2015 15th Non-Volatile Memory Technology Symposium (NVMTS)

# Nonvolatile Multibit SRAM, Bit Level Caching, and Multi-context Computing for IoT

Yanjun Ma
Intellectual Ventures
Bellevue, WA 98005
*Email:* yma@intven.com

***Abstract* — We propose a new memory hierarchy for energy conscious internet of things (IoT) integrated circuits. In this scheme a new type of SRAM cell – hybrid of SRAM/DRAM or SRAM/NVM memory – that is capable of storing multiple bits are used to bring caching to the bit level, near processor cores. This new scheme is expected to have energy efficiency advantage over traditional memory hierarchy for multi-context computation, especially suitable for many IoT applications.**

***Keywords: SRAM, Memory Hierarchy, Cache, RRAM, STT-RAM***

## I. Introduction

Semiconductor industry is driven by the economics of integration – integrating more and more transistors, which in turn enables the integration of more functional blocks on one die that can be processed in parallel, in particular, the integration of various types of memory for system on chip (SOC) applications.

Traditional memories, from register files, SRAM, DRAM, flash memory to disk and tape storage, are arranged in a hierarchy according to their speed, as illustrated in Figure 1. This memory hierarchy is designed for computing speed – due to the capacity constraint of the memory with the highest speed, i.e., SRAM, only a small SRAM can be integrated in the process core. In this scheme, data and codes are stored away from the processing cores, often off chip. During execution, instructions and data are brought to successive higher levels of cache, closer and closer to the cores.

The expected growth and unique characteristics of internet of things (IoT) applications require that we rethink this traditional memory hierarchy. For most IoT applications power availability and consumption is the most important concern. Also, IoT devices often have to switch context to handle different wireless protocols, such as Zigbee, Bluetooth, or Wifi, or manage many types of sensors. In this regard, the traditional computing memory hierarchy is not ideal because it overly relies on the volatile memory such as DRAM and SRAM. In addition, data and/or codes often reside relative far from the processing cores, resulting in high energy consumption due to data transfer over long bus. As is well known, the energy cost of transporting one bit of data from an off-chip memory or storage device, such as DRAM or flash, can be as much as 1000x more than the energy cost of transferring one bit from the local cache or register file [1]. This is especially costly during context switching as the whole cache usually needs to be refreshed..

## II. Cache at the Bit Level

In this paper we propose a new memory hierarchy, where data and code caching and storage are brought close to the computing cores, also shown in Figure 1. The basic technology is named caching at bit level (CABL). In this scheme, at each memory level, e.g. register file and L1-L3 cache, each memory cell will have its own cache or backup bits. This implementation allows each SRAM bit to store or cache content for later use in a lower cost memory, located right within the basic SRAM cell.

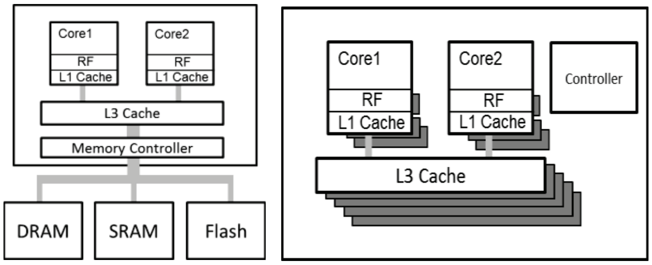


Figure. 1 Traditional memory hierarchy (left) and the proposed memory hierarchy using multibit SRAM cells and CABL technology (right).

These stored or cached bits for each SRAM bit can be used to store configuration or instructions for contexts different from the active context. A controller can be designed to be in charge of switching the data/instructions between the active context and the cached data/contexts.

Traditional cache management methods can be adapted to this new scheme. For example when an active cache line is to be ejected from the active L1 cache, it may instead be stored in the backup bits of the L1 cache instead.

The cached bits can be stored in either volatile DRAM or non-volatile memory (NVM) storage elements. When NVM elements are used, the new memory structure can be used to enable instant-on processors where at power on the processor can load the data and instructions from these cached bits and start processing quickly, instead of loading from the slower main storage in the conventional architecture.

## III. Multibit Hybrid SRAM Cells

CABL is enabled by a novel hybrid SRAM cell that can contain multiple bits, including one active bit and several cached bits [2]. Some of the SRAM cell designs are shown in Figure 2. They include a number of differential cell and single ended cell designs that are best implemented using the emerging STT-RAM or bipolar RRAM. In this type of SRAM cells, there is one active bit that is stored in the node Q and Q_bar, similar to a normal SRAM cell, and a number of “cached” or backup bits, stored in programmable resistors pairs, for example $R_{1\_1}$ and $R_{1\_0}$, $R_{2\_1}$ and $R_{2\_0}$,, in the differential design, or $R_1$ and $R_0$, $R_2$ and $R_0$ for the single ended design.

The programmable resistors can take two resistance states, a high resistance state (HRS) or a low resistance state (LRS). The difference in the resistance values of a resistor pair represents the state. For example, $R_{1\_1}$ = 20 kΩ (the HRS) and $R_{1\_0}$= 10 kΩ (the LRS) represents “1”, while $R_{2\_1}$ = 10 kΩ $R_{2\_0}$= 20 kΩ represents a “0” state. For the single ended cell, the common resistor may be a fixed value resistor taking an intermediate resistance value between HRS and LRS, e.g. ~15 kΩ in the above example.

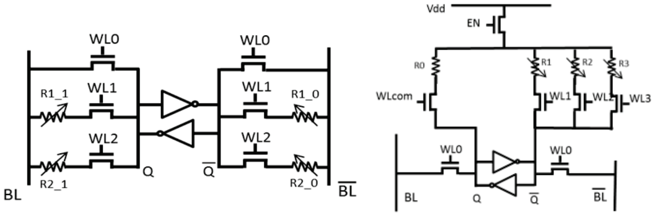


Figure 2 Examples of nonvolatile multibit SRAM cell designs, differential design (left) and single ended design (right). The SRAM cell can store one active bit in the Q/Q-bar node and multiple bits in the transistor-resistors pairs as cached bits. The resistors can be implemented using STT-RAM magnetic tunneling junctions or resistive memories such as conducting bridge RAM.

Figure 3 shows another multibit SRAM cell design and illustrates the programming and storing operations. In this cell, one of the cached bits can be restored to be the active bit in approximately one clock cycle [2]. We used STT-RAM as the implementation example where the programmable resistors are magnetic tunneling junctions (MTJs). In Figure 3 there is an additional equalization transistor controlled by signal WE.

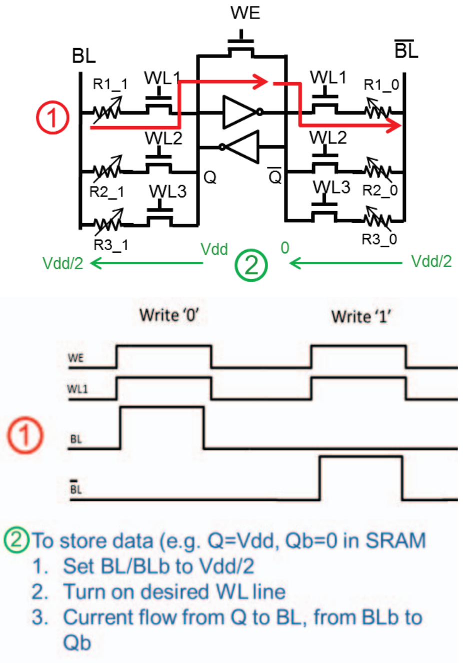
To PROGRAM the MTJs, we follow the case labeled (1) in Figure 3. With BL and BL_bar properly biased, e.g. shown in the figure and also in Table 1, current will flow along path labeled (1). When the current exceeds that critical current density of the MTJs, MTJ $R_{1\_1}$ may be programmed into a HRS. At the same time $R_{1\_0}$ may be programmed into a LRS. This process then stores one bit in the resistor pair. To store a complementary bit, the biases on BL and BL_bar are reversed.

Another operation of the SRAM cell is to STORE the active state of Q/Q_bar into one of the resistor pairs for non-volatile storage. This is accomplished by setting BL and BL_bar both to Vdd/2. Then when the corresponding WL transistors, e.g. WL3, are enabled, the current will follow as indicated by the path (2) in Figure 3. Because of the opposite directions the currents are following through the resistors pairs $R_{3\_1}$ and $R_{3\_0}$ will be set to different resistance state, one to HRS and the other to LRS.

Additional operations include:

(3) RESTORing data stored in one of the resistor pair when power is first turned on: keeping BL/BL_bar at GND, then ramping up Vdd to the inverters while enabling the selected WL transistors. The difference in resistance between the selected resistor pair will cause voltage imbalance between the two nodes Q and Q_bar. The positive feedback action of the cross coupled inverter pair will then swing these nodes to the correct value.

(4) SWITCHing to a cached state from an active state: setting BL/BL_bar to GND, then enabling WE briefly to equalized Q and Q_bar while enabling the selected WL and then quickly turning off WE. Again the positive feedback action of the cross coupled inverters will move the Q and Q_bar nodes to the correct value. The Vdd to the inverter pair can be either on or off during this switch. If the switch is performed fast enough, the energy consumed during the switching can be minimal.



②To store data (e.g. Q=Vdd, Qb=0 in SRAM

1. Set BL/BLb to Vdd/2
2. Turn on desired WL line
3. Current flow from Q to BL, from BLb to Qb

Figure 3. Programing (1) and restoring (2) data to from a STT-RAM based multibit SRAM cell.

Table1 summarizes the programming and restore operations described above. Simulations have shown that the cell can perform as described [2].

**Table 1. Operations of multibit SRAM cell**

| Operation | BL | BL_bar | WE |
|---|---|---|---|
| PROGRAM “1”/”0” | Vpp/GND | GND/Vpp | On |
| STORE | Vdd/2 | Vdd/2 | Off |
| RESTORE | GND | GND | Off |
| SWITCH | GND | GND | On then off |

These multibit nvSRAM cell designs can be considered a hybrid of a traditional SRAM cell with NVM cells. In essence, we utilize the cross coupled inverters of each SRAM cell also as the sense amplifiers for the cached bits to achieve efficiency in area. Another implementation of the multibit SRAM is a SRAM/DRAM hybrid, a volatile variation [3] of the multibit SRAM.

The key to the success of this type of multibit SRAM cell will be to use memories with much smaller bit cells than the "mother" 6T SRAM cell. Fortunately most of the candidate memories fit this requirement. Both STT-RAM and RRAM memories have projected cell size of ~6F$^2$ , much smaller than the area (60-120 F$^2$ ) of the traditional SRAM cell [4-9]. Similarly embedded DRAM cells can be as small as 1/6 of the equivalent SRAM cells [10,11]. As a consequence, multibit SRAMs have the advantage of smaller area than that of SRAM with equivalent number of bits while maintaining the SRAM interface and read/write process.

STT-RAM and RRAM are good candidates for implementing this type of hybrid multibit SRAM as recent reports indicate that these memories are close to commercial availability [4-9]. In fact, the differential design of our memory cell, e.g. Figure 3, doubles the window margin for a given single ended memory technology and can speed up the adoption of emerging memories such as STT-RAM and RRAM for volume production. These emerging memories often have issues with memory window size when manufactured in volume [4-9]. Single ended cell design can tolerate less process margin but has the advantage of being more area efficient.

## IV. Virtual Memory Bank

Moving from the bit level to the block level, we propose that the cached bit can be grouped into virtual memory banks (VMB) much like the virtual memory concept handled by operating systems. A memory with the VMB architecture is shown in Figure 4, where a memory built with multibit SRAM cells will have one active memory bank and several cached or virtual memory banks.

The active bank can be operated exactly like a traditional SRAM. When the memory content is not in the active bank, a page fault can be triggered and a cached page containing the address of the data in one of the virtual memory banks may be switched to be the active bank. The VMB architecture may significantly reduce the area of the memory with minimal read/write latency penalty. Switching is costly in energy consumption but the temporal and spatial locality will help reducing the switching frequency.

This architecture can be applied to all conventional memories. With a well-designed controller to perform the switching between virtual memory banks, they could be a drop-in replacement for current memories.

## V. Discussion and Conclusions

Caching at Bit Level can be considered a new layer in computing architecture, i.e., the usual L1-L2-L3 cache hierarchy, with the following advantages: higher density, fast switching and low power consumption when nonvolatile memory is used.

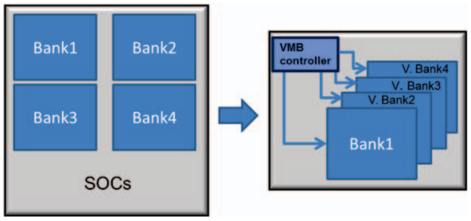


Figure 4. A conventional memory with multiple memory banks may be implemented into multiple virtual memory banks (VBM) using multibit SRAM cells and CABL technology.

The key limitation of the multibit SRAM, CABL, and the VMB architecture is that only one bit is active at any given time, the rest are dormant. As a result, this new architecture is more suited for use in time multiplexed computation tasks, where reduced area and power consumption can be an advantage. In particular, because of the fast switching enabled by the SRAM designs, it is suitable for multicontext computing tasks that need very frequent context switching, best for microseconds to milliseconds switching scenarios such as those typically found in wearable devices.

In conclusion we discussed a new memory hierarchy utilizing multibit SRAM for bit level caching. Several new concepts including caching at bit level and virtual memory bank are discussed. The new memory maintains the existing SRAM interface and can be adopted to work in the current computing memory architecture. We expect the new multibit SRAM cells to find extensive use in IoT applications, where the existence of multi-RF protocols and many varieties of sensors will demand more energy efficient multi-context processing that can be best addressed with the CABL technology.

## References

[1] S. Amarasinghe, et al, "Report of the workshop on exascale programming challenges," Technical Report, US Department of Energy, 2011.

[2] Y. Ma, "Novel Multi-bit Nonvolatile SRAM cells for Runtime Reconfigurable Computing", IEEE International Memory Workshop, 2015.

[3] W.-K. Yu *et al*., "SRAM-DRAM Hybrid Memory with Applications to Efficient Register Files in Fine-Grained Multi-Threading", *ISCA*, pp. 247-258, June 2011.

[4] See, e.g., E. Vianello, et al, "Resistive Memories for Ultra-Low-Power embedded computing design", pp.144-147, *IEDM Tech. Dig,* 2014.

[5] S. Yuasa, *et al.,* "Future Prospects of MRAM Technologies," pp.56-59, IEDM Tech. Dig 2013.

[6] C.J. Lin *et al.*, "45nm low power CMOS logic compatible embedded STT MRAM utilizing a reverse-connection 1T/1MTJ cell," pp. 279-282, *IEDM Tech. Dig*, 2009.

[7] G.S. Kar, "Co/Ni based p-MTJ stack for sub-20nm high density stand alone and high performance embedded memory application", IEDM 2014

[8] W. Shen *et al.,* "High-K Metal Gate Contact RRAM (CRRAM) in Pure 28nm CMOS Logic Process," *IEDM Tech. Dig*, p.745-748, 2012.

[9] J. Zahurak, et al, "Process integration of a 27nm, 16Gb Cu ReRAM", 2014 IEEE IEDM.

[10] R. Brain et al., *A 22nm High Performance Embedded DRAM SoC Technology Featuring Tri-gate Transistors and MIMCAP COB*, Proc VLSI Symp 2013, pp. 16-17.

[11] Y. Wang et al., *Retention Time Optimization for eDRAM in 22nm Tri-Gate CMOS Technology*, Proc IEDM 2013, pp. 240-243.