we use higher supply voltage (e.g. 1.1 volt) we can protect memory from operation failure at the cost of more power consumption. Therefore, We assume that our partial protection memory architecture has dual power rails as shown in Figure 7.

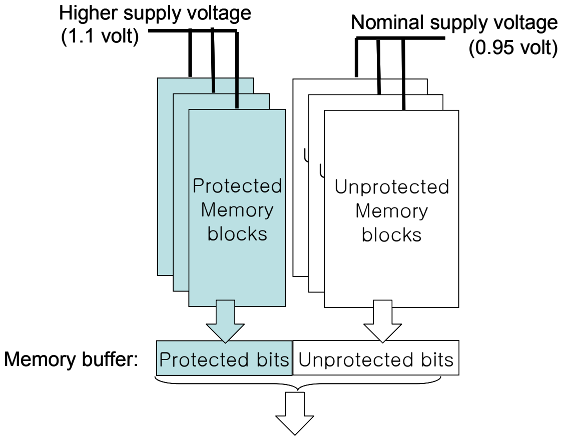


Figure 7. Our Memory Protection Scheme with dual power rails

Given the above, two questions can be posed:

(1) What is the least number of bits to be protected in order to get proper quality of video at different bit error rates of memory?

(2) What are the trade offs between the number of bits protected and the image quality at different bit error rates of memory?

In the following Section we experimentally attempt to answer those two questions.

3. Experimental Results

We performed experiments with software simulation to figure out the proper memory protection scheme for acceptable video quality at a range of memory error rates. We modified a public domain source code of an H.264 decoder to inject errors into the DPB video memory. When injecting errors, we protect a part of memory to test our partial protection scheme. A video stream named "foreman" is used for experiments which is encoded with the following parameter values: (1) QP = 28, (2) one I frame every 30 frames, (3) two reference frames and (4) Picture sequence type = IPBPBPB… IPBPBPB…

In the simulation we changed the bit error rates as follows: 0.01%, 0.1%, 0.5%, 1.0%, 1.2%, 1.5%, 2.0%. We changed the protected number of bits as 2,3,4, and 5 MSBs. We compared the decoded image quality in terms of PSNR (Peak Signal-to-Noise Ratio) values. The PSNR is a popular metric which compares the output image in sequences with a reference set. PSNR is computed using the following equations:

$$MSE = \frac{\sum [f(i,j) - F(i,j)]^2}{N^2}$$

$$PSNR = 20\log_{10}(\frac{255}{\sqrt{MSE}})$$

Where *f(i,j)* and *F(i,j)* are the pixels at location *i,j* of the output and reference images, respectively.

Figure 8, Figure 9, and Figure 10 show the PSNRs of video sequence with different memory protection schemes. In these figures, the MSB5, MSB4, MSB3, MSB2 represent 5 MSB bits protection, 4 MSB bits protection, 3 MSB bits protection, and 2 MSB bits protection, respectively. "None" in the figure represents zero-bit protection (no memory protection). The quality of Y, U, and V components are shown. As expected, with more bits are protected the higher quality of images we get. For example, with 5 bits protection (MSB5) the image quality is almost the same as the original uncorrupted image up to 2.0% bit error rate of memory. With 4 bit protection (MSB4), the PSNR is slightly degraded at higher bit error rates in memory (BER >1.0%). Also, at lower bit error rates in memory, MSB3 shows similar PSNR compared to that of MSB4 case. And MSB2 seems to be acceptable only at very low bit error rates (BER < 0.001%)

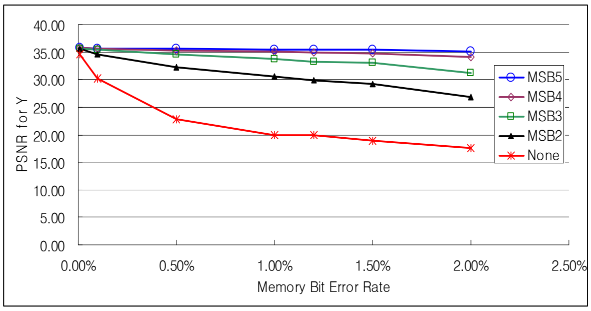


Figure 8. Y PSNR values of different number of protected bits in Video Memory

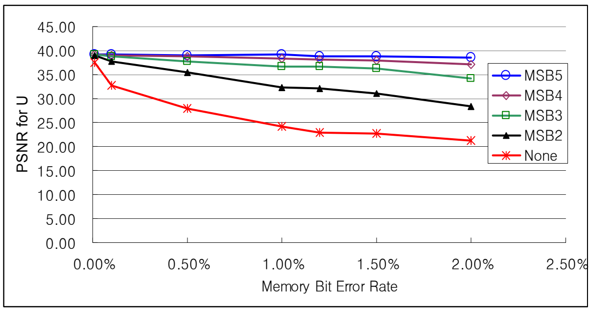


Figure 9. U PSNR values of different number of protected bits in Video Memory

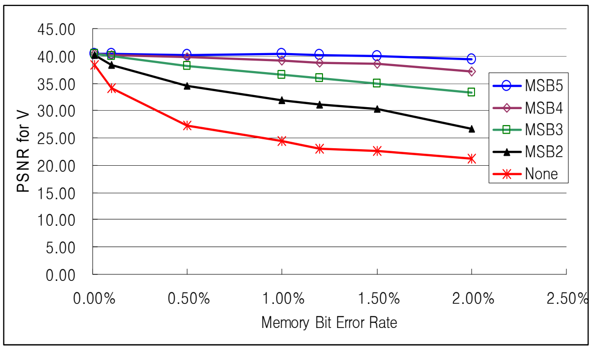
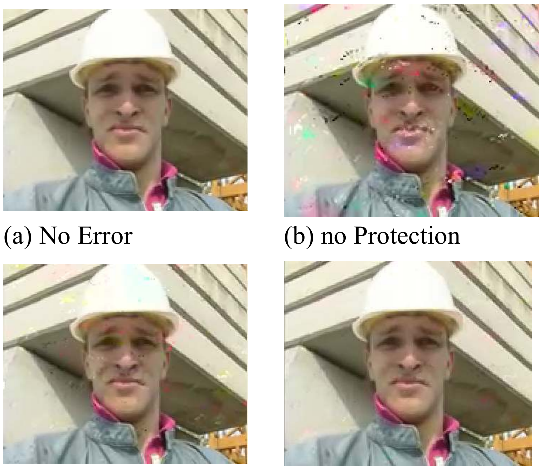


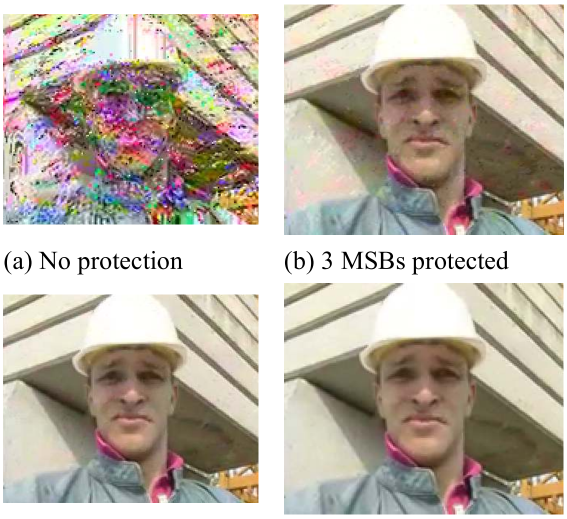
Figure 10. V PSNR values of different number of bits protected in Video Memory

We show the captured images from decoded foreman video at different condition in the Figure 11 for BER=0.1%, Figure 12 for BER=1.0%, Figure 13 for BER=2.0%. From each of the figures, we can find the minimum number of bits protected should be 3, 4, 5 bits for 0.1%, 1.0%, and 2.0% respectively.


(a) No Error (b) no Protection

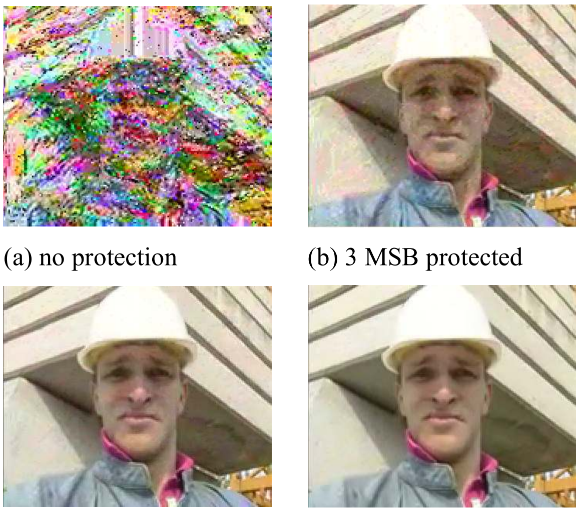
(c) 2 MSBs protected (d) 3 MSB protected

Figure 11. Captured Video Image at BER=0.1%


(a) No protection (b) 3 MSBs protected

(c) 4 MSBs protected (d) 5 MSBs protected

Figure 12. Captured Video Image at BER=1.0%


(a) no protection (b) 3 MSB protected

(c) 4 MSBs protected (d) 5 MSBs protected

Figure 13. Captured Video Image at BER=2.0%

Figure 14 shows the minimum number of bits to be protected in the DPB memory to sustain acceptable visual quality. For BER=0.01% and 0.1% 3, bits protection is enough. But, for BER=0.5%, 1.0%, and 1.2%, 4 bits protection is needed. Furthermore, for BER=1.5% and 2.0% 5 bits protection is needed to have reasonably acceptable visual quality. The overhead saving is the relative overhead saving that may be required for memory protection that is computed by comparing with the full memory word protection scheme. Note that to protect more bits we need to consume more power if we employ the memory protection scheme with higher supply voltage. Therefore, with more protected MSBs, more overhead is required.

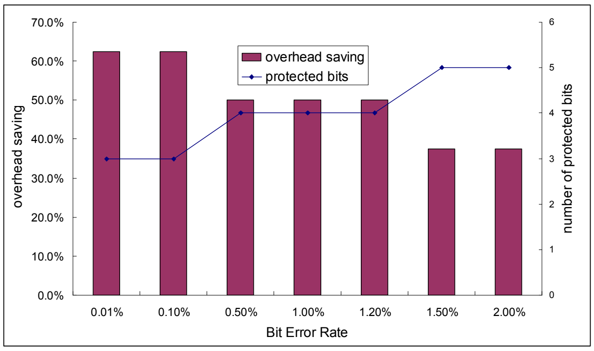


Figure 14. Required Minumum Memory Protection and Overhead at different Memory Error Rates

In Figure 15, we show a different view of the relationship between the number of bits protected and image quality. It implies that the larger the cost we pay, the higher the visual quality that we can expect. If we assume more power is needed to protect more memory bits, we can interpret the X axis in Figure 15 as the power consumption and the graph can be interpreted as the trade-off between power consumption and video quality. Therefore, Quality can be considered as a new variable in the memory design space exploration.

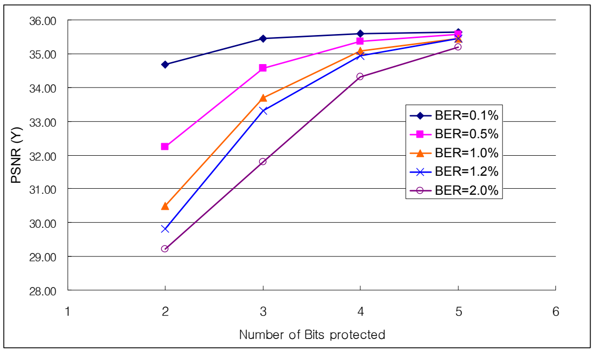


Figure 15. The Relationship between Memory Protection and image Quality

4. Conclusion

Low yield from memory defects due to variability in the manufacturing process is becoming the hurdle to widespread of embedded memory in SoC design. One example of memory-hungry SoCs are multimedia applications which need vast amount of data memory. Since multimedia data has redundancies in themselves, multimedia data like video has error resilience by nature. In this paper, we propose a new technique to enhance the chip yield for SoCs with large embedded multimedia memory by employing a partial memory bit protection technique. Simulation based experiments confirm that this idea is viable and identifies the relationship between quality and the number of protected bits. We conclude that 4 MSB bits protection is reasonable for acceptable video quality at 1.0% memory error rates rather than full memory protection scheme for the near future technology .

5. References

[1] http://public.itrs.net

[2] T. Gupta, A.H. Jayatissa, "Recent advances in nanotechnology: key issues & potential problem areas," Proceedings of IEEE Conference on Nanotechnology, Vol. 2, 2003, pp. 469 – 472.

[3] Y. Hamamura, et. Al "Repair yield simulation with iterative critical area analysis for different types of failure," Proceedings of IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems, Nov. 2002, pp. 305-313.

[4] Shoukourian, S.; Vardanian, V.; Zorian, Y.; SoC yield optimization via an embedded-memory test and repair infrastructure. Design & Test of Computers, IEEE Volume 21, Issue 3, May-June 2004 Page(s):200 – 207

[5] Fadi J. Kurdahi, A. M. Eltawil, Y.-H. Park, R. N. Kanj, S. R. Nassif, “System-Level SRAM Yield Enhancement”, Proceedings of the 7th International Symposium on Quality Electronic Design, (ISQED 2006), pp. 179 – 184, 2006.

[6] Kang Yi, Kyeong Hoon Jung, Shih-Yang Cheng, Young-Hwan Park, Fadi Kurdahi, Ahmed Eltawil, "Design and Analysis of Low Power Image Filters towards Defect-Resilience Embedded Memories for Multimedia SoCs", ACSAC06 (11th Asia-Pacific Computer Systems Architecture Conference), September, 2006. pp. 295 – 308.

[7] Fadi J. Kurdahi, Ahmed Eltawil, Kang Yi, Young-Hwan Park, Yervant Zorian, "Accounting for Chip Yield at the Application Level : A Case Study of a H.264 Video Application", IEEE International Workshop on Design for Manufacturability & Yield (DFM&Y 2006), Oct., 2006.

[8] H. Arakida et. al. “A 160mW, 80nA Standby, MPEG-4 Audiovisual LSI with 16Mb Embedded DRAM and a 5GOPS Adaptive Post Filter ». Proc. ISSCC 2003. Paper 2.3.