ISSN 0002-306X. Proc. of the RA NAS and NPUA Ser. of tech. sc. 2020. V. LXXIII, N3

UDC 621.382.13 MICROELECTRONICS

V. SH. MELIKYAN, N.E. MAMIKONYAN

A SELF-LEARNING DYNAMIC MEMORY DESIGN METHOD

A novel approach to memory design method is presented, which is targeted to generate memories with an acceptable value of timing performance and area increase while optimizing power consumption and the IR drop. The machine learning methods are used for IR drop and refresh cycle time estimations, which help to reduce power consumption.

The experimental results show that with the proposed method power consumption is reduced by 10…15% while having a loss of 5…14% in the area.

Keywords: DRAM, OpenRAM, Machine learning.

Introduction. It is well known [1], that the main macro parameters of the integrated circuits (IC) and their separated parts are performance, power consumption, and the occupied area in the semiconductor crystal. This refers to one of the most common types of ICs: memories. Due to the recent development of the ICs [2] and their separated parts design and prototyping, the power consumption reduction challenges in the ICs are dominant in comparison with the other mentioned macro parameters. Often, power consumption is being reduced by sacrificing the speed as it is being done in the multi-core processors [3]. But there are multiple applications of the ICs and their separated parts in which ensuring high-speed performance is more important than power consumption reduction. For this reason, the most common tools of designing memory ICs, memory compilers, should be able to meet the requirements ratio for speed and power consumption in each specific design. This means that memory compilers must be able to select the correct ratio between speed and power consumption. The best way of solving this problem is the machine learning (ML) application in the memory design process. But the famous memory compilers [4-8] either do not have a self-learning ability [9] or if they have that ability of self-learning, they do not provide the best ratio between speed and power consumption [10,11].

For example, OpenRAM is one of the best open access memory compilers [4]. OpenRAM is a Python-based and flexible compiler, which does not depending on a specific technology and can be easily ported from one node to another. Based on this, OpenRAM is the basic compiler on which many other types of research are carried out [5,7-9].

The OpenRAM memory compiler consists of eight blocks (Fig. 1).

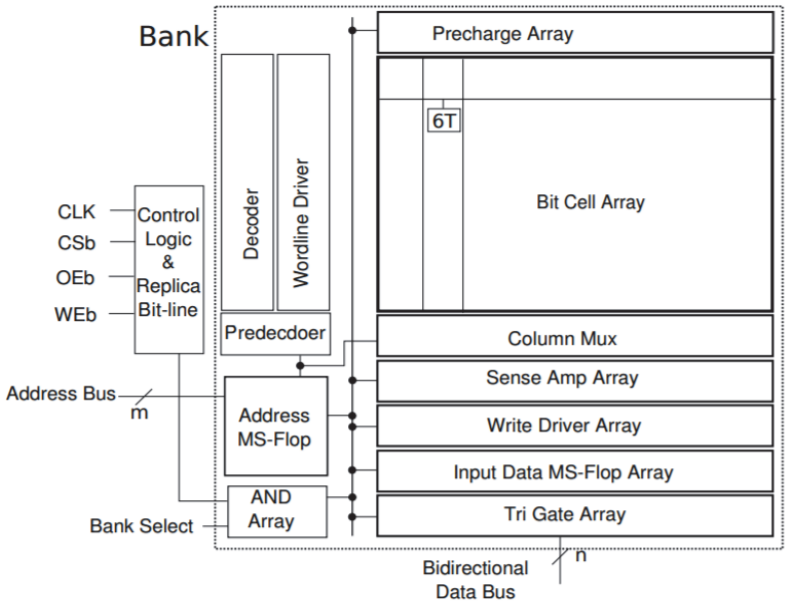

Fig. 1. An OpenRAM memory block

The main principle of the OpenRam is based on increasing the speed of the memory due to the mixing of the memory banks, as a result of which the access time to the memory bank is reduced. However, to achieve a reduction in the memory access time, the memory bank size must be small, otherwise, the wire connections between muxed memories will increase transition time. This can be treated as a disadvantage for this architecture since the technology demands big memory banks. Another example is an asynchronous memory compiler (AMC) [5], which is based on the OpenRAM memory compiler. This compiler solves the access time problem on the big memory systems, which comes from the OpenRAM memory compiler architecture. This approach is to combine sub-banking and tree-structure methods (Fig. 2). Each memory bank has sub-banks and a sub-bank decoder. And by combining with a tree-structure, the access time of any memory address will be the same. This approach allows having big memory banks in comparison with OpenRAM while adding a logical unit for the sub-banking process control. This will keep memory access time in the limits while the power consumption will be increased because of the standard cell count increase. The main disadvantages of this compiler are the high-power consumption and area increase.

The compilers do not consider the best trade-off between power consumption reduction and performance, a novel method is needed from the best ratio between the speed and power consumption, which is described in the next section.

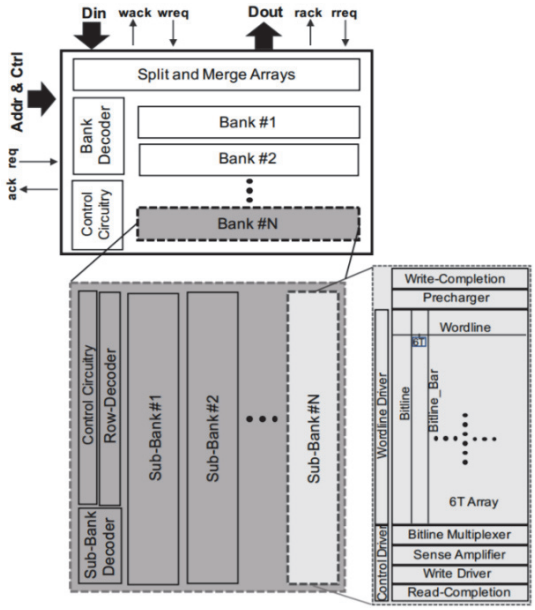

Fig. 2. An AMC multi-bank architecture

Self-learning dynamic design tool methodology. A novel method of memory compiler is presented, which is meant to meet power requirements using ML algorithms. This method is created considering most of the options of the OpenRAM memory compiler. One of the main additions to the OpenRAM memory compiler is the DRAM memory generator insertion into the OpenRAM logic. In addition to the inbuilt features, the self-learning dynamic design tool methodology has 4 main components (Fig. 3), which are controlled by switches and can be turned on and off from tool settings to add flexibility into the tool working process.

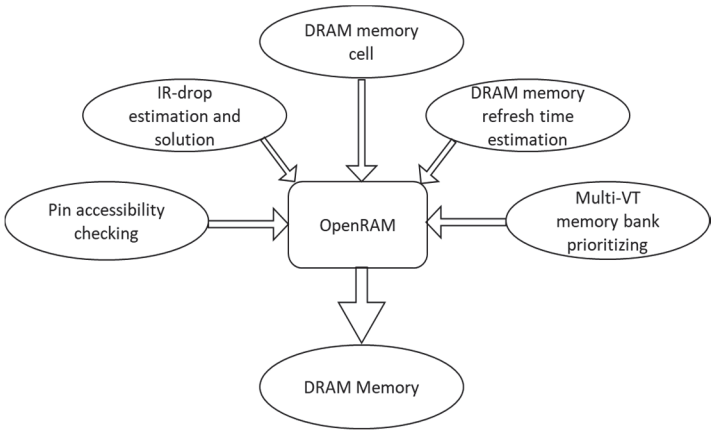

Fig. 3. Proposed DRAM compiler based on OpenRAM

The activation of the specific component can enhance and optimize the memory design:

• Multi-VT memory bank prioritizing method [12] – For reducing the power consumption of the memory addresses which are rarely used, different VT memory bit cells are combined in one memory block. To control the memory address selection form one block with the different VT, a prioritizer algorithm (Fig. 4) is used. The algorithm is aimed at creating priorities for the memory addresses based on counting the access of the addresses for one VT memory bank. Each memory bank from the memory system has its counter for counting the accesses. For example, if the addresses from one memory bank have been accessed n times, the counter value will be increased by n. Rather than separated bank counters, the memory block has one more counter which is controls the total amount of the accesses after which the priorities of the memory banks must be recalculated. Based on the counters, value priorities of the memories are generated. If the memory bank has a priority change the memory address mapper swaps the addresses to the appropriate address with the required priority.

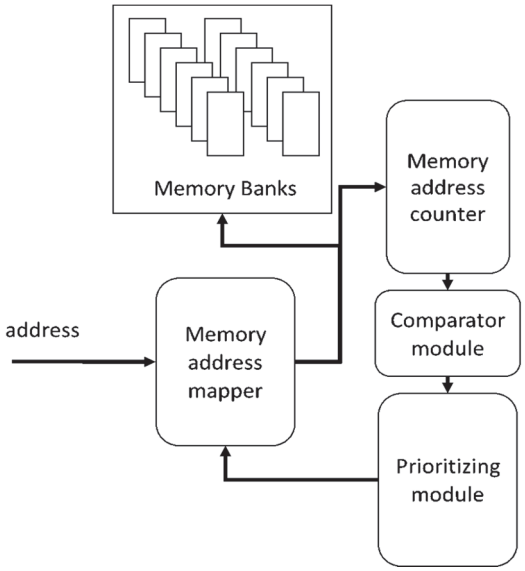

Fig. 4. A memory address prioritizer block diagram

• IR-drop estimation and solution using ML algorithms [13] – The supply networks of the IC are always created in a structured and symmetric way. The power and ground network must be enhanced to ensure the functionality of the IC if the design has blocks with the dominant amount of different VT cells as in multi-VT memory banks. The resistance of the network must be reduced in the areas

where a possible IR drop is estimated using the ML-based algorithm. The automated interconnect via insertion adds more connections from the top to lower metal layers with appropriate metal-via enclosures (Fig. 5). This technique decreases the voltage drop in the power and ground network.

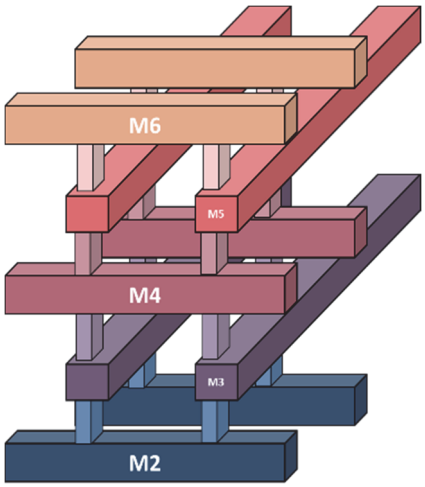

Fig. 5. Via ladder view

• DRAM memory refreshing time estimation based on statistical blockade [14]

– For ensuring stored data in the DRAMs, the refresh cycle time must be small. On the other hand, the refresh cycle time must be increased for optimizing power consumption coming from frequent refreshes. This means that the best refresh cycle time must be selected for power consumption reduction, which will not bring to the loss of the stored data. The Monte Carlo (MC) simulation is used with a statistical blockade method for estimating the refresh cycle time (Fig. 6). The statistical blockade is an ML-powered method with the help of which the MC simulation count is reduced. For a ML model training, the input set is being constructed from a small count of MC simulations. The trained model is used to reduce the simulations count in case if the MC simulation does not make a high impact on the estimation. As a result, the best refresh cycle time can be achieved, which will reduce power consumption. This kind of technique will ensure the consideration of the data which will have an impact on the results.

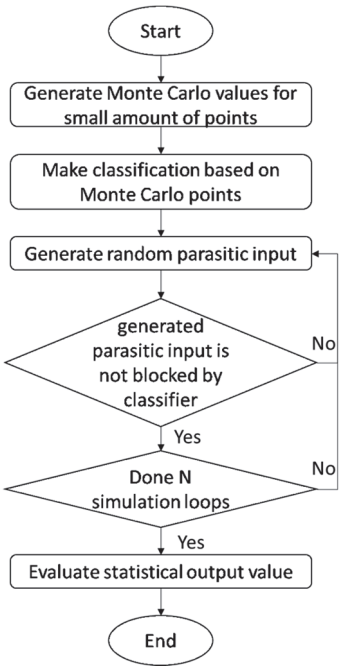

Fig. 6. The statistical blockade method

• Pin accessibility checking integration into the design [15] – As memory blocks have a large number of pins and mostly, they are placed near to each other, finding the best pin placement option will help in later processes. With pin accessibility checking, better options for memory block pin placement can be achieved, hence in memory blocks, placement and routing stage implementation can be done faster and with high accuracy. After the memory block design, the inbuilt method places and tests the memory blocks with different orientations, and on different placement rows (Fig. 7).

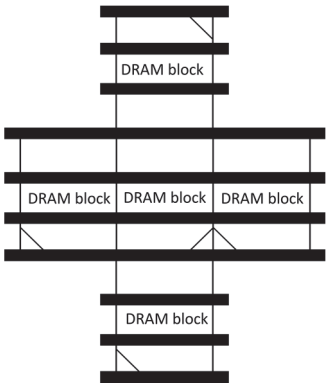

Fig. 7. Pin access check

This method is mostly devoted to tree-style memory block sub-banking enhancement for the best memory selection on the complex tree structure.

Experimental results. The implemented method is compared with OpenRAM and AMC memory compilers' methods.

Experiments are done on SRAM type memories for comparisons. Different configurations of SRAM memories are used:

- Word line size: 8bit, 32bit, and 64bit
- Memory capacitance: 32GB, 128GB and 256GB

The results of the comparisons are listed in Table 1.

Table 1

Comperison results

			OpenRAM			AMS			Our method		
Word size bit	Colunm muxing number	Number of rows	Cycle time, ns	Avg. Power, mW	Bit Efficiency, %	Cycle time, ns	Avg. Power, mW	Bit Efficiency, %	Cycle time, ns	Avg. Power, mW	Bit Efficiency, %
8	1	32	6.4	13	18	5.5	12.2	16.00	6.5	11	15
8	2	64	8.8	18	34	7	18.3	3.00	9	16.7	28
8	4	128	10.6	34.6	48	9.4	34	44.00	10.8	29.6	38
32	1	32	8.2	34	39	6.4	30	37.00	8.4	27.3	35
32	2	128	13.2	51	62	10.4	49	6.00	13.8	44.7	4
32	4	256	22	92	75	13.2	100	73.00	22.9	86.5	49
64	1	64	12	54	59	7.9	54	57.00	12.9	47.3	54

.0Table 2

The compiled DRAM characteristics

Word size bit	Colunm muxing number	Number of rows	Cycle time ns	Avg. Power, mW	Total Area, mm2	Bit Efficiency, %
8	1	32	7.3	20.02	0.64	3
8	2	64	9.7	25.7	1.23	48
8	4	128	12.1	32.8	3.21	69
32	1	32	8.7	51.3	1.1	63
32	2	128	14.7	72.7	5.9	71
32	4	256	24.2	113.7	23.7	76
64	1	64	13.7	89.4	2.6	63
64	2	256	28.1	135.6	30.3	78

Apart from SRAM memories, the method has been tested on DRAM memories. The results are listed in Table 2.

Conclusion. The presented novel method of memory compiler reduces power consumption by 10..15% by using smart machine learning methods, while the access time of the memory is increased by 5…14%.

REFERENCES

1. **Melikyan V.** Simulation and Optimization of Digital Circuits // Springer, Cham Switzerland.- 2018.- P. 247-299.
2. **Atkin E.V.** Trends in integrated circuit design for particle physics experiments // Journal of Physics: Conference Series. 10.1088/1742-6596/798/1/012204.- 2017.
3. **Gepner, Pawel & Kowalik, Michal.** Multi-Core Processors: New Way to Achieve High System Performance // PARELEC 2006: Proceedings: International Symposium on Parallel Computing in Electrical Engineering. -2016.- P. 9-13. 10.1109/PARELEC.2006.54.
4. OpenRAM: An open-source memory compiler / **M.R. Guthaus, J.E. Stine, S. Ataei, Brian Chen, et al //** 2016 IEEE/ACM International Conference on Computer-Aided Design (ICCAD).- Austin, TX, 2016.-P. 1-6,doi: 10.1145/2966986.2980098.
5. **Ataei S. and Manohar R.** AMC: An Asynchronous Memory Compiler // 2019 25th IEEE International Symposium on Asynchronous Circuits and Systems (ASYNC).- Hirosaki, Japan, 2019.-P. 1-8, doi: 10.1109/ASYNC.2019.00009.
6. Synopsys' educational generic memory compiler / **R. Goldman, K. Bartleson, T. Wood, V. Melikyan, and E. Babayan //** EWME.- May 2014.-P. 89–92.
7. **Wu S., Zheng X., Gao Z., and He X.** A 65nm embedded low power SRAM compiler// DDECS.-April 2010.-P. 123–124.
8. **Ataei, Samira & Manohar, Rajit.** A Unified Memory Compiler for Synchronous and Asynchronous Circuits // International Conference On Computer Aided Design, Second Workshop on Open-Source EDA Technology (WOSET).-2019.
9. **Ataei, Samira & Gaalswyk, Matthew & Stine, James.** A high performance multi-port SRAM for low voltage shared memory systems in 32 nm CMOS // IEEE 60th International Midwest Symposium on Circuits and Systems. 10.1109/MWSCAS.2017.8053153.- 2017.-P. 1236-1239.
10. **Last Felix, Max Haeberlein, and Ulf Schlichtmann.** Predicting Memory Compiler Performance Outputs Using Feed-Forward Neural Networks // ACM Transactions on Design Automation of Electronic Systems Crossref. Web.-2020-25.5.-P.1-19.
11. Sparse regression driven mixture importance sampling for memory design / **M. Malik, R.V. Joshi, R. Kanj, et al //** IEEE Transactions on Very Large Scale Integration (VLSI) Systems.-2017.-P. 63-72.
12. **Mamikonyan N.** DRAM structure with prioritized memory bank using multi-VT bit cells architecture // IEEE East-West Design & Test Symposium (EWDTS).-2020-P. 383-385.
13. **Mamikonyan N., Meliqyan N.V., Musayelyan R.H.** IR drop estimation and optimization on DRAM memory using machine learning algorithms // IEEE East-West Design & Test Symposium (EWDTS).-2020.-P. 386-388.
14. **Mamikonyan N.E.** DRAM memory refresh time estimation and optimization based on statistical blockade method // Proceedings of Engineering Academy of Armenia.- 2020.- V. 17, N 1.-P. 105-109.

15. **Abazyan S.S., Jampoladov V.A., Mamikonyan N.E.** Standard cell pin access checking integration into test design verification // Proceedings of NAS RA and NPUA. Series of Tech. sci.-2020.-V. 73, N. 1.- P. 74-81.

National Polytechnic University of Armenia. The material is received on 05.09.2020.

Վ.Շ. ՄԵԼԻՔՅԱՆ, Ն.Է. ՄԱՄԻԿՈՆՅԱՆ

ԻՆՔՆԱՈՒՍՈՒՑՎՈՂ ԴԻՆԱՄԻԿ ՀԻՇԱՍԱՐՔԵՐԻ ՆԱԽԱԳԾՄԱՆ ՄԵԹՈԴ

Ներկայացված է հիշողության նախագծման նոր մոտեցում, որի նպատակն է գեներացնել հիշողություն, որը կունենա օպտիմալ հզորության ծախս և IR անկում՝ ժամանակային պարամետրերի և մակերեսի ընդունելի աճի հաշվին։ Օգտագործվել են մեքենայական ուսուցման մեթոդներ՝ IR անկման և հիշողության թարմացման ժամանակի կանխատեսման համար, որոնք նվազեցնում են հզորության ծախսը։

Փորձնական արդյունքները ցույց են տալիս, որ մշակված մեթոդի կիրառմամբ հզորության ծախսը նվազել է 10…15%-ով, մինչդեռ մակերեսի կորուստը մոտ 5…14% է։

Առանցքային բառեր. DRAM հիշողություն, OpenRAM, մեքենայական ուսուցում։

В.Ш. МЕЛИКЯН, Н.Э. МАМИКОНЯН

САМООБУЧАЮЩИЙСЯ МЕТОД ПРОЕКТИРОВАНИЯ ДИНАМИЧЕСКОЙ ПАМЯТИ

Представлен новый подход к методу проектирования памяти с целью создания блоков памяти с приемлемыми значениями временных характеристик и увеличения площади при оптимизации энергопотребления и падении напряжения. Используются методы машинного обучения для оценки времени цикла обновления и падения напряжения, что помогает снизить энергопотребление.

Экспериментальные результаты показывают, что с помощью предлагаемого метода энергопотребление снизится на 10…15% при увеличении площади на 5…14%.

Ключевые слова: DRAM память, OpenRAM, машинное обучение.