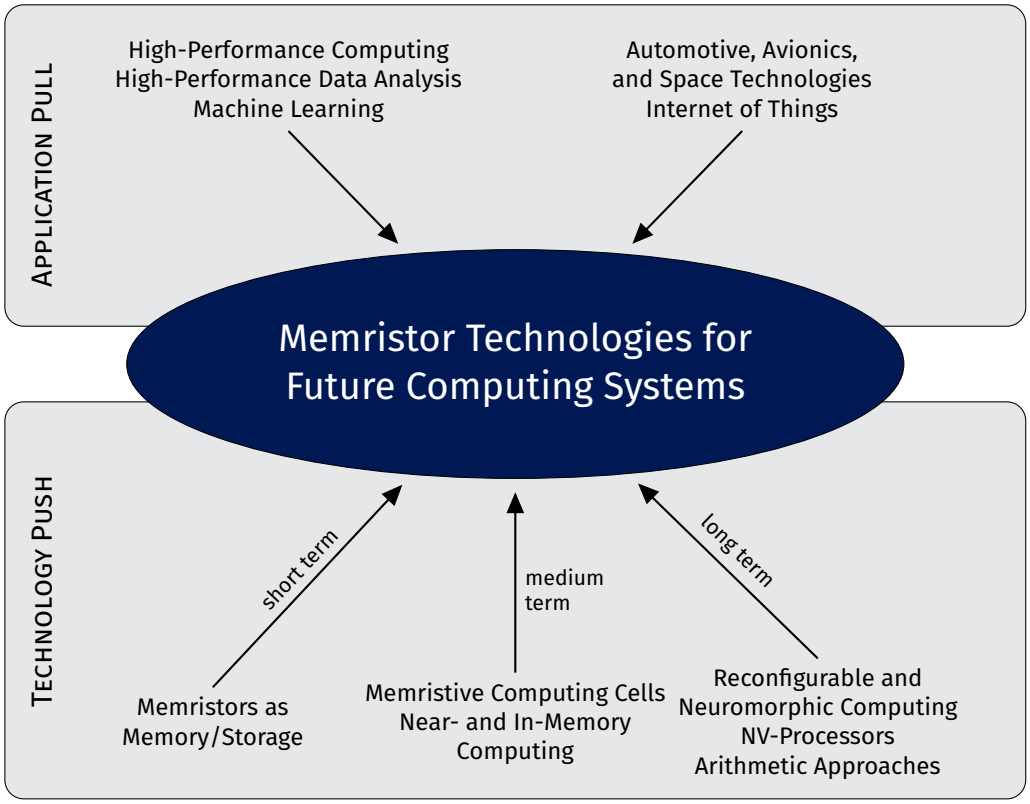
Informationstechnische Gesellschaft (ITG)
and Gesellschaft für Informatik (GI)
Fachbereich Technische Informatik

Implications of Memristor Technologies for Future Computing Systems

June 2019



VDE ITG
GI Gesellschaft für Informatik

Overall Editors and Authors

Prof. Dr. Dietmar Fey, University of Erlangen-Nuremberg
Prof. Dr. Wolfgang Karl, KIT
Prof. Dr. Theo Ungerer, University of Augsburg

Authors

Oliver Bringmann	University of Tübingen	applications, memristive computing
Uwe Brinkschulte	University of Frankfurt	memory hierarchy
Dietmar Fey	University of Erlangen-Nuremberg	overall organisation, memristor technologies, multi-level cells, near- and in-memory computing, memristive computing
Jan Haase	Universität zu Lübeck	applications
Christian Hochberger	Technische Universität Darmstadt	memory hierarchy
Wolfgang Karl	KIT	overall organisation
Dominik Meyer	Helmut-Schmidt-Universität Hamburg	reconfigurable computing, non-volatile processors
Ilia Polian	University of Stuttgart	approximate, stochastic and neuromorphic computing, security and privacy
Werner Schindler	Bundesamt für Sicherheit in der Informationstechnik (BSI)	security and privacy
Heidemarie Schmidt	Friedrich-Schiller-Universität Jena & Fraunhofer ENAS Chemnitz	memristor technologies
Theo Ungerer	University of Augsburg	overall organisation, memristor technologies, near- and in-memory computing

Layout

Florian Haas, Christoph Kühbacher, University of Augsburg

We also acknowledge the sporadic members of the Working Group, as well as the numerous people that provided valuable feedback.

Executive Abstract

Information and communication technology will face radical changes in the next decade. The changes are triggered by an application and software pull and a technology push: On the application side the Internet-of-Things, Industry 4.0, and new Cyper-physical systems require low-energy distributed smart embedded devices which pre-process sensor data. Additionally, High-Performance Data Analysis (HPDA) software filters out relevant information from huge amounts of data, often in in combination with Deep Learning software running on High-Performance Computers (HPC). Both techniques come together in embedded HPC pushing Advanced Driver-Assistance Systems (ADAS) towards future Autonomous Vehicles. On the technology side several upcoming new technologies such as memristors and quantum computing will enable new application possibilities and challenge basic assumtions of computing potentially leading to a "reboot of computing" (US IEEE society) or a "re-invention of computing" (HiPEAC Vision 2017).

Such upcoming changes motivated the "EuroLab-4-HPC Long-Term Vision on High-Performance Computing" of August 2017 [1], a road mapping effort within the EC CSA Eurolab-4-HPC that targets potential changes in hardware, software, and applications in High-Performance Computing (HPC). The Eurolab-4-HPC Vision reports on Die Stacking and 3D Chip Technologies, Non-volatile Memory (NVM) Technologies based on Memristors, Photonics, Memristive Computing, Neuromorphic Computing, Quantum Computing, Nanotubes, Graphene, and diamond-based transistors.

An expert working group within the Chapter ARCS (Architecture of Computing Systems) of the two German computer societies "Gesellschaft für Informatik (GI)" and "Informationstechnische Gesellschaft (ITG)" reviewed the Eurolab Vision and decided to focus in this report on Memristor Technologies and extend their potential usage over the replacement of Flash as NVM also towards usage as computing devices. We selected memristor technologies since we and many experts are convinced that their breakthrough will come even if it is unclear how it will look like in detail and to what extent memristor technologies will emerge not only for storing but also for processing.

[1] Eurolab-4-HPC Long-Term Vision on High-Performance Computing 2017, https://www.eurolab4hpc.eu/vision/

Memristors

Memristors, i.e. *resistive memories*, are an emerging class of Non-volatile Memory (NVM) technologies. The memristor's electrical resistance is not constant but depends on the previously applied voltage and the resulting current. The device remembers its history, the so-called *Non-Volatility Property*: when the electric power supply is turned off, the memristor remembers its most recent resistance until it is turned on again.

We are convinced that memristor technology as memory/storage will strongly influence the memory hierarchy of computer systems in near to mid-future. Usage of memristor technologies for computing by memristive computing, near- and in-memory computing, in reconfigurable and neuromorphic computing, and NV processors will bring radical changes to our compute and IT landscape in mid to long-term future. Since it is uncertain when the different memristor technologies will mature, it is hard to predict which ones will prevail. The security and privacy risks arising by the non-volatility of memristor technologies and its potential in several application domains makes it more than ever necessary to accompany and to shape this process from computer engineering side.

Memristors in the Memory Hierarchy

The computer architecture development in the last 1.5 decades was primarily characterized by energy driven advancement (better performance/Watt ratio). This led to a transition from single- to multi-/many-core, to heterogeneous architectures consisting of a multi-core processor and an accelerator and to deep memory hierarchies, from an on-chip cache hierarchy to still volatile DRAM memory and non-volatile SSDs and disk storage.

Currently, NAND Flash is the most common NVM technology, which finds its usages as on-chip memory in

embedded processors and microcontrollers, as SSDs, memory cards, memory sticks, and also as *Storage-Class Memory (SCM)* in supercomputers. Flash-based SCM is currently applied in supercomputers as an intermediate storage layer between DRAM memory and cheap disc-based bulk storage to bridge the access times of DRAM versus disks. NAND Flash uses floating-gate transistors for storing single bits. This technology is facing a big challenge, because scaling down decreases the endurance and performance significantly. Hence, the importance of memristors as alternative NVM technology increases.

New memristive NVM technologies will strongly influence the memory hierarchy of computer systems. Memristors will deliver non-volatile memory which can be used potentially in addition to DRAM, or as a complete replacement. Memristor technology blurs the distinction between memory and storage by enabling new data access modes and protocols that serve both "memory" and "storage".

The potential impact of memristors on computer architecture, system software and programming environments can be characterized for the memory hierarchy as follows:

- Read accesses will be faster than write accesses, though, software needs to deal with the read-/write disparity, e.g., by database algorithms that favor more reads over writes.
- Faster memory accesses by the combination of NVM and photonics could lead either to an even more complex or to a shallower memory hierarchy envisioning a flat memory where latency does not matter anymore.
- Memory and storage will be accessed in a uniform way.
- Memristor-based memory will allow in-memory checkpointing, i.e. checkpoint replication with memory to memory operations.
- Software and hardware needs to deal with limited endurance of memristor-based memory in case of replacement of DRAM memory and SRAM in last-level caches by memristors.
- The higher throughput and lower memory latency when stacking memory on top of processing may require changes in programming environments and application algorithms.
- Altogether computing will be more memory-centric and less CPU-centric than state-of-the-art computers.
- New security challenges (e.g., persistent attacks, read-out attacks, new types of side-channel and fault-injection attacks on memristive circuits and memories).
- New opportunities for security solutions (e.g., construction of memristive random number generators or physical unclonable functions).
- New challenges for privacy through easier tracking of memristive devices and the need for dedicated solutions such as secure erasure.

We expect that multiple NVM technologies will be successful in the market, based on differing characteristics such as cost, durability, performance, and application. All commercially available memristive memories feature better characteristics than Flash, however, are much more expensive. Even if it is currently unclear when most of the new technologies will be mature enough, and which of them will prevail by a competitive price, it is foreseeable that these technologies will emerge, and that they will influence the architecture of future computing and IT systems. Therefore to our view it is necessary to accompany this process of technology development from the computer engineering side by own research to make proposals to the device community in order to profit in the best possible way from the benefit these new technologies are offering.

Memristors in Near- and In-Memory Computing

Apart from using memristors as non-volatile memory, there are several other ways to use memristors in computing.

Near-memory computing is characterized by processing in proximity of memory to minimize data transfer costs. Compute logic, e.g. small cores, is physically placed close to the memory chips in order to carry out processing steps, memory itself is still separate. Currently the combination of compute logic with 3D stacked DRAM memory chips is mainly researched, but in future memristive memories could be integrated too replacing the volatile SRAM memory by non-volatile memory chips.

In-memory computing goes a step further such that the memory cell itself is not only a storage cell but it becomes an integral part of the processing step. This can help to further reduce the energy consumption and the area requirement in comparison to near-memory computing.

Near- and in-memory computing change the interface between the processor and the memory: memory will in future not only be accessed by loads and stores respectively cache-line misses, but additionally provide a semantically stronger access pattern based on simple operations on a large number of memory cells. It is preferable to process data in-situ directly where they are located before they are sent to the processor cores. However, Near- and in-memory computing technologies are still in the research stage, in particular its combination with resistive memory, and are considered at least as mid-term or probably as more long-term solutions.

Impact of Memristors on Security and Privacy

Recent attacks, such as Meltdown, Spectre and Rowhammer, demonstrated the crucial importance of a system's hardware for its security. Therefore, when a radically new hardware technology like memristors is being adopted, its implications on security and privacy must be carefully assessed. Security has several aspects, and memristors can have positive or negative impact on some of these aspects.

Hardware-based attacks are, in general, more difficult to mount than their more traditional software- and network-based counterparts, but if they are successful, a very large population of manufactured devices is at risk. Storing sensitive data, like passwords or financial records, in memristive NVMs might greatly simplify read-out attacks (accessing the memory content without authorization). In contrast to a conventional, volatile memory which the adversary has to read out in the running system, the same adversary could disconnect the circuit with a memristive NVM from power and analyze the content of its NVMs in a lab. A further representative threat relates to active manipulations: suppose an attacker manages to replace legitimate software in the memory with a Trojan, or essential parameter values with manipulated versions (e.g., in the context of unauthorized car engine tuning).

At the same time, memristive components can contribute to the system security. It is possible to construct modules such as random number generators (RNGs) or physical unclonable functions (PUFs) based on properties of memristors. RNGs are used for, e.g., on-chip generation of secret keys, masks for cryptographic algorithms, and the non-determinism of memristive devices might deliver random bits with very good entropy. Some RNG designs must store their internal state throughout several sessions, and memristive NVMs are an excellent choice for this purpose. PUFs are employed for deriving of secret keys directly from the properties of the hardware, and for special secure authentication protocols. Memristors are currently considered as one source of entropy for PUFs by several researchers.

A related challenge that is currently not intensively considered in the context of memristive technologies is privacy. NVMs, but also memristive PUFs, provide functionalities which can be (mis-)used for providing a unique identifier of a circuit that can be employed to track its owner. The uniqueness property is often beneficial, or even necessary, for security, e.g., for preventing counterfeiting or overbuilding. At the same time, such tracking can be undesired for privacy. Tracking how a circuit has been used can imply the behavioral patterns of its (current or previous) owner. Tracking such data for a large population of circuits, e.g., built into a car or into a smartphone, gives rise to undesired inferences about entire societies or its subgroups.

Security and privacy issues should be considered during the conception of new technologies and systems, rather than designing a product, waiting for an attack to happen, and adding security features a posteriori. In the context of memristive devices, one essential functionality is secure erasure of sensitive data. This functionality should be made available to security-critical software, but implemented directly on hardware level. For example, simply overwriting the NVMs with, e.g., values 0 may not reliably eliminate the erased data due to the hysteresis properties of memristors and multiple rounds of overwriting using pseudorandom values may be needed.

Further Usages of Memristors in Computing

Memristive computing is based on logic circuits that are built by memristors. The memristive cells are used for storing data and processing of small operations

on the data. It is the basis for memristive in-memory computing, but also for far reaching new computing devices.

Neuromorphic computing based on memristive technology utilizes analog properties of memristors and uses memristive devices as biorealistic models of synapses and axons. Memristive NVMs are also attractive for storing weights in neuro-inspired architectures, i.e., hardware-implemented neural network. Furthermore, memristive components are an attractive fabric for novel approximate and stochastic architectures which can tolerate some degree of their components' unreability. Stochastic circuits rely on high-quality random numbers for correct operations, and memristors can provide true random bits with excellent properties.

Another important benefit of memristor technology for its use in computing devices is its *CMOS compatibility* and *small area and power footprint* that can be exploited in hybrid circuits, i.e. combining the integration of memristors and transistors, in which the CMOS is for example used as control circuitry for both storing or pre-processing memristors. In particular this is an important feature of memristor technology for reconfigurable hardware. At the moment reconfigurable hardware is produced in a standard CMOS fabrication process. Configuration memory, BlockRAM, and Look-Up Tables (LUTs) are implemented using SRAM cells or Flash memory. Crossbar switches consisting out of multiple transistors provide routing and communication infrastructure. The main challenges for reconfigurable hardware are a high static power consumption and long interconnection delays. Memristor technology, applied to important building blocks of reconfigurable hardware, i.e. in BlockRAM, CLBs and the Interconnection Network, can help overcoming these challenges.

Memristive cells can be implemented as *multi-level cells* meaning that not only binary but multiple bit values can be stored in one physical cell. This makes it possible to use this feature for implementing carry-free adders based on ternary coding schemes which add independent of the operand's word length in constant time.

Many new application opportunities arise with the use of memristors, due to their non-volatile behavior, their low-energy demand and their compatibility to analog and bio-medical sensors. A promising solution for various application domains is a well-tailored partitioning into memristor-based and classical CMOS-based components to cover dedicated requirements of each application domain. In many application domains, the computation can be partitioned into edge computing (sensor-near computing) and central processing, which can be implemented very differently when using memristor devices. Thus, a lot of different application domains can strongly benefit from the memristor technologies, like IoT devices and wireless sensors, automotive, avionics, and space applications as well as HPC, HPDA, and server applications.

One of these potential application domains are NV processors, i.e. building the full processor by memristive devices. NV processors are able to stop and restart processing immediately. NV processors apply memristors not only for memory but, in principle, also every flipflop within the electronic circuitry is memristive. If every component of a device, e.g. the full CPU, is based on memristive cells, these components can independently be powered off or on according to the current computation and combine power consumption with computational progress.

Conclusions on Research Opportunities

Memristor technologies have already successfully applied into first application domains and are about to influence many other application domains in a very positive way, so it is time to investigate the effects/impact of memristors in a wide range.

Because of the high potential of memristor technologies for research and industry, we call upon researchers to grasp the many opportunities of groundbreaking research in memristor developments and usages. Such opportunities arise from basic research in the different memristor technologies as well as new system and even programming paradigms to utilizing memristors at a higher application level.

The recently accepted DFG priority program on "Memristive Devices Toward Smart Technical Systems" is an important step, in which research focused on modeling and material development of memristive devices is moving to systems. We think that research in computer science targeting system and hardware architectures based on memristive devices has to complement these efforts.

Contents

1 Introduction

Radical changes in computing for the next decade are also foreseen by the US IEEE society, which wants to "reboot computing", and by the HiPEAC Vision 2017, which sees the time to "re-invent computing"—both by challenging its basic assumptions. An expert working group within the Chapter ARCS (Architecture of Computing Systems) of the two German computer societies "Gesellschaft für Informatik (GI)" and "Informationstechnische Gesellschaft (ITG)" took up the challenge to survey the most important topics of technological change that may be upcoming in the next decade. Our target is to provide insight in research challenges and opportunities for German and European research funding which may finally bring forward environment, industry and society.

Our efforts were motivated by the "EuroLab-4-HPC Long-Term Vision on High-Performance Computing" of August 2017 [1], a roadmapping effort within the EC CSA Eurolab-4-HPC that targets potential changes in hardware, software, and applications in High-Performance Computing (HPC) after upcoming Exascale computers, i.e. in the time period of 2023 til 2030.

Because of the necessity to create less energy consuming devices and systems, new technologies are under development. The Eurolab-4-HPC Vision of 2017 reports on Die Stacking and 3D Chip Technologies, Non-volatile Memory (NVM) Technologies based on Memristors, Photonics, Memristive Computing, Neuromorphic Computing, Quantum Computing, Nanotubes, Graphene, and diamond-based transistors.

The ARCS expert group met several times between end of 2017 and spring 2019 to review the existing Visions and discuss upcoming technological challenges. We selected Memristor Technologies as the most important subject because these memristors show in our opinion the best opportunities for research, development and industrial uptake. Several research challenges based on memristor technologies were identified and short chapters on each challenge prepared by the members of the expert group. In spring 2019 the report was finalized, reviewed and discussed by the expert group, and made publicly available in June 2019.

Memristors, i.e. *resistive memories*, are an emerging class of Non-volatile Memory (NVM) technology. The memristor's electrical resistance is not constant but depends on the previously applied voltage and the resulting current. The device remembers its history, the so-called *Non-Volatility Property*: when the electric power supply is turned off, the memristor remembers its most recent resistance until it is turned on again.

In this report we focus on memristor technologies and extend their potential usage over the replacement of Flash as NVM also towards usage of computing devices. We selected memristor technologies because we and many experts are convinced that their breakthrough will come even if it is unclear how it will look like in detail and to what extent memristor technologies will emerge not only for storing but also for processing.

This report is based on the following three observations and time horizons:

- *Already today, and near to mid-future:* We are convinced that memristor technology as memory/storage will strongly influence the memory hierarchy of computer systems in near to mid-future. Memristive memory/storage features much faster access, a much higher write endurance and higher resistance to radiation than Flash memory. it may even be able to replace DRAM memory or be integrated within CMOS processor chips as last-level caches. Memristive memory devices are already available, but currently too expensive to be widely applied. The NV property, however, may lead to security challenges that go beyond current threats, e.g. by non-volatile main memory that retains information after restart of the computer.
- *Mid to long-term future:* Second, we expect usage of memristor technologies for computing by development of memristive computing cells. Memristor technology may first be employed in upcoming near-memory computing by simply ex-

[1] Eurolab-4-HPC Long-Term Vision on High-Performance Computing 2017, https://www.eurolab4hpc.eu/vision/

changing DRAM by memristive memory devices and second in future in-memory computing by applying memristive computing devices itselves. These approaches are still in research stage and an introduction in the market may happen in mid- to long-term future.

- *Long-term future:* Third, totally new technological usages of memristors are devised and currently researched targeting a long-term future. Such opportunities are the use of memristors in reconfigurable and neuromorphic computing, as NV processors building the full processor by memristive devices able to stop and restart processing immediately, and for new arithmetic approaches extending Ternary arithmetic by exploiting the ability to store several bits in a single memristive memory cell. Such technologies, if mature, will bring radical changes to our compute and IT landscape.

Our report is structured as follows (see also the Figure 1.1): First we review the current state of different memristor technologies in Chapter 2.

The succeeding Chapter 3 surveys the research towards memristive multi-level cells. Memristive cells can be implemented as so-called Multi-level cells such that several layers can store several bit values within a single cell. This makes a new arithmetic extending the Ternary arithmetic possible by exploiting the ability to store several bits in a single memristive memory cell.

Next, Chapter 4 looks at opportunities of memristor technologies within the memory hierarchy, i.e. as NV memory/storage replacing Flash, DRAM and SRAM.

Chapter 5 points to security and privacy issues with such technological replacements, in particular, because volatile RAM may partly be replaced by non-volatile memristive memory devices.

Deep memory hierarchies exploit the locality of data and code to improve memory access time and also lead to energy savings by placing the data close to the cores. However, a DRAM access costs several thousand times more energy than a single operation in the cores. A logical step seems to bring computing closer to memory by extending the current High-Bandwidth Memory towards Near-Memory Computing where simple operations take place on the row buffers of an extended memory controller or HBM memory access logic. Memristive memories could replace DRAM memory in such a model. The next logical step is In-Memory Computing where operations are performed directy on the potentially memristive memory cells. Near- and In-Memory Computing based on memristors are topic of Chapter 6.

Chapter 7 suggests memristor technology applied to important building blocks of reconfigurable hardware, i.e. in Configuration memory, Block-RAM, and Look-Up Tables (LUTs) as well as in CLBs and the Interconnection Network, which can help overcoming the main challenges for reconfigurable hardware, i.e. a high static power consumption and long interconnection delays.

Chapter 8 introduces research on Memristive computing, i.e. computing based on memristive technology, applies memristive devices for storing and processing. The necessary devices are still under a strong development process and topic of long-term research.

Memristive computing for support of neural networks and deep learning, for neuromorphic computing and particularly STDP (Spike-Timing-Dependent Plasticity) is reviewed in Chapter 9. Such neuromorphic computing approaches utilize analog properties of memristors, targeting to use memristors as memristive synapses. Related are approaches to apply memristor technologies in approximate and stochastic computing.

Lastly, Chapter 10 introduces several application domains, where memristive devices could be applied. One particularly interesting application could be NV processors, i.e. building the full processor by memristive devices, are able to stop and restart processing immediately. NV processors apply memristors not only for memory but, in principle, also every flipflop within the electronic circuitry is potentially replaced by a memristive NV flipflop. If every component of a device, e.g. the full CPU, is based on memristive cells, these components can independently be powered off or on according to the current computation and combine power consumption with computational progress.

We expect that in near future multiple NVRAM technologies will be successful in the market, based on differing characteristics such as cost, durability, performance, and application. All commercially available memristive memories feature better characteristics than Flash, however, are much more expensive. It is unclear when most of the new technologies will be mature enough and which of them will prevail by a

competitive price. “It’s a veritable zoo of technologies and we’ll have to wait and see which animals survive the evolutionary process," said Thomas Coughlin, founder of Coughlin Associates. Nobody knows what creative engineers will invent from new technologies. So our application chapter gives just some suggestions on future innovation and should point out the myriad of potential innovation chances that will arise in future.

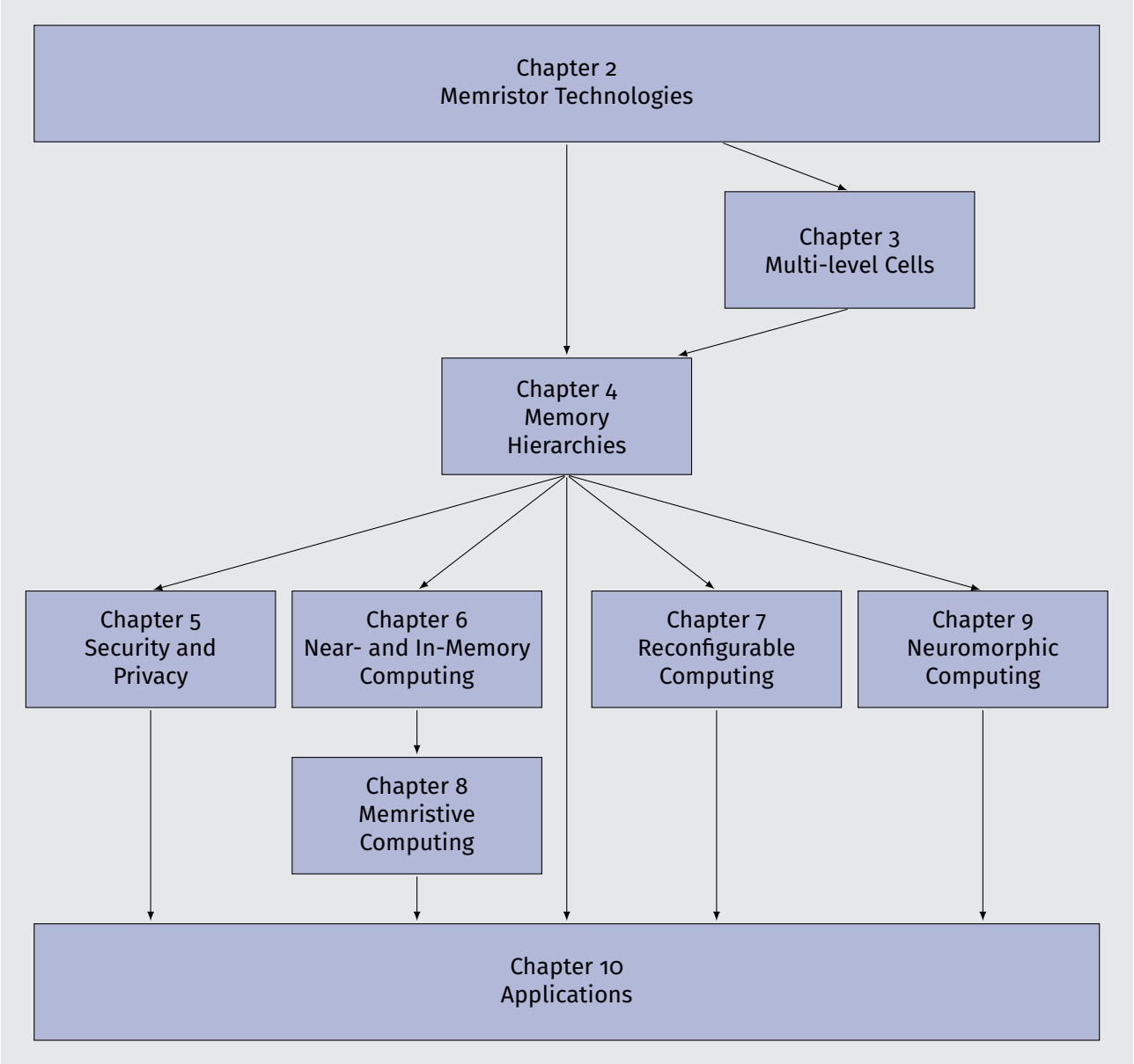


Figure 1.1: Structure of the report

2 Memristor Technologies

Memristors, i.e. *resistive memories*, are an emerging class of Non-volatile Memory (NVM) technology. The memristor's electrical resistance is not constant but depends on the previously applied voltage and the resulting current. The device remembers its history—the so-called *Non-Volatility Property*: when the electric power supply is turned off, the memristor remembers its most recent resistance until it is turned on again [1].

Currently NAND Flash is the most common NVM technology, which finds its usages on SSDs, memory cards, and memory sticks. Flash-based SCM is currently also applied in supercomputers as so-called *Storage-Class Memory (SCM)* (see also section 4), i.e., an intermediate storage layer between DRAM memory and cheap disc-based bulk storage to bridge the access times of DRAM versus disks. NAND and also NOR flash use floating-gate transistors for storing single bits. This technology is facing a big challenge, because scaling down decreases the endurance and performance significantly [2]. Hence the importance of memristors as alternative NVM technology increases. New NVM technologies will strongly influence the memory hierarchy of computer systems. Memristors will deliver non-volatile memory which can be used potentially in addition to DRAM, or as a complete replacement. The latter will lead to a new Storage Memory Class (SCM) in high-performance computers that is much faster than Flash.

Memristor technology blurs the distinction between memory and storage by enabling new data access modes and protocols that serve both "memory" and "storage". Moreover, memristor technology may lead to Memristive Computing by integrating memory and compute capabilities such that in-memory computing is enabled (see section 8) and to new Neuromorphic processing that utilizes analog properties of memristors (see section 9). Using emerging NVM technologies in computing systems is a further step towards energy-aware measures for future computer architectures.

2.1 Memristor Defined by Leon Chua's System Theory

L. Chua [3] assumed already in 1971 that a fourth fundamental two-terminal passive circuit element exists besides the resistor, the capacitor, and the inductor. He called this element a memristor. A memristor should be able to change its resistive features non-volatile in dependence on an outer appearing electrical flux that controls the relation of the devices' inner charge. Since then such memristive features were discovered in nanoscaled devices by a research group around S. Williams at HP labs in 2008 [4].

A *memristor* is defined by Leon Chua's system theory as a memory device with a hysteresis loop that is pinched, i.e. its I–U (current–voltage) curve goes to the zero point of the coordinate system. Considered from a system theoretical view according to Chua a dynamical system is characterized by an internal state variable, x, an external excitation of the system, u, and its output y, which is characterized by a non-linear function h (2.1). The change of its internal state, $\dot{x}$, over time, t, is determined by the time-dependent non-linear function f. In general y and u can be multi-dimensional functions.

$$\begin{aligned} \vec{y} &= h(x, \vec{u}, t) \\ \dot{x} &= f(x, \vec{u}, t) \end{aligned} \tag{2.1}$$

For a memristive system it holds the special case of a dynamic system in which y and u are scalar values. According to (2.2) y is 0 when $u = 0$, which corresponds to a Lissajous figure with pinched hysteresis loop (see Fig. 2.1).

$$\begin{aligned} y &= h(x, t, u) \times u \\ \dot{x} &= f(x, u, t) \end{aligned} \tag{2.2}$$

A memristor itself is a special case of a memristive system with only one state variable, x. Such a memristive system is either current-controlled (2.3), in which

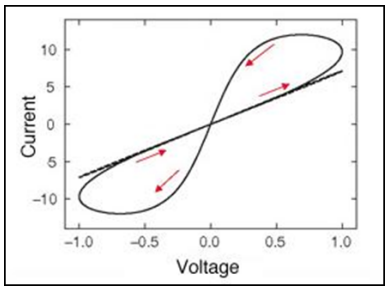


Figure 2.1: Pinched hysteresis I-U curve.

case the internal state variable is the charge, q, controlled by a current I, and an output voltage, V, or it is voltage-controlled (2.4), in which case the system's state variable is the flux, ϕ, controlled by the voltage V, and the output of the system is the current, I.

$$\begin{aligned} V &= R(q) \times I; \\ \dot{q} &= I \end{aligned} \tag{2.3}$$

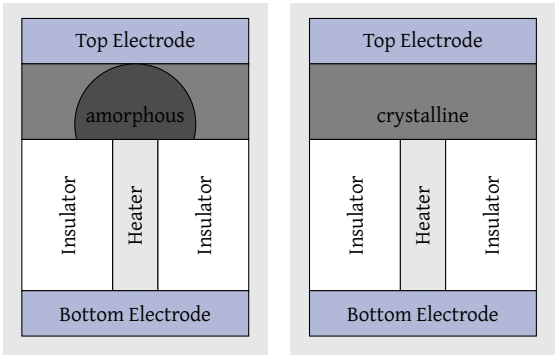
$$\begin{aligned} I &= G(\phi) \times V; \\ \dot{\phi} &= V; \end{aligned} \tag{2.4}$$

2.2 Overview of Memristor Technologies

In practice, several NVM technologies belong to Chua's memristor class:

- *PCM (Phase Change Memory)*, which switches crystalline material, e.g. chalcogenide glass, between amorphous and crystalline states by heat produced by the passage of an electric current through a heating element,
- *ReRAM* (*Resistive RAM*) with the three sub-classes
  - *CBRAM (Conductive Bridge RAM)*, which generates low resistance filament structures between two metal electrodes,
  - *OxRAM (Metal Oxide Resistive RAM)*, in which the thickness ratio between the resistance switching layer and the base layer in a bi-layer bi-layer oxide structure with nearly stoichiometric, oxide layer (resistance switching layer) with higher resistivity and a metal-rich layer with lower resistivity (base layer) is changed by redistribution of oxygen vacancies, and
  - *DioxRAM (Diode Metal Oxide Resistive RAM)*, in which oxygen vacancies are redistributed and trapped close to one of the two metal electrodes and lower the barrier height of corresponding metal electrode,
- *MRAM (Magnetoresistive RAM)* storing data by magnetic tunnel junctions (MTJ), which is a component consisting of two ferromagnets separated by a thin insulator,
- *STT-RAMs (Spin-Transfer Torque RAMs)* as newer technology that uses spin-aligned ("polarized") electrons to directly torque the domains, and
- *NRAM (Nano RAM)* based on Carbone-Nanotube-Technique.

PCM or also called *PRAM or PCRAM* is implemented by a material with thermally induced phase change property. The material changes its atomic structure from highly disordered, highly resistive amorphous structure to long ordered low resistive crystalline state. The structure of the PCM cell used in this work is referred to as mushroom cell as shown in Fig. 2.2.



(a) high resistance (b) low resistance

Figure 2.2: PCM cell structure [5]

In this structure a phase change material layer is sandwiched between two electrodes. When current passes through the heater it induces heat into the phase change layer and thereby eliciting the structure change. To read the data stored in the cell, a low amplitude reading pulse is applied, that is too small to induce phase change. By applying such a low reading voltage and measuring the current across the cell, its resistance and hence the binary stored value can be read out. To program the PCM cell into high resistance state, the temperature of the cell has to be higher than the melting temperature of the material, while to program the cell into the low resistance state the temperature of the cell must be well above the

crystalizing temperature and below melting temperature for a duration sufficient for crystallization to take place [5].

PCM [6, 7, 8, 9, 10] can be integrated in the CMOS process and the read/write latency is only by tens of nanoseconds slower than DRAM whose latency is roughly around 100ns. The write endurance is a hundred million or up to hundreds of millions of writes per cell at current processes. The resistivity of the memory element in PCM is more stable than Flash; at the normal working temperature of 85 °C, it is projected to retain data for 300 years. Moreover, PCM exhibits higher resistance to radiation than Flash memory. PCM is currently positioned mainly as a Flash replacement.

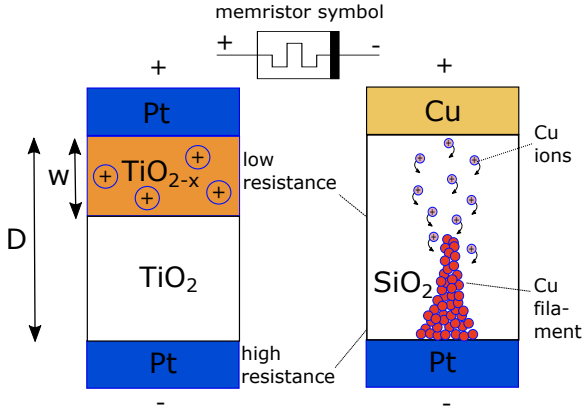


Figure 2.3: Scheme for OxRAM and CBRAM based memristive ReRAM devices.

ReRAM or also called *RRAM* offers a simple cell structure which enables reduced processing costs. Fig. 2.3 shows the technological scheme for ReRAM devices based on OxRAM or CBRAM. the different non-volatile resistance values are stored as follows.

In CBRAM [11] metal is used to construct the filaments, e.g. by applying a voltage on the top copper electrode Cu^+ ions are moving from the top electrode to the bottom negative electrode made in platinum. As result the positively charged copper ions reoxidize with electrons and a copper filament is growing that offers a lower resistance. By applying an opposite voltage this filament is removed and the increasing gap between the tip of the filament and the t op electrode increases resulting in a higher resistance. In an OxRAM-based ReRAM [12, 13] oxygen ionization is exploited for the construction of layers with oxygen vacancies which have a lower resistivity. The thickness ratio between the resistance switching layer, e.g. TiO_{2-x} in Fig. 2.3, and the base layer, e.g. TiO_2 in Fig. 2.3, in a bi-layer oxide structure with nearly stoichiometric, oxide layer (resistance switching layer) with higher resistivity and a metal-rich layer with lower resistivity (base layer) is changed by redistribution of oxygen vacancies.

DioxRAM oxygen vacancies are redistributed and trapped, e.g. by Ti ions in $BiFeO_3$ [14], close to one of the two metal electrodes. The accumulation of oxygen vacancies lowers the barrier height of the corresponding metal electrode [15]. If both metal electrodes have a reconfigurable barrier height, the DioxRAM works as a complementary resistance switch [16]. The resistance of the DioxRAM depends on the amplitude of writing bias and can be controlled in a fine-tuned analog manner [17]. Local ion irradiation improves the resistive switching at normal working temperature of 85 °C [18].

The endurance of ReRAM devices can be more than 50 million cycles and the switching energy is very low [19]. ReRAM can deliver 100x lower read latency and 20x faster write performance compared to NAND Flash [20]. In particular, CBRAM can be written with relatively low energy and with high speed featuring read/write latencies close to DRAM.

MRAM is a memory technology that uses the magnetism of electron spin to provide non-volatility without wear-out. MRAM stores information in magnetic material integrated with silicon circuitry to deliver the speed of SRAM with the non-volatility of Flash in a single unlimited-endurance device. Current Everspin NRAM features a symmetric read/write access of 35 ns, a data retention of more than 20 years, unlimited endurance, and a reliability that exceeds 20 years lifetime at 125 °C. It can easily be integrated with CMOS. [21]

MRAM requires only slightly more power to write than read, and no change in the voltage, eliminating the need for a charge pump. This leads to much faster operation and lower power consumption than Flash. Although MRAM is not quite as fast as SRAM, it is close enough to be interesting even in this role. Given its much higher density, a CPU designer may be inclined to use MRAM to offer a much larger but somewhat slower cache, rather than a smaller but faster one. [22]

STT (*spin-transfer torque* or *spin-transfer switching*) is a newer MRAM technology technique based on Spintronics, i.e. the technology of manipulating the spin state of electrons. STT uses spin-aligned ("polarized") electrons to directly torque the domains. Specifically,

if the electrons flowing into a layer have to change their spin, this will develop a torque that will be transferred to the nearby layer. This lowers the amount of current needed to write the cells, making it about the same as the read process.

Instead of using the electrons charge, spin states can be utilized as a substitute in logical circuits or in traditional memory technologies like SRAM. An STT-RAM [23] cell stores data in a magnetic tunnel junction (MTJ). Each MTJ is composed of two ferromagnetic layers (free and reference layers) and one tunnel barrier layer (MgO). If the magnetization direction of the magnetic fixed reference layer and the switchable free layer is anti-parallel, resp. parallel, a high, resp. a low, resistance is adjusted, representing a digital "0" or "1". Recently it was reported that by adjusting intermediate magnetization angles in the free layer 16 different states can be stored in one physical cell, enabling to realize multi-cell storages in MTJ technology [24].

The read latency and read energy of STT-RAM is expected to be comparable to that of SRAM. The expected 3x higher density and 7x less leakage power consumption in the STT-RAM makes it suitable for replacing SRAMs to build large NVMs. However, a write operation in an STT-RAM memory consumes 8x more energy and exhibits a 6x longer latency than a SRAM. Therefore, minimizing the impact of inefficient writes is critical for successful applications of STT-RAM [25].

NRAMs, a proprietary technology of Nantero, are a very prospective NVM technology in terms of manufacturing maturity, according to their developers. The NVMs are based on nano-electromechanical carbon nano tube switches (NEMS). In [27, 28] pinched hysteresis loops are shown for the current-voltage curve for such NEMS devices. Consequently, also NEMS and NRAMs are memristors according to Leon Chua's theory. The NRAM uses a fabric of carbon nanotubes (CNT) for saving bits. The resistive state of the CNT fabric determines, whether a one or a zero is saved in a memory cell. The resistance depends on the width of a bridge between two CNT. With the help of a small voltage, the CNTs can be brought into contact or be separated. Reading out a bit means to measure the resistance. Nantero claims that their technology features show the same read- and write latencies as DRAM, has a high endurance and reliability even in high temperature environments and is low power with essentially zero power consumption in standby mode. Furthermore NRAM is compatible with existing CMOS fabs without needing any new tools or processes, and it is scalable even to below 5 nm [29].

Fig. 2.4 gives an overview of some memristive devices' characteristics. This is just a snapshot of an assessment. Assessments of other authors differ widely in terms of better of worse values concerning different features. The International Technology Roadmap for Semiconductors (ITRS 2013) [30] reports an energy of operation of 6 pJ and projects 1 fJ for the year 2025 for PCMs. Jeong and Shi [31] report in 2019 an energy of operation of 80 fJ to 0.03 nJ for prototype and research PCM devices, 0.1 pJ to 10 nJ for RAM based devices, whereas the commercial OxRAM based ReRAMs from Panasonic have a write speed of 100 ns and an energy value of 50 pJ per memory cell. A record breaking energy efficiency is published in Vodenicarevic [32] for STT-MRAMs with 20 fJ/bit for a device area of 2 μm^2, compared to 3 pJ/bit and 4000 μm^2 for a state-of-the-art pure CMOS solution. The price for this perfect value is a limited speed dynamics of a few dozens MHz. However, for embedded IoT devices this can be sufficient. Despite of this distinguishing numbers it is clear that these devices offer a lot of potential and it is to expect that some of this potential can be exploited for future computer architectures.

The NVSim simulator [33] is popular in computer architecture science research to assess architectural structures based on the circuit-level performance, energy and area model of emerging non-volatile memories. It allows the investigation of architectural structures for future NVM based high-performance computers. Nevertheless, there is still a lot of work to do on the tool side. Better models for memristor technology, both physical and analytical ones, have to be integrated in the tools and besides that also the models themselves have to be fine tuned.

2.3 Multi-Level Cell Capability of Memristors

One of the most promising benefits that memristive technologies like ReRAM, PCMs, or STT-RAMs offer is their capability of storing more than two bits in one physical storage cell. MLC is necessary if memrisors are used to emulate synaptic plasticity [34]. Compared to conventional SRAM or DRAM storage technology this is an additional qualitative advantage to their feature of non-volatility. In literature this benefit is often denoted as multi-level-cell (MLC) or sometimes also as multi-bit capability. The different memristive

TABLE 1. DEVICE CHARACTERISTICS OF MAINSTREAM AND EMERGING MEMORY TECHNOLOGIES.

	MAINSTREAM MEMORIES				EMERGING MEMORIES		
			FLASH				
	SRAM	DRAM	NOR	NAND	STT-MRAM	PCRAM	RRAM
Cell area	>100 F^2	6 F^2	10 F^2	<4F^2 (3D)	6~50F^2	4~30F^2	4~12F^2
Multibit	1	1	2	3	1	2	2
Voltage	<1 V	<1 V	>10 V	>10 V	<1.5 V	<3 V	<3 V
Read time	~1 ns	~10 ns	~50 ns	~10 µs	<10 ns	<10 ns	<10 ns
Write time	~1 ns	~10 ns	10 µs–1 ms	100 µs–1 ms	<10 ns	~50 ns	<10 ns
Retention	N/A	~64 ms	>10 y	>10 y	>10 y	>10 y	>10 y
Endurance	>1E16	>1E16	>1E5	>1E4	>1E15	>1E9	>1E6~1E12
Write energy (J/bit)	~fJ	~10fJ	~100pJ	~10fJ	~0.1pJ	~10pJ	~0.1 pJ

Notes: F: feature size of the lithography. The energy estimation is on the cell-level (not on the array-level). PCRAM and RRAM can achieve less than 4F^2 through 3D integration. The numbers of this table are representative (not the best or the worst cases).

Figure 2.4: Snapshot of different memristive devices' characteristics, reprinted from S.Yu, P.-Y. Chen, Emerging Memory Technologies, 2016 [26]

technologies offer different benefits and drawbacks among each other concerning the realization of the MLC feature. Details about these benefits and drawbacks as well as the possibilities of usage of this MLC feature in future computing systems for caches, associative memories and ternary computing schemes can be found in Chapter 3.

2.4 Current State

The above mentioned memristor technologies PCM, ReRAMs, MRAM, STT-RAM, an advanced MRAM technology which uses a spin-polarized current instead of a magnetic field to store information in the electron's spin allowing therefore higher integration densities, and NRAM are among the most prominent memristor candidates, which are already commercialized or close to commercialization. Several more varieties exist, which are not mentioned here.

Intel and Micron already deliver the new 3D XPoint memory technology [35] as flash replacement which is based on PCM technology. Their Optane-SSDs 905P series is available on the market and offers 960 GByte for an about four times higher price than current 1 TByte SSD-NAND flash SSDs but provides 2.5 to 77 times better performance than NAND-SSDs. Intel and Micron expect that the X-Point technology could become the dominating technology as an alternative to RAM devices offering in addition NVM property in the next ten years. But the manufacturing process is complicated and currently, devices are expensive.

IBM published in 2016 achieved progress on a multi-level-cell (MLC-)PCM technology [36] replacing Flash and to use them e.g. as storage class memory (SCM) of supercomputers to fill the latency gap between DRAM main memory and the hard disk based background memory.

Adesto Technologies is offering CBRAM technology in their serial memory chips [37]. The company recently announced it will present new research showing the significant potential for Resistive RAM (RRAM) technology in high-reliability applications such as automotive. RRAM has great potential to become a widely used, low-cost and simple embedded non-volatile memory (NVM), as it utilizes simple cell structures and materials which can be integrated into existing manufacturing flows with as little as one additional mask. Adesto's RRAM technology (trademarked as CBRAM), making it a promising candidate for high-reliability applications. CBRAM consumes less power, requires fewer processing steps, and operates at lower voltages as compared to conventional embedded flash technologies [38].

MRAM is a NVM technology that is already available today, however in a niche market. MRAM chips are produced by Everspin Technologies, GlobalFoundries and Samsung [22].

Everspin delivered in 2017 samples of STT-MRAMs in perpendicular Magnetic Tunnel Junction Process (pMTJ) as 256-MBit-MRAMs und 1 GB-SSDs. Samsung is developing an MRAM technology. IBM and Samsung reported already in 2016 an MRAM device capable of

scaling down to 11 nm with a switching current of 7.5 microamps at 10 ns [22]. Samsung and TSMC are producing MRAM products in 2018.

Everspin offers in August 2018 a 256Mb ST-DDR3 STT-MRAM storage device designed for enterprise-style applications like SSD buffers, RAID buffers or synchronous logging applications where performance is critical and endurance is a must. The persistence of STT-MRAM protects data and enables systems to dramatically reduce latency, by up to 90%, boosting performance and driving both efficiency and cost savings [39]. Everspin is focusing with their MRAM products on areas where there is a need for fast, persistent memory by offering near-DRAM performance combined with non-volatility.

Right now, the price of MRAM is still rather high, but it is the most interesting emerging memory technology because its performance is close to SRAM and DRAM, and its endurance is very high. MRAM makes sense for cache buffering, and for specific applications, such as the nvNITRO NVMe storage accelerator for financial applications, where "doing a transaction quickly is important, but having a record is just as important" [40].

TSMC is also developing Embedded MRAM and Embedded ReRAM, as indicated by the TSMC roadmap in 2018 [41].

Nantero together with Fujitsu announced a Multi-GB-NRAM memory in Carbone-Nanotube-Technique expected for 2018. Having acquired the license to produce Nantero's NRAM (Nano-RAM), Fujitsu targets 2019 for NRAM Mass Production. Nantero's CNT-based devices can be fabricated on standard CMOS production equipment, which may keep costs down. NRAM could be Flash replacement, able to match the densities of current Flash memories and, theoretically, it could be made far denser than Flash.

Nantero also announced a multi-gigabyte DDR4-compatible MRAM memory with speed comparable to DRAM at a lower price per gigabyte. Cache, based on nonvolatile technology, will remove the need for battery backup. Nantero said that this allows for a dramatic expansion of cache size, substantially speeding up the SSD or HDD. Embedded memory will eventually be able to scale to 5nm in size (the most advanced semiconductors are being produced at the 10-nm and 7-nm nodes); operate at DRAM-like speeds, and operate at very high temperature, said Nantero. The company said that the embedded memory devices will be well-suited for several IoT applications, including automotive. [42]

Perspective

It is foreseeable, that memristor technologies will supersede current Flash memory. Memristors offer orders of magnitude faster read/write accesses and also much higher endurance. They are resistive switching memory technologies, and thus rely on different physics than that of storing charge on a capacitor as is the case for SRAM, DRAM and Flash. Some memristor technologies have been considered as a feasible replacement for SRAM [43, 44, 45]. Studies suggest that replacing SRAM with STT-RAM could save 60% of LLC energy with less than 2% performance degradation [43].

Besides the potential as memories, memristors which are complementary switches offer a highly promising approach to realize memory and logic functionality in a single device, e.g. for reconfigurable logics [16], and memristors with multi-level cell capabilities enable the emulation of synaptic plasticity [34] to realize neuromorphic computing, e.g. for machine learning with memristor-based neural networks.

One of the challenges for the next decade is the provision of appropriate interfacing circuits between the SCMs, or NVM technologies in general, and the microprocessor cores. One of the related challenges in this context is the developing of efficient interface circuits in such a way that this additional overhead will not corrupt the benefits of memristor devices in integration density, energy consumption and access times compared to conventional technologies.

STT-RAM devices primarily target the replacement of DRAM, e.g., in Last-Level Caches (LLC). However, the asymmetric read/write energy and latency of NVM technologies introduces new challenges in designing memory hierarchies. Spintronic allows integration of logic and storage at lower power consumption. Also new hybrid PCM / Flash SSD chips could emerge with a processor-internal last-level cache (STT-RAM), main processor memory (ReRAM, PCRAM), and storage class memory (PCM or other NVM).

All commercially available memristive memories feature better characteristics than Flash, however, are much more expensive. It is unclear when most of the new technologies will be mature enough and which of them will prevail by a competitive price. "It's a veritable zoo of technologies and we'll have to wait and see

which animals survive the evolutionary process," said Thomas Coughlin, founder of Coughlin Associates.

References

[1] Wikipedia. *Memristor*. URL: http://en.wikipedia.org/wiki/Memristor.

[2] A. L. Shimpi. *Samsung's V-NAND: Hitting the Reset Button on NAND Scaling*. AnandTech. 2013. URL: http://www.anandtech.com/show/7237/samsungs-vnand-hitting-the-reset-button-on-nand-scaling.

[3] L. Chua. "Memristor-The missing circuit element". In: *IEEE Transactions on Circuit Theory, Vol. 18, Iss. 5*. 1971.

[4] D. Strukov, G. Snider, D. Stewart, and R. Williams. "The missing memristor found". In: *Nature* 453 (2008).

[5] P. W. C. Ho, N. H. El-Hassan, T. N. Kumar, and H. A. F. Almurib. "PCM and Memristor based nanocrossbars". In: *2015 IEEE 15th International Conference on Nanotechnology (IEEE-NANO)*. IEEE. 2015. URL: https://ieeexplore.ieee.org/%20document/7388636/.

[6] B. C. 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". In: *IEEE micro* 30.1 (2010).

[7] C. Lam. "Cell Design Considerations for Phase Change Memory as a Universal Memory". In: *VLSI Technology, Systems and Applications*. IEEE. 2008, pp. 132–133.

[8] B. C. Lee, E. Ipek, O. Mutlu, and D. Burger. "Architecting Phase Change Memory As a Scalable Dram Alternative". In: *Proceedings of the 36th Annual International Symposium on Computer Architecture*. ISCA '09. 2009, pp. 2–13.

[9] M. K. Qureshi, V. Srinivasan, and J. A. Rivers. "Scalable High Performance Main Memory System Using Phase-change Memory Technology". In: *Proceedings of the 36th Annual International Symposium on Computer Architecture*. ISCA '09. 2009, pp. 24–33.

[10] P. Zhou, B. Zhao, J. Yang, and Y. Zhang. "A Durable and Energy Efficient Main Memory Using Phase Change Memory Technology". In: *Proceedings of the 36th Annual International Symposium on Computer Architecture*. ISCA '09. 2009, pp. 14–23.

[11] W. Wong. *Conductive Bridging RAM*. electronic design. 2014. URL: http://electronicdesign.com/memory/conductive-bridging-ram.

[12] C. Xu, D. Niu, N. Muralimanohar, R. Balasubramonian, T. Zhang, S. Yu, and Y. Xie. "Overcoming the Challenges of Crossbar Resistive Memory Architectures". In: *21st International Symposium on High Performance Computer Architecture (HPCA)*. IEEE. 2015, pp. 476–488.

[13] C. Xu, X. Dong, N. P. Jouppi, and Y. Xie. "Design Implications of Memristor-based RRAM Cross-point Structures". In: *Design, Automation & Test in Europe Conference & Exhibition (DATE), 2011*. IEEE. 2011, pp. 1–6.

[14] Y. Shuai et al. "Substrate effect on the resistive switching in BiFeO3 thin films". In: *Journal of Applied Physics* 111.7 (Apr. 2012), p. 07D906. DOI: 10.1063/1.3672840.

[15] T. You et al. "Bipolar Electric-Field Enhanced Trapping and Detrapping of Mobile Donors in BiFeO3 Memristors". In: *ACS Applied Materials & Interfaces* 6.22 (2014), pp. 19758–19765. DOI: 10.1021/am504871g.

[16] T. You et al. "Exploiting Memristive BiFeO3 Bilayer Structures for Compact Sequential Logics". In: *Advanced Functional Materials* 24.22 (2014), pp. 3357–3365. DOI: 10.1002/adfm.201303365.

[17] N. Du et al. "Field-Driven Hopping Transport of Oxygen Vacancies in Memristive Oxide Switches with Interface-Mediated Resistive Switching". In: *Phys. Rev. Applied* 10 (5 Nov. 2018), p. 054025. DOI: 10.1103/PhysRevApplied.10.054025.

[18] X. Ou et al. "Forming-Free Resistive Switching in Multiferroic BiFeO3 thin Films with Enhanced Nanoscale Shunts". In: *ACS Applied Materials & Interfaces* 5.23 (2013), pp. 12764–12771. DOI: 10.1021/am404144c.

[19] B. Govoreanu et al. "10x10nm 2 Hf/HfO x crossbar resistive RAM with excellent performance, reliability and low-energy operation". In: *International Electron Devices Meeting (IEDM)*. IEEE. 2011, pp. 31–6.

[20] Crossbar. *ReRAM Advantages*. URL: https://www.crossbar-inc.com/en/technology/reram-advantages/.

[21] Everspin. *MRAM Technology Attributes*. URL: www.everspin.com/mram-technology-attributes.

[22] Wikipedia. *Magnetoresistive random-access memory*. URL: http://en.wikipedia.org/wiki/Magnetoresistive_random-access_memory.

[23] D. Apalkov et al. "Spin-transfer Torque Magnetic Random Access Memory (STT-MRAM)". In: *ACM Journal on Emerging Technologies in Computing Systems (JETC)* 9.2 (2013), p. 13.

[24] *Spintronic Devices for Memristor Applications*. Talk at Meeting of EU COST ACTION MemoCis IC1401, "Memristors: at the crossroad of Devices and Applications", Milano, 28th March 2016. Mar. 2016.

[25] H. Noguchi et al. "A 250-MHz 256b-I/O 1-Mb STT-MRAM with Advanced Perpendicular MTJ Based Dual Cell for Nonvolatile Magnetic Caches to Reduce Active Power of Processors". In: *VLSI Technology (VLSIT), 2013 Symposium on*. IEEE. 2013, pp. 108–109.

[26] S.Yu and P.-Y. Chen. "Emerging Memory Technologies". In: *SPRING 2016 IEEE Solid-sate circuits magazine*. 2016.

[27] Z. F. ; X. F. ; A. L. ; L. Dong. "Resistive switching in copper oxide nanowire-based memristor". In: *12th IEEE International Conference on Nanotechnology (IEEE-NANO)*. 2012.

[28] Q. L. ; S.-M. K. ; C. A. R. ; M. D. E. ; J. E. Bon. "Precise Alignment of Single Nanowires and Fabrication of Nanoelectromechanical Switch and Other Test Structures". In: *IEEE Transactions on Nanotechnology, Vol. 6 , Iss. 2*. 2007.

[29] Nantero. *Nantero NRAM Advances in Nanotechnology*. URL: http://nantero.com/technology/.

[30] *International Technology Roadmap for Semiconductors (ITRS), Emerging research devices, 2013*. URL: http://www.itrs2.net/itrs-reports.html.

[31] H. Jeong and L. Shi. "Memristor devices for neural networks". In: *Journal of Physics D: Applied Physics* 52.2 (Jan. 2019), p. 023003. DOI: 10.1088/1361-6463/aae223.

[32] D. Vodenicarevic et al. "Low-Energy Truly Random Number Generation with Superparamagnetic Tunnel Junctions for Unconventional Computing". In: *Physical Review Applied* 8.5 (). DOI: 10.1103/PhysRevApplied.8.054045. URL: https://link.aps.org/doi/10.1103/PhysRevApplied.8.054045.

[33] X. Dong, C. Xu, N. Jouppi, and Y. Xie. "NVSim: A Circuit-Level Performance, Energy, and Area Model for Emerging Nonvolatile Memory". In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 31.7 (July 2012), pp. 994–1007. DOI: 10.1109/TCAD.2012.2185930.

[34] N. Du, M. Kiani, C. G. Mayr, T. You, D. Bürger, I. Skorupa, O. G. Schmidt, and H. Schmidt. "Single pairing spike-timing dependent plasticity in BiFeO3 memristors with a time window of 25 ms to 125 ms". In: *Frontiers in Neuroscience* 9 (June 2015), p. 227. DOI: 10.3389/fnins.2015.00227.

[35] J. Evangelho. *Intel And Micron Jointly Unveil Disruptive, Game-Changing 3D XPoint Memory, 1000x Faster Than NAND*. Hot Hardware. 2015. URL: https://hothardware.com/news/intel-and-micron-jointly-drop-disruptive-game-changing-3d-xpoint-cross-point-memory-1000x-faster-than-nand.

[36] N. Papandreou, H. Pozidis, T. Mittelholzer, G. Close, M. Breitwisch, C. Lam, and E. Eleftheriou. "Drift-tolerant Multilevel Phase-Change Memory". In: *3rd IEEE International Memory Workshop (IMW)*. 2011.

[37] Adesto Technologies. *Mayriq Products*. URL: http://www.adestotech.com/products/mavriq/.

[38] P. Release. *Adesto Demonstrates Resistive RAM Technology Targeting High-Reliability Applications such as Automotive*. URL: https://www.adestotech.com/news-detail/adesto-demonstrates-resistive-ram-technology-targeting-high-reliability-applications-such-as-automotive/.

[39] E. technologies. *STT-MRAM Products*. URL: https://www.everspin.com/stt-mram-products.

[40] G. Hilson. *Everspin Targets Niches for MRAM*. URL: https://www.eetimes.com/document.asp?doc_id=1332871.

[41] TSMC. *eFlash*. URL: http://www.tsmc.com/english/dedicatedFoundry/technology/eflash.htm.

[42] B. Santo. *NRAM products to come in 2019*. URL: https://www.electronicproducts.com/Digital_ICs/Memory/NRAM_products_to_come_in_2019.aspx.

[43] H. Noguchi, K. Ikegami, N. Shimomura, T. Tetsufumi, J. Ito, and S. Fujita. "Highly Reliable and Low-power Nonvolatile Cache Memory with Advanced Perpendicular STT-MRAM for High-performance CPU". In: *Symposium on VLSI Circuits Digest of Technical Papers*. IEEE. June 2014, pp. 1–2.

[44] H. Noguchi et al. "A 3.3ns-access-time 71.2 uW/MHz 1Mb Embedded STT-MRAM Using Physically Eliminated Read-disturb Scheme and Normally-off Memory Architecture". In: *International Solid-State Circuits Conference-(ISSCC)*. IEEE. 2015, pp. 1–3.

[45] J. Ahn, S. Yoo, and K. Choi. "DASCA: Dead Write Prediction Assisted STT-RAM Cache Architecture". In: *International Symposium on High Performance Computer Architecture (HPCA)*. IEEE. 2014, pp. 25–36.

3 Multi-Level Cell Capability of Memristors

One of the most promising benefits that memristive technologies like ReRAM, PCMs, or STT-RAMs offer is their capability of storing more than two bits in one physical storage cell. Compared to conventional SRAM or DRAM storage technology this is an additional qualitative advantage to their feature of non-volatility. In literature this benefit is often denoted as multi-level-cell (MLC) or sometimes also as multi-bit capability.

3.1 Multi-level-cell (MLC)

The different memristive technologies offer different benefits and drawbacks among each other concerning the realization of the MLC feature. E.g., one of the main challenges in MLC-PCM systems is the read reliability degradation due to resistance drift [1]. Resistance drift means that the different phase states in the used chalcogenide storage material can overlap since each reading step changes a little bit the phase what is not a real problem in single-level cells (SLC) but in MLCs. In a recently published work the impressive number of 47 distinct resistance levels was demonstrated for a so-called bi-layer ReRAM structure [2]. In such a bi-layer structure not only one metal-oxide layer is used as storage material, like e.g. usually HfO2 or TiO2 technology, which is enclosed between a metallic top and a bottom electrode. Moreover, a sequence of metal-oxide layers separated by an isolating layer is used leading to a better separation of different resistance levels for the prize of a much more difficult manufacturing process. Memristive MLC technique based on MRAM technology without spin-polarized electrons was proposed to store up to 8 different levels [3]. In STT-MRAM technology, using spin-polarized electrons, 2-bit cells are most common and were also physically demonstrated on layout level [4].

3.2 MLC as Memory

In its general SLC form STT-MRAM is heavily discussed as a candidate memory technology for near-term realization of future last-level-caches due to its high density characteristics and comparatively fast read/write access latencies. On academic site the next step is discussed how to profit from the MLC capability [5].

The last example, concerning MLC caches, is representative for all memristive NVM technologies and their MLC capability. It shows that the MLC feature are of interest for improving the performance of future computer or processor architectures. In this context they are closely related to future both near-memory and in-memory computing concepts for both future embedded HPC systems and embedded smart devices for IoT and CPS. For near-memory-computing architectures, e.g. as embedded memories, they can be used for a better high-performance multi-bit cache in which different tasks store their cached values in the same cache line.

Another more or less recent state-of-the-art application is their use in micro-controller units as energy-efficient, non-volatile check-pointing or normally-off/instant-on operation with near zero latency boot as it was just announced by the French company eVaderis SA [6].

To this context also belongs research work on ternary content-addressable memories (TCAM) with memristive devices, in which the third state is used for the realization of the don't care state in TCAMs. In many papers, e.g. in [7], is shown that using memristive TCAMs need less energy, less area than equivalent CMOS TCAMs. However, most of the proposed memristive TCAM approaches don't exploit the MLC capability. They are using three memristors to store 1, 0, and X (don't care). In a next step this can be expanded to exploit the MLC capability of such devices for a further energy and area improvement.

3.3 Ternary Arithmetic Based on Signed-Digit (SD) Number Systems

Another promising aspect of the MLC capability of memristive devices is to use them in ternary arithmetic circuits or processors based on signed-digit (SD) number systems. In a SD number system a digit can

have also a positive and a negative value, e.g. for the ternary case we have not given a bit but a trit with the values, -1, 0, and +1. It is long known that ternary or redundant number systems generally, in which more than two states per digit are mandatory, improve the effort of an addition to a complexity of O(1) compared to log(N) which can be achieved in the best case with pure binary adders. In the past conventional computer architectures did not exploit this advantage of signed-digit addition. One exception was the compute unit of the ILLIAC III [8] computer manufactured in the 1960's, at a time when the technology was not so mature than today and it was necessary to achieve high compute speeds with a superior arithmetic concept even for paying a price of doubling the memory requirements to store a ternary value in two physical memory cells. In course of the further development the technology and pipeline processing offering latency hiding, the ALUs become faster and faster and it was not acceptable for storing operands given in larger than binary redundant representation. This would double the number of registers, double the size of the data cache and double the necessary size of data segments in main memory. However, with the occurrence of CMOS-compatible NVM technology offering MLC capability the situation changed. This calls for a re-evaluation of these redundant computer arithmetic schemes under a detailed consideration of the technology of MLC NVM.

3.4 Perspectives and Research Challenges

Different work has already investigated the principal possibilities of ternary coding schemes using MLC memristive memories. This was carried out both for hybrid solutions, i.e. memristors are used as ternary memory cells for digital CMOS based logic circuits [9], [10], and in proposals for in-memory computing like architectures, in which the memristive memory cell was used simultaneously as storage and as logical processing element as part of a resistor network with dynamically changing resistances [11]. The goal of this work using MLC NVM technology for ternary processing is not only to save latency but also to save energy since the number of elementary compute steps is reduced compared to conventional arithmetic implemented in state-of-the-art processors. This reduced number of processing steps should also lead to reduced energy needs.

As own so far unpublished work, carried out in the group of the author of this chapter, shows that in CMOS combinatorial processing, i.e. without storing the results, the energy consumption could be reduced about 30 % using a ternary adder compared to the best parallel pre-fix binary adders for a 45 nm CMOS process. This advantage is lost if the results are stored in binary registers. To keep this advantage and exploit it in IoT and embedded devices, which are "energy-sensible" in particular, in future ternary storage and compute schemes based on MLC based NVMs have to be integrated in near- and in-memory computing schemes.

To achieve this goal, research work is necessary on following topics: (i) design tools, considering automatic integration and evaluation of NVMs in CMOS, what (ii) requires the development of appropriate physical models not only on analogue layer but also on logic and RTL level, (iii) appropriate interface circuitry for addressing NVMs, and (iv) in general the next step that has to be made is going from existing concepts and demonstrated single devices to real systems.

References

[1] W. Zhang and T. Li. "Helmet: A resistance drift resilient architecture for multi-level cell phase change memory system". In: *IEEE/IFIP 41st International Conference on Dependable Systems & Networks (DSN)*. 2011, pp. 197–208.

[2] S. Stathopoulos et al. "Multibit memory operation of metal-oxide bi-layer memristors". In: *Scientific Reports* 7 (2017), p. 17532.

[3] H. Cramman, D. S. Eastwood, J. A. King, and D. Atkinson. "Multilevel 3 Bit-per-cell Magnetic Random Access Memory Concepts and Their Associated Control Circuit Architectures". In: *IEEE Transactions on Nanotechnology* 11 (2012), pp. 63–70.

[4] L. Xue et al. "An Adaptive 3T-3MTJ Memory Cell Design for STT-MRAM-Based LLCs". In: *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 99 (2018), pp. 1–12.

[5] L. Liu, P. Chi, S. Li, Y. Cheng, and Y. Xie. "Building energy-efficient multi-level cell STT-RAM caches with data compression". In: *2017 22nd Asia and South Pacific Design Automation Conference (ASP-DAC)*. Jan. 2017, pp. 751–756. DOI: 10.1109/ASPDAC.2017.7858414.

[6] *Startup tapes out MRAM-based MCU*. 2018. URL: http://www.eenewsanalog.com/news/startup-tapes-out-mram-based-mcu.

[7] Q. Guo et al. "A Resistive TCAM Accelerator for Data-Intensive Computing". In: *MICRO'11*.

[8] D. E. Atkins. "Design of the Arithmetic Units of ILLIAC III: Use of Redundancy and Higher Radix Methods". In: *IEEE Transactions on Computers* C-19 (1970), pp. 720–722.

[9] D. Fey, M. Reichenbach, C. Söll, M. Biglari, J. Röber, and R. Weigel. "Using Memristor Technology for Multi-value Registers in Signed-digit Arithmetic Circuits". In: *MEMSYS 2016*. 2016, pp. 442–454.

[10] D. Wust, D. Fey, and J. Knödtel. “A programmable ternary CPU using hybrid CMOS memristor circuits”. In: *Int. JPDES* (2018). DOI: https://doi.org/10.1080/17445760.2017.1422251.

[11] A. A. El-Slehdar, A. H. Fouad, and A. G. Radwan. “Memristor-based redundant binary adder”. In: *International Conference on Engineering and Technology (ICET)*. 2014, pp. 1–5.

4 Memory Hierarchies in Systems with NV Memories

4.1 Situation

The Von Neumann architecture assumes the use of central execution units that interface with memory hierarchies of several layers. This model serves as the execution model for more than five decades. Locality of references is a central assumption of the way we design systems. The consequence of this assumption is the need of hierarchically arranged memories.

The memory hierarchy of modern computer systems (PCs and servers) typically consists of three disjoint layer types. Closest to the processor, we find the cache hierarchy layers that are based on SRAM cells located on the processor chip. Below the cache layer, we find the main memory layer that usually consists of DRAM cells. Eventually, the last layer of the memory hierarchy represents the non-volatile mass storage. Traditionally, this layer was realized using magnetic disk drives. In recent years these drives have been replaced by solid state drives which use Flash memory to store the data. Memory is accessed by linear adresses in chunks of word or cache-line size, mass storage as files.

But this model of a memory hierarchy is not effective in terms of performance for a given power envelop. The main source of inefficiency in the meantime became data movement: the energy cost of fetching a word of data from off-chip DRAM is up to 6400 times higher than operating on it [1]. The current consequence is to move the RAM memory closer to the processor by providing High-Bandwidth Memories.

4.2 High-Bandwidth Memory (HBM)

A state-of-the-art memory hierarchy for server-class of computers contains *High-Bandwidth Memory (HBM)* [2] (see Fig. 4.1), which provides higher memory-bandwidths to the cores. Memory is connected in a HBM system via an interposer in the same package with the processor. Memory chips are vertically stacked and connected by TSVs (Through-Silicon Via) with an access logic chip that serves the memory requests of the processor.

HBM provides a tight 3D integration of DRAM memory modules to reduce latency and to increase bandwidth by reducing the energy costs for the data transfer simultaneously.

HBM is based on Die Stacking, which denoteis the concept of stacking integrated circuits (e.g. processors and memories) vertically in multiple layers. Die stacking diminishes wire length between memory and logic chips and is applied to three-dimensional DRAM memories, where the bottom layer is active and hosts the physical interface of the memory to the external system. NVIDIA, AMD and Intel already apply HBM to exploit the high-bandwidth and low latencies given by 3D stacked memories for a high-dense memory architecture.

3D stacking also enables heterogeneity, by integrating layers, manufactured in different processes, e.g., memristor technologies, which would be incompatible among each other in monolithic circuits. Power consumption is reduced because of the short wire lengths of TSVs and interposers. Simultaneously, a high communication bandwidth between layers can be expected leading to particularly high processor-to-memory bandwidth.

4.3 Storage-Class Memory (SCM)

Storage-Class Memory (SCM) currently fills the latency gap between fast and volatile RAM-based memory and slow, but non-volatile disk storage in supercomputers. It is currently filled by Flash storage, but could in future be extended by memristive NVM with access times, which are much closer to RAM access times as Flash technology.

In that case memristive NVM based SCM could blur the distinction between memory and storage and require new data access modes and protocols that serve both “memory” and “storage”. These new SCM types of non-volatile memory could even be integrated on-chip with the microprocessor cores as they use CMOS-compatible sets of materials and require different device fabrication techniques from Flash. In a VLSI post-processing step they can be integrated on top of the

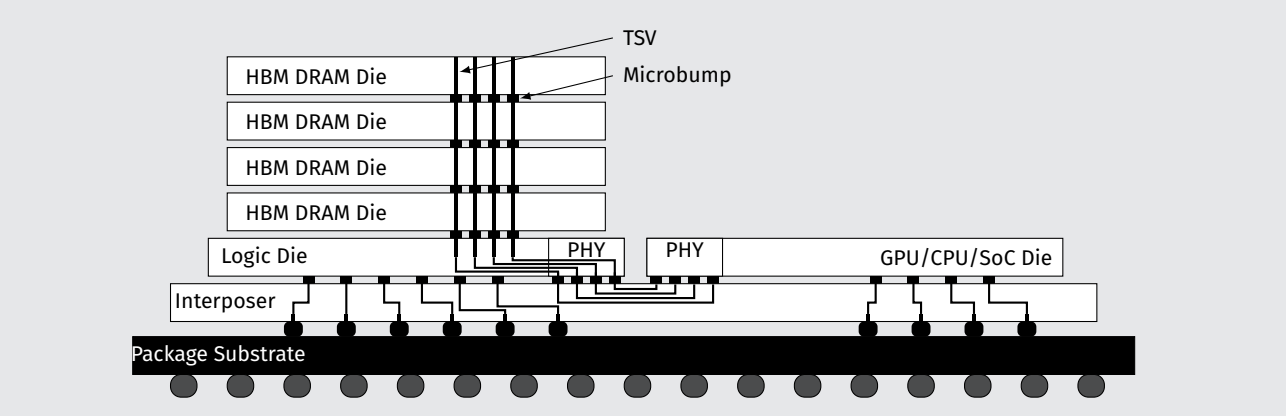


Figure 4.1: High Bandwidth Memory utilizing an active silicon Interposer [2]

last metal layer, which is often denoted as a back-end of line (BEOL) step.

4.4 Potential Memory Hierarchy of a Future Supercomputer

Low-speed non-volatile memories might lead to additional levels in the memory hierarchy to efficiently close the gap between mass-storage and memory as demonstrated by Fig. 4.2 for a potential memory hierarchy of a future supercomputer. Memristors as new types of NV memories can be used in different layers of the memory hierarchy not only in supercomputers but in all kinds of computing devices. Depending on which memory technologies mature, this can have different impacts. Fast non-volatile memories (e.g. STT-RAM) offer the opportunity of merging cache and memory levels. Mid-speed NV memories (e.g. PCM) could be used to merge memory and storage levels.

4.5 Potential Memory Hierarchy of Future Embedded Systems

On the other hand, the memory hierarchy might become flatter by merging main memory with storage in particular for smaller systems. Such a shallow memory hierarchy might be useful for future embedded systems. The cache, main memory and mass storage level might be merged to a single level, as shown in Figure 4.3. As a result, the whole system would provide an improved performance, especially in terms of real-time operation. An increased resistance against radiation effects (e.g. bit flips) would be another positive effect. Also, a shallow memory hierarchy would enable applications to use more non-uniform or highly random data access.

4.6 Implications

Merging main memory and mass storage allows applications to start much faster. It might be helpful for crash recovery and it can reduce energy consumption as it takes less time to activate/deactivate applications.

Programs can be run in intermittent operation (being active for short periods and then stay calm for longer periods) without large overhead. Also, the whole system might be put into standby in a very short time. E.g. if the 2^{nd} / 3^{rd} level of cache is built from NV memory, the processor only needs to purge the 1^{st} or 2^{nd} level of the cache and then the system can be shut off.

However, this also implies security issues. If data in cache is not lost on power down, this could be exploited to retrieve sensible data from the system.

Also, other problems have to be considered. Realizing such merged levels by NV memory technology might increase the cost to a point where it becomes no longer economically justifiable. The tradeoff between cost and performance has to be well evaluated. The durability and reliability of NV technologies might raise additional problems.

On the other hand, fault tolerance could also be improved by new NV memory concepts. Non volatile memory significantly simplifies checkpointing. If an error is detected, a valid state saved in NM memory could be easily retrieved. Checkpointing could be done on a very fine-grain level. Using so-called in-memory checkpointing, the checkpoint replication would be done automatically for memory to memory operations.

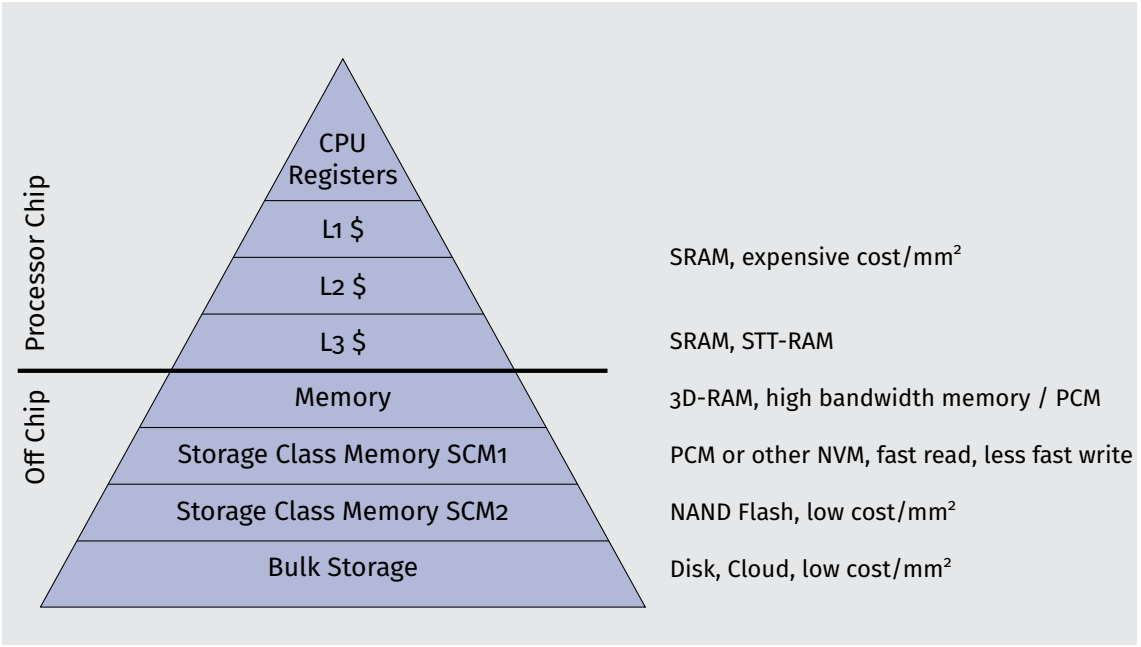


Figure 4.2: Usage of NVM in a future complex supercomputer memory hierarchy.

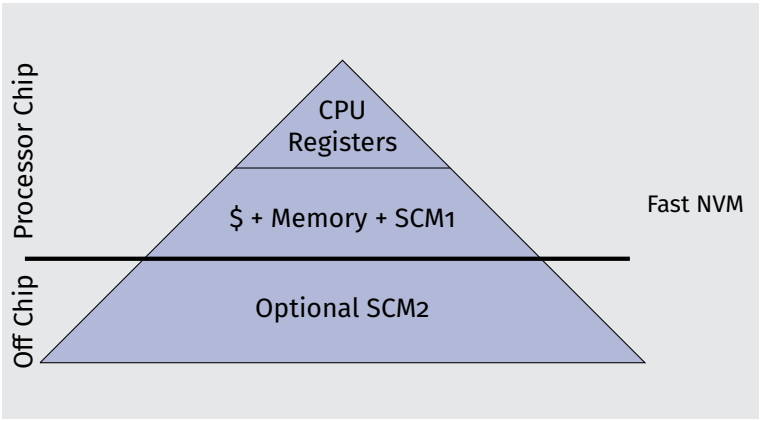


Figure 4.3: Usage of NVM in a future embedded systems shallow memory hierarchy.

4.7 Research Challenges

From the viewpoint of operating systems, modifying the memory hierarchy in one or the other way changes the design of memory maps. Code and data might be shared between processes at different hierarchy levels and might be moved in different ways between levels. Therefore, the operating systems will need to provide new models for program execution and data exchange.

With data stored in non-volatile memory, applications can be active for an arbitrary time. Thus, operating systems must provide different ways to start/stop/deactivate/reactivate and secure programs.

From the viewpoint of computer architecture, changing the memory map has also strong implications for the design of distributed and multi-/many-core systems. Hardware support for memory management in the processor architecture might have to be reconsidered. Different endurance capabilities of different memory levels might demand for new cache and virtual memory replacement strategies. Memory coherence protocols will also be affected. Overall, cache-, memory- and storage interactions on the hardware and OS level will offer research opportunities.

From the viewpoint of application programming, changes in memory hierarchy can modify application models and might improve the behavior of some application classes. If e.g. memory and storage levels are merged, information retrieval applications (e.g. database systems) do no longer need to think in separate categories for storing and memorizing data.. Nev-

ertheless, this will require different structuring of data than usual today.

References

[1] M. Drumond, A. Daglis, N. Mirzadeh, D. Ustiugov, J. Picorel, B. Falsafi, B. Grot, and D. Pnevmatikatos. "The Mondrian Data Engine". In: *Proceedings of the 44th Annual International Symposium on Computer Architecture*. ACM. 2017, pp. 639–651.

[2] AMD. *High Bandwidth Memory*. 2017. URL: `https://www.amd.com/Documents/High-Bandwidth-Memory-HBM.pdf`.

5 Impact of Memristors on Security and Privacy

This section discusses security and privacy implications of memristive technologies, including emerging memristive non-volatile memories (NVMs). The central property that differentiates such memories from conventional SRAM and DRAM is their non-volatility; therefore, we refer to these memories as "NVMs".

We cover potential inherent security risks, which arise from these emerging memory technologies and on the positive side security potentials in systems and applications that incorporate emerging NVMs. Further, we also consider the impact of these new memory technologies on privacy.

5.1 Background

The relevance of security and privacy has steadily increased over the years. This concerns from highly complex cyber-physical infrastructures and systems-of-systems to small Internet of Things (IoT) devices if they are applied for security critical applications [1]. A number of recent successful attacks on embedded and cyber-physical systems has drawn the interest not only of scientists, designers and evaluators but also of the legislator and of the general public. Just a few examples are attacks on online banking systems [2] and malware, in particular ransomware [3] and spectacular cyber attacks to critical infrastructures, as the Stuxnet attack [4], attacks on an industrial installation in German steel works [5] and on a driving Jeep car [6], to name but a few. Meanwhile entire botnets consisting of IoT devices exist [7]. These examples may shed light on present and future threats to modern IT systems, including embedded devices, vehicles, industrial sites, public infrastructure, and cyber-physical installations. Consequently, security and privacy may determine the future market acceptance of several classes of products, especially if they are increasingly enforced by national and EU-wide legislation [8]. Consequently, security and privacy should be considered together with (and in certain cases weighted against) the more traditional system attributes such as latency, throughput, energy efficiency, reliability, or cost.

Historically, the networks connecting the system with the outside world and the software running on the system's components were considered as a source of security weaknesses, giving rise to the terms "network security" and "software security" [9]. However, the system's hardware components are increasingly shifting into the focus of attention, becoming the Achilles' heels of systems. Researchers have been pointing to hardware-related vulnerabilities since long times, including side channels [10], fault-injection attacks [11], counterfeiting [12], covert channels [13] and hardware Trojans [14]. Several potential weaknesses in hardware components were exposed; some of the widely publicized examples were: counterfeit circuits in missile-defense installations in 2011 [15], potential backdoors in FPGAs (later identified as undocumented test access circuitry [16]) in 2012 [17], (hypothetical) stealthy manipulations in a microprocessor's secure random number generator in 2013 [18]. Very recently, two hardware-related security breaches, Meltdown [19] and Spectre [20] were presented. They exploit advanced architectural features of modern microprocessors and affect several microprocessors that are in use today.

Meltdown and Spectre are indicative of hardware-based attacks on PCs: On the one hand, it is difficult for an attacker to find such weaknesses (compared to many conventional methods, from social engineering to malware and viruses), and even when the weaknesses are known it may be difficult to develop and mount concrete attacks. On the other hand, once such an attack has been found, it affects a huge population of devices. It is also extremely difficult or may even be impossible to counteract because hardware cannot easily be patched or updated in field. Corrective actions, which require the replacement of the affected hardware components by (to be produced) secure versions are usually extremely costly and may even be infeasible in practice. Healing the problem by patching the software that runs on the component is not always effective and is often associated with a barely acceptable performance penalty [19]. Consequently, new architectural features should undergo a thorough security analysis before being used in mass products.

In this chapter, we consider potential implications of emerging memristors, and in particular memristive non-volatile memories (NVMs) and NVM-based computer architectures on security and privacy of systems (compared to conventional memory architectures). The discussion of emerging NVM technologies and architectures themselves, as well as their advantages and disadvantages compared to other concepts, are not in our scope; please refer to the other chapters for this information. We will discuss both: the vulnerabilities of systems due to integration of emerging NVMs, and the potential of NVMs to provide new security functions and features.

5.2 Memristors and Emerging NVMs: Security Risks

The crucial property of NVMs is – rather expected – their non-volatility: An NVM retains the stored information even when it is suddenly disconnected from the power supply. The first obvious consequence is the *persistency of attacks:* if the adversary managed to place malicious content (e.g., software code or manipulated parameter values) into a device's main memory, this content will not disappear by rebooting the device or powering it off. (Of course, to get rid of the malware usually additional security measures are necessary.) This is in stark contrast to volatile memories where reboot and power-off are viable ways to "heal" at least the volatile memory of an attacked system; the same system with an NVM will stay infected.

The non-volatility can simplify *read-out attacks* on unencrypted memory content. In such attacks, sensitive data (e.g., passwords or cryptographic keys) within an electronic component are accessed by an adversary with physical access to the device using either direct read-out (for smart cards: probing by electrical needles while the device is in operation or the use of an electron microscope) or using side-channels, e.g., measuring data-dependent power consumption or electromagnetic emanations. Usually, volatile memory must be read out in the running system, with the full system speed (which can be in Gigahertz range); moreover the system may be equipped with countermeasures, e.g., tamper-detectors which would delete the memory content once they identify the attempted attack. An exception are so-called cold boot attacks where the memory content may persist for several minutes or even hours [21]. An attacker who powered off a system with sensitive data in an NVM can analyze the NVM block offline. It is even possible to physically detach it from the system (and all countermeasures).

It is currently not clear whether emerging memristive NVMs bear *new side-channel vulnerabilities.* For example, many security architectures are based on encrypting sensitive information and overwriting the original data in the memory by an encrypted version or randomness. It is presently not clear whether memristive elements within NVMs exhibit a certain extent of "hysteresis", which may allow the adversary to reconstruct the state, which a memory cell had before the last writing operation with some degree of accuracy. This property was discussed in [22] from the forensic point of view. Whether this vulnerability indeed exists, must be established for each individual NVM technology (like STT-RAM or Resistive RAM (ReRAM)) by physical experiments. If it exists this might allow or at least support side-channel attacks. In fact, also conventional persistent memory (e.g., a hard disk) should completely be erased to save sensitive data, when its owner wants to give it back or to sell a hard disk. Special deletion tools exist, which overwrite the memory content by random data multiply to compensate technical inaccuracies within the writing process [23]. The situation there yet is more convenient since the erasing process does not need to be carried out while security critical processes run on the device.

First thoughts whether emerging NVMs would have impact on the vulnerability against implementation attacks can be found in [24]. The attack scenarios mentioned therein are typically counted as fault attacks and probing attacks. (In the field of implementation attacks the nomenclature is not always unique.) The authors conclude that ReRAMs would prevent these attacks. In [24] experiments were not conducted but the authors announce future experiments. To our knowledge typical side-channel attacks (power, timing, cache etc.) have not been considered so far in the context of NVMs.

Some of the memristive NVMs are also prone to *active manipulations, enabling fault attacks.* For example, the recent paper [25] considers non-invasive magnetic field attacks on STT-RAMs, where the adversary overrides the values of the cell by applying either a static or an alternating magnetic field. The authors of [25] note that this attack can be mounted on a running system or in passive mode, where it could, e.g., compromise the boot process.

While all of the mentioned attack scenarios can have severe consequences already against an adversary

who has physical access to the end-product, they may be even more dangerous if an attacker manages to compromise the system design and the manufacturing process, and was able to insert a *Hardware Trojan* into the circuitry. Trojans can be inserted during semiconductor manufacturing [26], they can be lurking in third-party intellectual-property cores [27], and even CAD tools used for circuit design may plant Trojans [28]. Emerging NVMs might facilitate both the establishment of Trojans in the system (e.g., by placing their trigger sequences in a non-volatile instruction cache) and also multiply the damaging potential of Trojans.

5.3 Memristors and Emerging NVMs: Supporting Security

On the positive side, memristors can be the basis for security primitives that are difficult or expensive to realize technically by conventional hardware and software. Depending on the scenario one such primitive might a *random number generator (RNG),* which is useful, for instance, for on-chip generation of secure cryptographic keys, signature parameters, nonces and for creating masks to protect cryptographic cores against side-channel analysis. Roughly speaking, Random-Number-Generators (RNGs) can be divided into deterministic RNGs (DRNGs) (a.k.a. pseudorandom number generators) and true RNGs. The class of true RNGs can further be subdivided into physical RNGs (PTRNGs, using dedicated hardware) and non-physical trueRNGs (NPTRNGs) [29]. Memristors and NVMs on their basis might be beneficial for both DRNGs and true RNGs. For DRNGs, NVMs might be used to store the internal state, thus reducing the need for additional non-volatile memory, saving the copy process of the internal state to non-volatile memory, or reseeding upon each power-on. This feature might in particular be important for resource-constrained IoT devices. Of course, such NVM cells must be secure against read-out and manipulation since otherwise an attacker might be able to predict all future random numbers. In TRNGs, memristors might serve as sources of entropy (see e.g. [30] and [31]), providing sources for physical RNGs or for non-physical non-deterministic RNGs as Linux /dev/random, for instance. Whether this use is realistic depends on the outcome of physical experiments for individual memristive technologies. To this end, suitable random parameters (e.g., the duration of the transition between stable states) must be identified; then, a stochastic model (for PTRNGs) or at least a reliable lower entropy bound per random bit (for NPTRNGs) must be established and validated, and finally the entropy per bit must be estimated [32]; see also [33, 34, 35]. In [30] and [31] the authors focus only on the statistical properties of the generated random numbers, which are verified by NIST randomness tests.

Another possible memristor-enabled security primitive could be a Physically Unclonable Function (PUF). A PUF is a "fingerprint" of an individual circuit instance among a population of manufactured circuits [36]. It should reliably generate a unique, circuit-specific bitstring, and it shall be impossible to produce another circuit with the same fingerprint. PUFs are used for on-chip generation of secret keys and for authentication protocols, for instance, but also for tracking circuits and preventing their counterfeiting [37]. PUFs based on memory cells are well-known [38], and these insights can perhaps directly be applied to emerging NVMs [39]. However, the emerging near-memory and in-memory concepts where NVMs are tightly coupled with logic, create potentials for richer varieties of PUF behavior, such as "strong PUFs" which support challenge-response authentication protocols [40]. A strong PUF proposal based on memristive elements has been proposed in [41]. Moreover, it was suggested to leverage non-linearity of memristors to define "public PUFs" which overcome certain deficiencies of traditional PUFs [42]. PUFs can protect not only circuits in which they are integrated, but also other objects to which these circuits are attached, such as valuable medicines prone to counterfeiting.

An interesting question might be whether emerging memristive cells and NVM-enabled architectures are better or worse protected against *counterfeiting and reverse engineering* compared to conventional circuits. On the one hand, the designer can replace identifiable circuit structures by a regular fabric similar to reconfigurable gate-arrays that is controlled by values stored in an NVM. This makes it difficult for an attacker to apply the usual flow to reconstruct the circuit functionality: depackage the circuit, extract its individual layers, and apply optical recognition to find logic gates, memory cells, interconnects, and other structures. In fact, if the content of the "configuration" NVM cells is lost during deprocessing, its functionality is irretrievably lost as well. Possibly, attackers may find ways to read out the values in memristive elements prior to deprocessing. In addition, memristors can power anti-counterfeiting solutions, like PUFs. As with other security attributes, the resistance of cir-

cuits to reverse engineering is a cat-and-mouse game where the defender invents new protections and the attacker finds way around this protection; NVMs could substantially change the rules of this game.

5.4 Memristors, Emerging NVMs and Privacy

Privacy stands in a non-trivial relationship with security, and therefore security implications of memristors can have positive or negative consequences for privacy [43]. On the one hand, security breaches that lead to unauthorized access to user data (e.g., leaked secret keys), or compromise their authenticity and integrity, are clearly detrimental for privacy (loss of privacy or of availability). If the personal data of a user is stored in an encrypted memory (and especially in an NVM) a compromised secret key will destroy privacy. To this end, all properties of NVMs that simplify attacks on encryption negative privacy impact, and all beneficial features of NVMs, e.g., schemes (e.g., read-out attacks or new side-channel attacks) have generation of secure secret keys, have positive consequences. Here, security and privacy requirements are consistent.

Security and privacy may get in conflict when it comes to methods which track in an undesired and unnecessary way individual circuit instances, e.g., by storing a unique identifier in an on-chip NVM, or by creating such an identifier using a PUF. This functionality is beneficial for security and in particular to prevent counterfeiting or overbuilding [37]. The flip side of the coin is the close connection between the circuit and its human user. Whoever possesses the history of the circuit being used can reconstruct, to some extent, behavioral patterns of its owner and thus violate his or her privacy. Possessing these data for a whole population of widely used circuits (e.g., ones built into a popular smartphone model) may allow undesired inferences about behavior of large subgroups of the society.

5.5 Conclusions and Recommendations

Security and privacy should be essential design targets for a steadily increasing number of applications, and hardware components play a crucial role in this context. The emergence and propagation of memristors, and in particular memristive NVMs in main memories and caches of computational devices may change established assumptions on their security and privacy properties. It is important to consider security and privacy already in the system conceptualization and design phase as it becomes increasingly difficult to "add security" to a system that has been created without considering such aspects.

A particular need are new possibilities for secure erasure of sensitive data when the device is in operation, which could circumvent the non-volatility of information in cases when it is undesirable. The secure erasure function should be supported on the hardware level that, e.g., overwrites the data designated for deletion (possibly several times) by random data. This problem occurs today when hard disks with sensitive data are reused (e.g., sold), but if large parts of the system's memories and caches become non-volatile, the secure erasure would resolve many of the security vulnerabilities mentioned in this chapter. Moreover, a better understanding of the new memory technologies might be useful for the design of RNGs.

References

[1] F. Regazzoni and I. Polian. "Securing the hardware of cyber-physical systems". In: *2017 22nd Asia and South Pacific Design Automation Conference (ASP-DAC)*. 2017, pp. 194–199.

[2] BSI für Bürger. *Online-Banking*. https://www.bsi-fuer-buerger.de/BSIFB/DE/DigitaleGesellschaft/OnlineBanking/onlinebanking_node.html.

[3] Bundesamt für Sicherheit in der Informationstechnik (BSI). *Ransomware - Bedrohungslage, Prävention & Reaktion*. https://www.bsi.bund.de/DE/Themen/Cyber-Sicherheit/Empfehlungen/Ransomware/Ransomware.pdf. Mar. 2016.

[4] R. Langner. "Stuxnet: Dissecting a Cyberwarfare Weapon". In: *IEEE Security & Privacy* 9.3 (2011), pp. 49–51.

[5] BBC News. *Hack attack causes 'massive damage' at steel works*. URL: http://www.bbc.com/news/technology-30575104.

[6] A. Greenberg. *Hackers remotely kill a Jeep on the highway—With me in it*. Wired. 2015. URL: https://www.wired.com/2015/07/hackers-remotely-kill-jeep-highway/.

[7] K. on Security. *Source code for IoT botnet 'Mirai' released*. 2016. URL: https://krebsonsecurity.com/2016/10/source-code-for-iot-botnet-mirai-released/.

[8] *The EU General Data Protection Regulation (GDPR) Portal*. URL: https://www.eugdpr.org/.

[9] C. Eckert. *IT-Sicherheit - Konzepte, Verfahren, Protokolle*. 6th ed. 2009.

[10] S. Mangard, E. Oswald, and T. Popp. *Power analysis attacks – Revealing the secrets of smart cards*. 2007.

[11] A. Barenghi, L. Breveglieri, I. Koren, and D. Naccache. "Fault injection attacks on cryptographic devices: Theory, practice, and countermeasures". In: *Proceedings of the IEEE* 100.11 (2012), pp. 3056–3076.

[12] F. Koushanfar, S. Fazzari, C. McCants, W. Bryson, M. Sale, and M. P. P. Song. "Can EDA combat the rise of electronic counterfeiting?" In: *DAC Design Automation Conference 2012*. 2012, pp. 133–138.

[13] Z. Wang and R. B. Lee. "Covert and side channels due to processor architecture". In: *ASAC*. 2006, pp. 473–482.

[14] S. Bhunia, M. S. Hsiao, M. Banga, and S. Narasimhan. "Hardware Trojan attacks: Threat analysis and countermeasures". In: *Proceedings of the IEEE* 102.8 (2014), pp. 1229–1247.

[15] D. Lim. *Counterfeit Chips Plague U.S. Missile Defense*. Wired. 2011. URL: https://www.wired.com/2011/11/counterfeit-missile-defense/.

[16] S. Skorobogatov. *Latest news on my Hardware Security Research*. URL: http://www.cl.cam.ac.uk/~sps32/sec_news.html.

[17] S. Skorobogatov and C. Woods. "Breakthrough silicon scanning discovers backdoor in military chip". In: *Cryptographic Hardware and Embedded Systems – CHES 2012*. 2012, pp. 23–40.

[18] G. T. Becker, F. Regazzoni, C. Paar, and W. P. Burleson. "Stealthy dopant-level hardware Trojans". In: *Cryptographic Hardware and Embedded Systems - CHES 2013*. 2013, pp. 197–214.

[19] M. Lipp et al. *Meltdown*. 2018. URL: https://meltdownattack.com.

[20] P. Kocher et al. *Spectre attacks: Exploiting speculative execution*. 2018. URL: https://meltdownattack.com.

[21] A. Halderman et al. "Lest we remember: Cold boot attacks on encryption keys". In: *17th USENIX Security Symposium*. 2008, pp. 45–60.

[22] J. Rajendran, R. Karri, J. B. Wendt, M. Potkonjak, N. McDonald, G. S. Rose, and B. Wysocki. "Nano meets security: Exploring nanoelectronic devices for security applications". In: *Proceedings of the IEEE* 103.5 (2015), pp. 829–849.

[23] BSI für Bürger. *Daten auf Festplatten richtig löschen*. https://www.bsi-fuer-buerger.de/BSIFB/DE/Empfehlungen/RichtigLoeschen/richtigloeschen_node.html.

[24] Z. Dyka, C. Walczyk, D. Walczyk, C. Wenger, and P. Langendörfer. "Side channel attacks and the non volatile memory of the future". In: CASES. 2012, pp. 13–16.

[25] A. De, M. N. I. Khan, J. Park, and S.Ghoosh. "Replacing eFlash with STTRAM in IoTs: Security challenges and solutions". In: *Journal of Hardware and Systems Security* 1.4 (2017), pp. 328–339.

[26] R. Kumar, P. Jovanovic, W. P. Burleson, and I. Polian. "Parametric Trojans for fault-injection attacks on cryptographic hardware". In: *2014 Workshop on Fault Diagnosis and Tolerance in Cryptography*. 2014, pp. 18–28.

[27] I. Polian, G. T. Becker, and F. Regazzoni. *Trojans in early design steps—An emerging threat*. TRUDEVICE Conf. 2016. URL: http://upcommons.upc.edu/handle/2117/99414.

[28] M. Potkonjak. "Synthesis of trustable ICs using untrusted CAD tools". In: *Design Automation Conference*. 2010, pp. 633–634.

[29] W. Schindler. "Random number generators for cryptographic applications". In: *Cryptographic Engineering*. 2009, pp. 5–23.

[30] C. Huang, W. Shen, Y. Tseng, C. King, and Y. C. Lin. "A Contact-resistive random-access-memory-based true random number generator". In: *IEEE Electron Device Letters* 33.8 (2012), pp. 1108–1110.

[31] H. Jiang et al. "A novel true random number generator based on a stochastic diffusive memristor". In: *Nature Communications* 8 (2017). DOI: 10.1038/s41467-017-00869-x.

[32] W. Schindler. "Evaluation criteria for physical random number generators". In: *Cryptographic Engineering*. 2009, pp. 25–54.

[33] *AIS 20: Funktionalitätsklassen und Evaluationsmethodologie für deterministische Zufallszahlengeneratoren. Version 3*. https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Zertifizierung/Interpretationen/AIS_20_pdf.pdf. May 2013.

[34] *AIS 31: Funktionalitätsklassen und Evaluationsmethodologie für physikalische Zufallszahlengeneratoren. Version 3*. https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Zertifizierung/Interpretationen/AIS_31_pdf.pdf. May 2013.

[35] W. Killmann and W. Schindler. *A proposal for: Functionality classes for random number generators. Mathematical-technical reference AIS20 and AIS31, Version 2.0*. https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Zertifizierung/Interpretationen/AIS_20_Functionality_classes_for_random_number_generators_e.pdf?__blob=publicationFile&v=1. Sept. 2011.

[36] U. Rührmair and D. E. Holcomb. "PUFs at a glance". In: *2014 Design, Automation Test in Europe Conference Exhibition (DATE)*. 2014, pp. 1–6.

[37] F. Koushanfar. "Integrated circuits metering for piracy protection and digital rights management: An overview". In: *Proceedings of the 21st Edition of the Great Lakes Symposium on Great Lakes Symposium on VLSI*. 2011, pp. 449–454.

[38] J. Guajardo, S. S. Kumar, G. J. Schrijen, and P. Tuyls. "FPGA intrinsic PUFs and their use for IP protection". In: *Cryptographic Hardware and Embedded Systems - CHES 2007*. 2007, pp. 63–80.

[39] P. Koeberl, Ü. Koçabas, and A.-R. Sadeghi. "Memristor PUFs: A new generation of memory-based physically unclonable functions". In: *2013 Design, Automation Test in Europe Conference Exhibition (DATE)*. 2013, pp. 428–431.

[40] U. Rührmair, H. Busch, and S. Katzenbeisser. "Strong PUFs: Models, Constructions, and Security Proofs". In: *Towards Hardware-Intrinsic Security: Foundations and Practice*. 2010, pp. 79–96.

[41] Y. Gao, D. C. Ranasinghe, S. F. Al-Sarawi, O. Kavehei, and D. Abbott. "Memristive crypto primitive for building highly secure physical unclonable functions". In: *Scientific Reports* 5 (2015).

[42] J. Rajendran, G. Rose, R. Karri, and M. Potkonjak. "Nano-PPUF: A memristor-based security primitive". In: *2012 IEEE Computer Society Annual Symposium on VLSI*. 2012, pp. 84–87.

[43] N. Rathi, S. Ghosh, A. Iyengar, and H. Naeimi. "Data privacy in non-volatile cache: Challenges, attack models and solutions". In: *2016 21st Asia and South Pacific Design Automation Conference (ASP-DAC)*. 2016, pp. 348–353.

6 Near- and In-Memory Computing

Data movement is identified in chapter 4 as the main source of inefficiency in current computing systems. HBM is a first but still conventional step to more efficiency of CPU-centric computing, but in consequence the memory should be in the center of our notion of computing and lastly leading to processing in memory.

6.1 Memory-Centric Computing

A paradigm shift is proposed towards so-called Memory-Centric Computing, which sees the memory instead of processors in the centre of hardware and software design, and stands in contrast to contemporary CPU-centric computing. The *Memory-Centric Computing* paradigm is favoured by HPE's visionary computer architecture called "The Machine", which unifies memory and storage into one vast pool of memory which originally should be based on ReRAM memristor technology and photonic busses. Processing is still performed by outside processors or processor cores.

HPE's latest prototype of The machine of 2017 houses 160 TB of so-called HPE Persistant Memory as shared memory. However, the HPE Persistant Memory is still implemented as DRAM-based NVDIMMs with Flash as memory back-up [1]. HPE is still struggling to apply memristor technology within their "The Machine" project, as it was the original plan for The Machine. The HPE Persistent Memory used in the prototype is still seen as an interim step to a more advanced byte-addressable non-volatile memory, which may be ReRAM- or PCM-based [2].

Near- and in-memory computing belong to the Memory-Centric Computing paradigm but go one step further than "The Machine" towards blurring the difference between memory access and computing. Near- and in-memory computing concern moving part of the computation to where the data resides specifically, in a 3D stacked memory context.

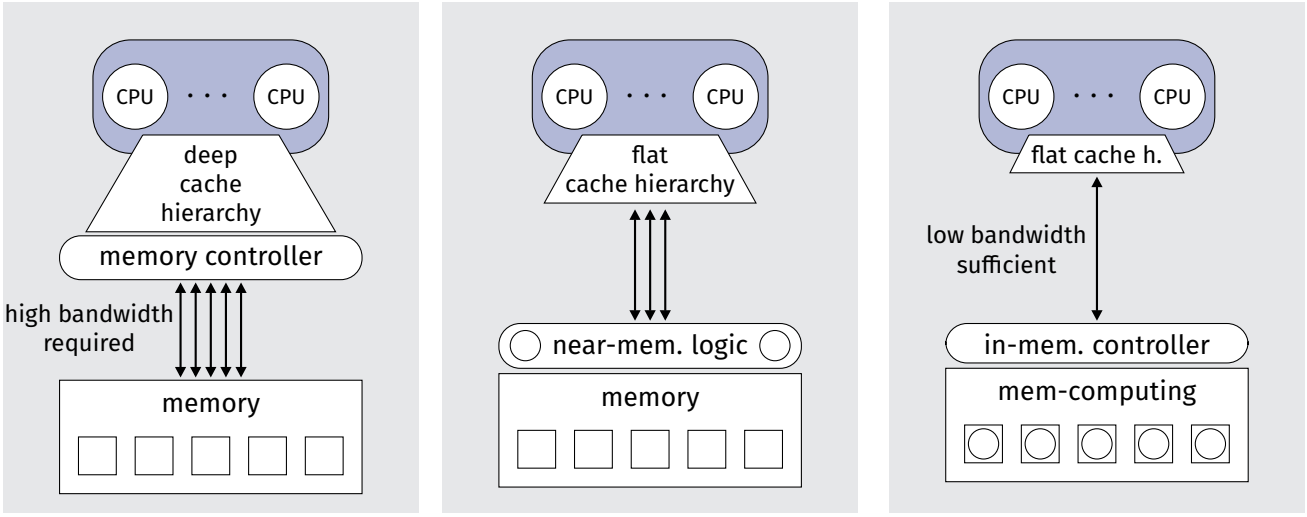
6.2 Near- and In-Memory Computing

Near-memory computing (Near-Memory Processing, NMP) is characterized by processing in proximity of memory to minimize data transfer costs [3]. Compute logic, e.g. small cores, is physically placed close to the memory chips in order to carry out processing steps, like e.g. stencil operations, or vector operations on bulk of data. Near-memory computing can be seen as a co-processor or hardware accelerator. Near-memory computing can be realized by replacing or enhancing the memory controller to be able to perform logic operations on the row buffer. In HBM the Logic Die (see Fig. 4.1 could be enhanced by processing capabilities, and the memory controller can be enabled to perform semantically richer operations than load and store, respectively cache line replacements.

Near-memory computation can provide two main opportunities: (1) reduction in data movement by vicinity to the main storage resulting in reduced memory access latency and energy, (2) higher bandwidth provided by Through Silicon Vias (TSVs) in comparison with the interface to the host limited by the pins [4].

Processing by near-memory computing reduces energy costs and goes along with a reduction of the amount of data to be transferred to the processor. Near-memory computing is to be considered as a near- and mid-term realizable concept.

Proposals for near-memory computing architectures currently don't rely yet on memristor technologies but on innovative memory devices which are commercially available in the meantime such as the Hybrid Memory Cube from Micron [5] [6]. It stacks multiple DRAM dies and a separate layer for a controller which is vertically linked with the DRAM dies. The Smart Memory Cube proposed by [4] is the proposal of a near-memory computing architecture enhancing the capabilities of the logic die in the Hybrid Memory Cube. The Mondrian Data Engine [7] investigates algorithms of data analytics for near-memory computing.



(a) CPU centric, "far-memory" (b) near-memory computing (c) in-memory computing

Figure 6.1: From CPU-centric to near-memory computing and in-memory computing. A memory cell is depicted as □, computing logic is drawn as a circle ○, and ◙ represents a memory cell with processing capability.

Such technologies exploit similar to HBM tight 3D integration of DRAM memory modules. However, data buffers and small accelerator cores are attached for near-memory computing in the data path between memory controller and DRAM modules. This forms a near-DRAM acceleration architecture which demonstrated in conceptional investigations for known benchmark suites both speed-up increase and energy decrease compared to non near-DRAM acceleration architectures [8].

Near-memory computing can use available basic memory modules. The challenge is more focused on building new and efficient system architectures. Also the software side is affected, namely new specific instructions have to be created in the instruction sets that consider near-memory computing accelerator instructions [9].

In-memory computing (In-Memory Processing, IMP) goes a step further such that the memory cell itself is not only a storage cell but it becomes an integral part of the processing step. This can help to further reduce the energy consumption and the area requirement in comparison to near-memory computing. However, this technology has to be improved and therefore it is considered at least as a mid-term or probably as a more long-term solution.

Near- and in-memory computing are two processor in memory (PIM) approaches that change the interface between the processor and the memory: memory will in future not only be accessed by move operations between processor and RAM memory, but additionally provide a semantically stronger access pattern based on simple compute operations on a large number of memory cells. It is preferable to process data in-situ directly where they are located before they are sent to the processor cores.

Near- and future in-memory computing will profit from die stacking and 3D chip integration. The main challenge in establishing 3D chip stacking technology with near-memory computing is gaining control of the thermal problems that have to be overcome to realize reliably very dense 3D interconnections. While near-memory computing still requires a potentially heat emitting logic cell in the 3D stack, future in-memory computing based on memristors may solve the heat problem. Computing is done within the memristor cells. Memristors as memory technology do not need a refresh which is required by DRAM in HBM.

Fig. 6.1 demonstrates the principal differences between the state-of-the-art CPU-centric multicores and near- and in-memory computing. Current CPU-centric architectures feature a "far-memory" of DRAM chips and require a deep cache hierarchy on the processor chip to exploit locality of data and omit costly DRAM transfers. Near-memory computing requires a more complex near-memory logic as the memory controller and may rely on a flatter cache hierarchy, because less data is transferred between processor and near-memory computing device. Similarly, fu-

ture in-memory computing may need an even smaller cache hierarchy by transferring most computing requirements to the in-memory computing device.

Applications for near- and in-memory computing accelerators could be data intensive applications often categorized as "Big Data" workloads. Such accelerators are especially fitting for data analytics, as they provide immense bandwidth to memory-resident data and dramatically reduce data movement, the main source of energy consumption. Analytic engines for business intelligence are increasingly memory resident to minimize query response time [7]. Graph traversal applications are fitting well due to their unpredictable memory access patterns and high ratio of memory access to computation. Such algorithms are common in social network analysis as e.g. Average Teenage Follower (ATF) that counts for each vertex the number of its teenage followers by iterating over all teenager, in Breadth-First Search (BFS), PageRank (PR), and Bellman-Ford Shortest Path (BF) [4].

6.3 Future Near- and In-Memory Computing with Memristors

Some memristive technologies can be integrated with CMOS circuits in a so-called BEOL (back end of line) process without costly 3D stacking processes. In this case the memristive elements are deposited inside holes on the top metal layer and an additional top electrode for the memristive element has to be realized on the top layer while the bottom electrode is realized in the layers beneath. Another approach is the direct integration of memristive behavior directly in MOSFET gate transistors as so-called MemFlash which was demonstrated for neuromorphic memristive cells [10]. For both approaches holds that current design tools do not support automatic integration techniques and simulations of both technologies, CMOS and memristive devices. This is one of the future research tasks.

6.4 Implication

Near- and in-memory computing will influence the concept of Storage-class Memory (SCM, see chapter 4), i.e., a non-volatile memory technology in between memory and storage, which may enable new data access modes and protocols that are neither "memory" nor "storage". In-memory computing will also influence strongly edge computing approaches, in which new architectures have to be found that are characterized by processing data directly at sensors where the data is captured to reduce as described above the amount of data that has to be transferred to more-coarse grained cores for post-processing.

Assuming that near- and in-memory computing technologies will be mature, we need to change algorithms and data structures to fit the new design and thus allow memory-heavy "in-memory" computing algorithms to achieve significantly better performance. We may need to replace the notion of general purpose computing with clusters of specialized compute solution. Accelerators will be "application class" based, e.g. for deep learning (such as Google's TPU and Fujitsu's DLU), molecular dynamics, or other important domains.

But new technologies, such as optical networks on die and Terahertz based connections, may reduce the need for preserving locality, since the differences in access time and energy costs to local memory vs. remote storage or memory may not be as significant in future as it is today. When such new technologies find their practical use, we can expect a massive change in the way we are building hardware and software systems and are organizing software structures. It is an open research question how near-memory computing, in-memory computing, memory-centric computing and the potential adverse trend given by optics may match.

References

[1] H. P. Enterprise. *HPE Persistent Memory for HPE ProLiant Servers, White paper 2016*. URL: https://h20195.www2.hpe.com/v2/GetPDF.aspx/4AA6-4680ENW.pdf.

[2] M. Feldman. *HPE Unveils New Prototype of Memory-Driven Computer, May 17, 2017*. URL: https://www.top500.org/news/hpe-unveils-new-prototype-of-memory-driven-computer/.

[3] S. Khoram, Y. Zha, J. Zhang, and J. Li. "Challenges and Opportunities: From Near-memory Computing to In-memory Computing". In: *Proceedings of the 2017 ACM on International Symposium on Physical Design*. ACM. 2017, pp. 43–46.

[4] E. Azarkhish, D. Rossi, I. Loi, and L. Benini. "Design and Evaluation of a Processing-in-Memory Architecture for the Smart Memory Cube". In: *Proceedings of the Architecture of Computing Systems – ARCS 2016, Lecture Notes in Computer Science, vol 9637. Springer*. 2016.

[5] AMD. *High Bandwidth Memory*. 2017. URL: https://www.amd.com/Documents/High-Bandwidth-Memory-HBM.pdf.

[6] J. Jeddeloh and B. Keeth. "Hybrid memory cube new DRAM architecture increases density and performance". In: *VLSI Technology (VLSIT), 2012 Symposium on*. IEEE. 2012, pp. 87–88.

[7] M. Drumond, A. Daglis, N. Mirzadeh, D. Ustiugov, J. Picorel, B. Falsafi, B. Grot, and D. Pnevmatikatos. "The Mondrian Data Engine". In: *Proceedings of the 44th Annual International Symposium on Computer Architecture*. ACM. 2017, pp. 639–651.

[8] N. S. Kim, D. Chen, J. Xiong, and W.-m. W. Hwu. "Heterogeneous Computing Meets Near-Memory Acceleration and High-Level Synthesis in the Post-Moore Era". In: *IEEE Micro* 37.4 (2017), pp. 10–18. DOI: 10.1109/MM.2017.3211105. URL: http://ieeexplore.ieee.org/document/8013455/.

[9] J. Ahn, S. Yoo, O. Mutlu, and K. Choi. "PIM-enabled instructions: a low-overhead, locality-aware processing-in-memory architecture". In: 2015, pp. 336–348. DOI: 10.1145/2749469.2750385. URL: http://dl.acm.org/citation.cfm?doid=2749469.2750385.

[10] M. Ziegler, M. Oberländer, D. Schroeder, W. Krautschneider, and H. Kohlstedt. "Memristive operation mode of floating gate transistors: A two-terminal MemFlash-cell". In: *Applied Physics Letters* 101 (Dec. 2012), p. 263504.

7 Reconfigurable Computing exploiting Memristor Technology

Reconfigurable computing combines the advantages of programmability of software with the performance of hardware. Industry and research exploit this ability for fast prototyping of hardware, update hardware in the field or to reduce costs in environments where a company only requires a small volume of chips. Even in High Performance Computing (HPC), reconfigurable hardware plays an important role by accelerating time consuming functions. Reconfigurable hardware is well-integrated in modern computational environments, like in System on Chips (SoCs) or additional accelerator cards. The most common chip types used for reconfigurable hardware are Field Programmable Gate Arrays (FPGAs). Their importance has increased in the last years because FPGA vendors like Xilinx and Intel switched to much smaller chip fabrication processes and could double the size of the available reconfigurable hardware per chip.

At the moment reconfigurable hardware is produced in a standard CMOS fabrication process. Configuration memory, Block-RAM, and Look-Up-Tables (LUTs) are implemented using Static-Random-Access-Memory (SRAM) cells or flash-based memory. Crossbar switches consisting of multiple transistors provide routing and communication infrastructure.

7.1 Applying Memristor Technology in Reconfigurable Hardware

CMOS compatibility and a small area and power footprint are the important features of memristor technology for reconfigurable hardware. At the moment the main challenges for reconfigurable hardware are a high static power consumption and long interconnection delays. Memristor technology, applied to important building blocks of reconfigurable hardware, can help overcoming these challenges.

The following sections describe the impact of memristor technology to key parts of reconfigurable hardware.

7.1.1 Memristors in Block RAM

Block RAM is the most obvious part of reconfigurable hardware for the deployment of memristor technology. Current *Block RAM* is SRAM based and one SRAM cell consists of six CMOS transistors.

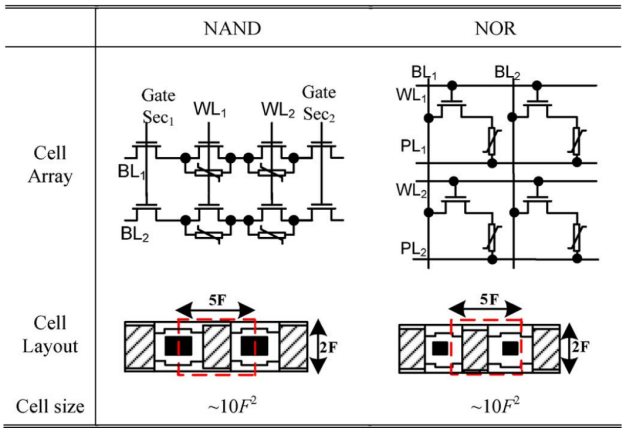


Figure 7.1: NAND and NOR Cells in 1T1R technique[1]

The 1T1R[1] memristor technique introduced by Tanachutiwat et al. [1] reduces the number of transistors required for one memory cell to one. Figure 7.1 displays the implementation of NAND and NOR cells in 1T1R technique. The first row displays the abstract implementation technique and the second row the actual cell layout in CMOS technology. In contrast to flash based memory cells, the preferred implementation for 1T1R are NOR cells because they provide a faster random access. A NAND cell array uses a serial interface to access each cell, reducing the access bandwidth.

The memristor based memory cells require a small encode/decode hardware, but this technique still has an area density enhancement of six times to the SRAM based approach. The memristor based cells only require power if their content changes, reducing the static power consumption of reconfigurable hardware. Because of the density improvements even more *Block RAM* can be deployed on the reconfigurable hardware

[1] 1 Transistor Element and 1 Resistive Element

than currently available. Another important improvement using memristor technology is its non-volatile feature. Even if the whole reconfigurable hardware looses power, the content of the *Block RAM* is still available after power restoration.

7.1.2 Memristors in CLBs

The CLBs are another important building block of reconfigurable hardware because they implement the different hardware functions. In general this is achieved by using/ combining LUTs and/or multiplexers. Like *Block RAM*, LUTs are, at the moment, based on SRAM cells. The 1TR1 approach of Section 7.1.1 is also a simple approach to improve area density and power consumption within LUTs. For example, Kumar[2] uses this approach. At the moment the improvements of CLBs with memristors is not the focus of research because the logic building blocks are not related to current challenges in reconfigurable hardware. Still, using the the 1T1R approach, the non-volatile feature of memristors would improve configuration management of reconfigurable hardware because the configuration of the hardware does not need to be reloaded after a power loss.

7.1.3 Memristors in the Interconnection Network

The interconnection network of reconfigurable hardware is responsible for 50%-90% of the total reconfigurable hardware area usage, 70%-80% of the total signal delay and 60%-85% of the total power consumption[3]. Improving the interconnection network will have a huge impact on the overall reconfigurable hardware performance. Routing resources of the interconnection network are implemented using seven CMOS transistors at the moment. Six transistors for a SRAM cell and one transistor for controlling the path.

Tanachutiwat etal. [1] extend their 1TR1 approach for *Block RAM* cells to a 2T1R and 2T2R technique for routing switches. The second is fully compatible to the current implementation because one transistor controls the path, while in the 2T1R technique a memristor does.

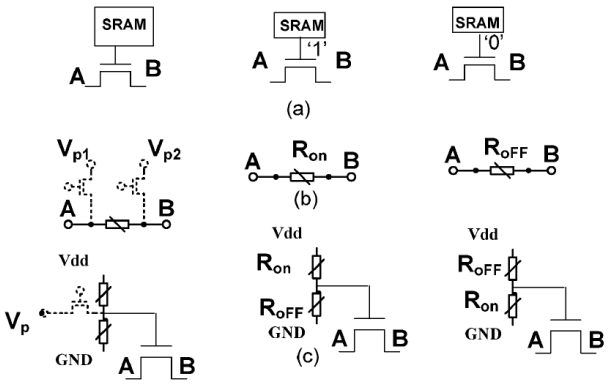


Figure 7.2: Comparison of SRAM, 2TR1 and 2TR2 based switch[1]

Figure 7.2 compares the three different approaches. Row *a* shows the implementation of a SRAM based switch, its on and off state. The same information is described in row *b* and *c* for the 2T1R and the 2T2R approach.

The 2T1R and 2T2R approach is also used by Hasan etal. [4] to build complex crossbar switches. A complex routing switch built out of many 2T1R or 2R2R elements can save even more transistors by combining different programming transistors.

The memristor based improvements for the interconnection network reduce the standby power of reconfigurable hardware considerably. They also reduce the area requirements for the interconnection network, allowing a more dense placement of the logic blocks and, therefore, improving the overall signal delay. Like in the previous sections, the non-volatile nature of the memristors prevents configuration loss on power disconnect.

7.2 Conclusion and Research Perspective

Memristor technology will have a high impact on reconfigurable hardware development and research. Section 7.1 presented improvements through memristor technology on important building blocks of reconfigurable hardware. These improvements target power consumption and area reduction, both important challenges of modern reconfigurable hardware development.

Still, researchers must evaluate the proposed solutions more because *spice* is the preferred and only method for evaluation. Next steps have to include prototype development.

At the moment, the non-volatile nature of the memristor technology is not the focus of research. But this aspect can be a game changer for certain application areas and even open up new application areas for reconfigurable computing. For example, a reconfigurable hardware system would not require any external configuration memory and the initialization time of a system could be reduced multiple times. Deep sleep states are easily implemented, reducing the power consumption even more. These improvements are important for application areas like building automation, wearables and safety critical applications.

Further research areas include the evaluation of memristor technology in the logic building block of reconfigurable hardware, more research in the optimization of routing and interconnection resources with memristors, and the evaluation of the non-volatile aspects of memristors for reconfigurable hardware applications.

References

[1] S. Tanachutiwat, M. Liu, and W. Wang. "FPGA Based on Integration of CMOS and RRAM". In: *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 19.11 (2011), pp. 2023–2032.

[2] T. Nandha Kumar. "An Overview on Memristor-Based Nonvolatile LUT of an FPGA". In: *Frontiers in Electronic Technologies: Trends and Challenges*. Singapore, 2017, pp. 117–132. DOI: 10.1007/978-981-10-4235-5_8. URL: https://doi.org/10.1007/978-981-10-4235-5_8.

[3] J. Cong and B. Xiao. "FPGA-RPI: A Novel FPGA Architecture With RRAM-Based Programmable Interconnects". In: *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 22.4 (Apr. 2014), pp. 864–877. DOI: 10.1109/TVLSI.2013.2259512.

[4] R. Hasan and T. M. Taha. "Memristor Crossbar Based Programmable Interconnects". In: *2014 IEEE Computer Society Annual Symposium on VLSI*. July 2014, pp. 94–99. DOI: 10.1109/ISVLSI.2014.100.

8 Memristive Computing

In this section, memristive (also called resistive) computing is discussed in which logic circuits are built by memristors [1].

8.1 Overview of Memristive Computing

Memristive computing is one of the emerging and promising computing paradigms [1, 2, 3]. It takes the data-centric computing concept much further by interweaving the processing units and the memory in the same physical location using non-volatile technology, therefore significantly reducing not only the power consumption but also the memory bottleneck. Resistive devices such as memristors have been shown to be able to perform both storage and logic functions [1, 4, 5, 6].

Memristive gates have a lower leakage power, but switching is slower than in CMOS gates [7]. However, the integration of memory into logic allows to reprogram the logic, providing low power reconfigurable components [8], and can reduce energy and area constraints in principle due to the possibility of computing and storing in the same device (computing in memory). Memristors can also be arranged in parallel networks to enable massively parallel computing [9].

Memristive computing provides a huge potential as compared with the current state-of the art:

- It significantly reduces the memory bottleneck as it interweaves the storage, computing units and the communication [1, 2, 3].
- It features low leakage power [7].
- It enables maximum parallelism [3, 9] by in-memory computing.
- It allows full configurability and flexibility [8].
- It provides order of magnitude improvements for the energy-delay product per operations, the computation efficiency, and performance per area [3].

Serial and parallel connections of memristors were proposed for the realization of Boolean logic gates with memristors by the so-called *memristor ratio logic*. In such circuits the ratio of the stored resistances in memristor devices is exploited for the set-up of Boolean logic. Memristive circuits realizing AND gates, OR gates, and the implication function were presented in [10, 11, 12].

Hybrid memristive computing circuits consist of memristors and CMOS gates. The research of Singh [13], Xia et.al. [14], Rothenbuhler et.al. [12], and Guckert and Swartzlaender [15] are representative for numerous proposals of hybrid memristive circuits, in which most of the Boolean logic operators are handled in the memristors and the CMOS transistors are mainly used for level restoration to retain defined digital signals.

Figure 8.1 summarizes the activities on memristive computing. We have the large block of hardware support with memristive elements for neural networks, neuromorphic processing, and STDP (spike-timing-dependent plasticity) (see section 9). Concerning the published papers a probably much smaller branch of digital memristive computing with several sub branches, like ratioed logic, imply logic or CMOS-like equivalent memristor circuits in which Boolean logic is directly mapped onto crossbar topologies with memristors. These solutions refer to pure in-memory computing concepts. Besides that, proposals for hybrid solutions exist in which the memristors are used as memory for CMOS circuits in new arithmetic circuits exploiting the MLC capability of memristive devices.

8.2 Current State of Memristive Computing

A couple of start-up companies appeared in 2015 on the market who offer memristor technology as BEOL (Back-end of line) service in which memristive elements are post-processed in CMOS chips directly on top of the last metal layers. Also some European institutes reported just recently at a workshop meeting

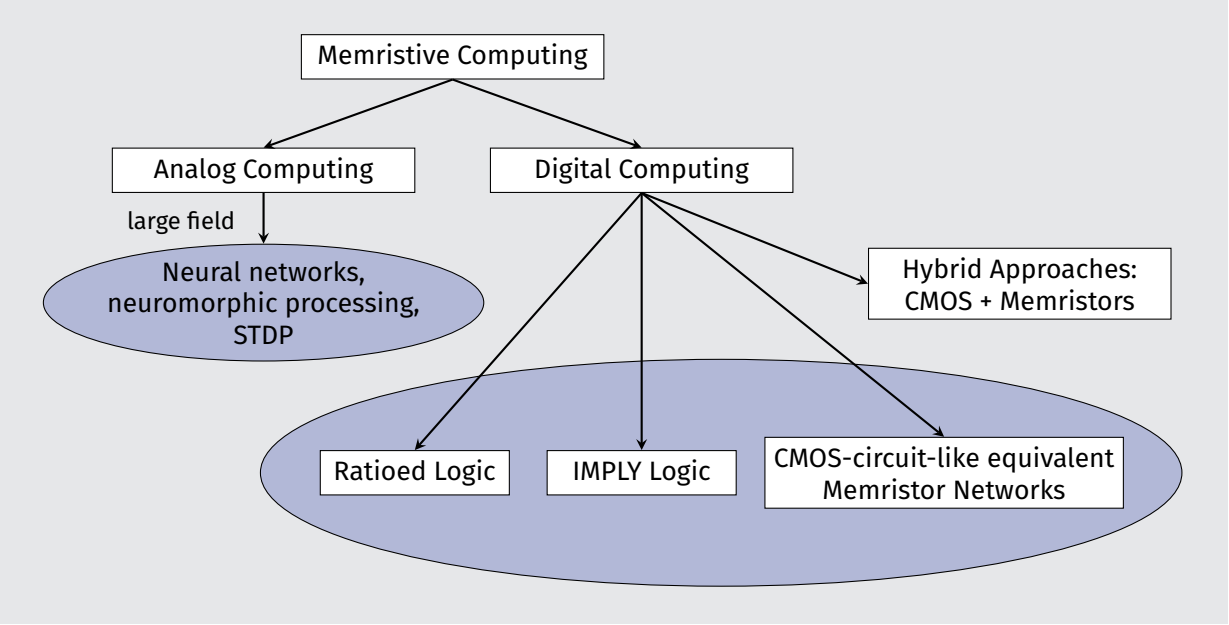


Figure 8.1: Summary of activities on memristive computing.

"Memristors: at the crossroad of Devices and Applications" of the EU cost action 1401 MemoCiS[1] the possibility of BEOL integration of their memristive technology to allow experiments with such technologies [16]. This offers new perspectives in form of hybrid CMOS/memristor logic which uses memristor networks for high-dense resistive logic circuits and CMOS inverters for signal restoration to compensate the loss of full voltage levels in memristive networks.

Multi-level cell capability of memristive elements can be used to face the challenge to handle the expected huge amount of Zettabytes produced annually in a couple of years. Besides, proposals exist to exploit the multi-level cell storing property for ternary carry-free arithmetic [17, 18] or for both compact storing of keys and matching operations in future associative memories realized with memristors [19], so-called ternary content-addressable memories.

8.3 Impact on Hardware

Using NVM technologies for resistive computing is a further step towards energy-aware measures for future HPC architectures. In addition, there exist technology facilities at the IHP in Frankfurt/O which at least for small feature sizes allow to integrate memristors and CMOS logic in an integrated chip without a separate BEOL process step. It supports the realization of both near-memory and in-memory computing concepts which are both an important brick for the realization of more energy-saving HPC systems. Near-memory could be based on 3D stacking of a logic layer with DRAMs, e.g. extending Intel's High Bandwidth Memory (HBM) with NVM devices and stacked logic circuitry in future. In-memory computing could be based on memristive devices using either ReRAM, PCM, or STT-RAM technology for simple logic and arithmetic pre-processing operations.

A further way to save energy, e.g. in near-memory computing schemes, is to use non-volatile register cells as flip-flops or in memory cell arrays. During the last decade, the basic principle in the design of non-volatile FlipFlop (nvFF) has been to compose them from a standard CMOS Flip-Flop (FF) and a *non-volatile* memory cell, either as part of a flip-flop memristor register pair or as pair of a complete SRAM cell array and a subsequent attached memristor cell array (hybrid NVMs). At predefined time steps or on power loss, this *non-volatile* memory cell backups the contents of the standard FF. At power recovery, this content is restored in the FF and the Non-Volatile-Processor (NVP) can continue at the exact same state. nvFFs following this approach require a centralized controller to initiate a backup or a restore operation. This centralized controller has to issue the backup signal as fast as possible after a no-power standby, otherwise data and processing progress may be lost.

Four different implementation categories of nvFFs using hybrid retention architectures are available to-

[1] www.cost.eu/COST_Actions/ict/IC1401

day:

- **Ferroelectric nvFF:** This category uses a ferroelectric capacitor to store one bit. Masui et al. [20] introduced this kind of nvFFs, but different approaches are also available.
- **Magnetic RAM (MRAM) nvFF:** This approach uses the spin direction of Magnetic Tunnel Junctions to store a bit. [21]
- **CAAC-OS nvFF:** CAAC-OS transistors have an extremely low off-state current. By combining them with small capacitors a nvFF can be created[22]. The access times of these nvFFs are very low.
- **ReRAM nvFF:** ReRAMs are a special implementation of NVM using memristor technology. They do not consume any power in their off-state. nvFFs implementations using ReRAM are currently evaluated[23, 24].

These approaches can also be applied to larger hybrid NVMs, where data, which has to be processed, is stored in conventional faster SRAM/DRAM devices. By using pipeline schemes, e.g. under control of the OS, part of the data is shifted from NVM to SRAM/-DRAM before it is accessed in the fast memory. Then, the latency for the data transfer from NVM to DRAM can be hidden by a timely overlapping of data transfer with simultaneous processing of other parts of the DRAM. The same latency hiding principle can happen in the opposite direction. Data that is newly computed and that is not needed in the next computing steps can be saved in NVMs.

Table 8.1 displays performance parameters of these nvFFs. According to overall access time and energy requirements the MRAM and the ReRAM approach are the most promising ones. But the ReRAM approach has more room for improvements because the fabrication technology is still very large compared to the current standard of seven nanometer. Table 8.1 also shows the impact memristor technology can have on NVPs. At the moment memristor-based nvFFs are only produced for research at a very large fabrication process of 180 nm. Still they can compete with nvFFs produced at a much smaller size, using a different technology.

Latest research papers propose an integrated FF design either by using a single memristor combined with pass transistors and a high-valued resistor [26] or by using a sense amplifier reading the differential state of two memristors, which are controlled by two transmission gates [27]. The latter approach seems to be beneficial in terms of performance, power consumption, and robustness and shows a large potential to be used for no-power standby devices which can be activated instantaneously upon an input event.

8.4 Perspective

Memristive computing, if successful, will be able to significantly reduce the power consumption and enable massive parallelism, hence, increase computing energy and area efficiency by orders of magnitudes. This will transform computer systems into new highly parallel architectures and associated technologies, and enable the computation of currently infeasible big data and data-intensive applications, fuelling important societal changes.

Research on resistive computing is still in its infancy stage, and the challenges are substantial at all levels, including material and technology, circuit and architecture, tools and compilers, and algorithms. As of today most of the work is based on simulations and small circuit designs. It is still unclear when the technology will be mature and available. Nevertheless, some start-ups on memristor technologies are emerging such as KNOWM[2], BioInspired[3], and Crossbar[4].

References

[1] J. Borghetti, G. S. Snider, P. J. Kuekes, J. J. Yang, D. R. Stewart, and R. S. Williams. "Memristive switches enable stateful logic operations via material implication". In: *Nature* 464.7290 (Apr. 2010), pp. 873–876. DOI: 10.1038/nature08940. URL: https://www.nature.com/nature/journal/v464/n7290/full/nature08940.html.

[2] M. Di Ventra and Y. V. Pershin. "Memcomputing: a computing paradigm to store and process information on the same physical platform". In: *Nature Physics* 9.4 (Apr. 2013), pp. 200–202. DOI: 10.1038/nphys2566. URL: http://arxiv.org/abs/1211.4487.

[3] S. Hamdioui et al. "Memristor based computation-in-memory architecture for data-intensive applications". In: *2015 Design, Automation Test in Europe Conference Exhibition (DATE)*. Mar. 2015, pp. 1718–1725.

[4] G. Snider. "Computing with hysteretic resistor crossbars". In: *Applied Physics A* 80.6 (Mar. 2005), pp. 1165–1172. DOI: 10.1007/s00339-004-3149-1. URL: https://link.springer.com/article/10.1007/s00339-004-3149-1.

[2] www.knowm.org
[3] http://www.bioinspired.net/
[4] https://www.crossbar-inc.com/en/

nvFF	FeRAM	MRAM	ReRAM	CAAC-OS
Technology	130nm	90nm	180nm	1um
Store Time	320ns	4ns	10ns	40ns
Store Energy(pJ/bit)	2.2	6	0.84	1.6
Recall Time	384ns	5ns	3.2ns	8ns
Recall Energy(pJ/bit)	0.66	0.3	N/A	17.4

Table 8.1: Performance comparison of nvFF types[25]

[5] L. Gao, F. Alibart, and D. B. Strukov. "Programmable CMOS/Memristor Threshold Logic". In: *IEEE Transactions on Nanotechnology* 12.2 (Mar. 2013), pp. 115–119. DOI: 10.1109/TNANO.2013.2241075.

[6] L. Xie, H. A. D. Nguyen, M. Taouil, S. Hamdioui, and K. Bertels. "Fast boolean logic mapped on memristor crossbar". In: *2015 33rd IEEE International Conference on Computer Design (ICCD)*. Oct. 2015, pp. 335–342. DOI: 10.1109/ICCD.2015.7357122.

[7] Y. V. Pershin and M. D. Ventra. "Neuromorphic, Digital, and Quantum Computation With Memory Circuit Elements". In: *Proceedings of the IEEE* 100.6 (June 2012), pp. 2071–2080. DOI: 10.1109/JPROC.2011.2166369.

[8] J. Borghetti, Z. Li, J. Straznicky, X. Li, D. A. A. Ohlberg, W. Wu, D. R. Stewart, and R. S. Williams. "A hybrid nanomemristor/transistor logic circuit capable of self-programming". In: *Proceedings of the National Academy of Sciences* 106.6 (Feb. 2009), pp. 1699–1703. DOI: 10.1073/pnas.0806642106. URL: http://www.pnas.org/content/106/6/1699.

[9] Y. V. Pershin and M. Di Ventra. "Solving mazes with memristors: a massively-parallel approach". In: *Physical Review E* 84.4 (Oct. 2011). DOI: 10.1103/PhysRevE.84.046703. URL: http://arxiv.org/abs/1103.0021.

[10] J. J. Yang, D. B. Strukov, and D. R. Stewart. "Memristive devices for computing". In: *Nature Nanotechnology* 8.1 (Jan. 2013), pp. 13–24. DOI: 10.1038/nnano.2012.240. URL: http://www.nature.com/nnano/journal/v8/n1/full/nnano.2012.240.html.

[11] S. Kvatinsky, A. Kolodny, U. C. Weiser, and E. G. Friedman. "Memristor-based IMPLY Logic Design Procedure". In: *Proceedings of the 2011 IEEE 29th International Conference on Computer Design*. ICCD '11. Washington, DC, USA, 2011, pp. 142–147. DOI: 10.1109/ICCD.2011.6081389. URL: http://dx.doi.org/10.1109/ICCD.2011.6081389.

[12] T. Tran, A. Rothenbuhler, E. H. B. Smith, V. Saxena, and K. A. Campbell. "Reconfigurable Threshold Logic Gates using memristive devices". In: *2012 IEEE Subthreshold Microelectronics Conference (SubVT)*. Oct. 2012, pp. 1–3. DOI: 10.1109/SubVT.2012.6404301.

[13] T. Singh. "Hybrid Memristor-CMOS (MeMOS) based Logic Gates and Adder Circuits". In: *arXiv:1506.06735 [cs]* (June 2015). URL: http://arxiv.org/abs/1506.06735.

[14] Q. Xia et al. "Memristor?CMOS Hybrid Integrated Circuits for Reconfigurable Logic". In: *Nano Letters* 9.10 (Oct. 2009), pp. 3640–3645. DOI: 10.1021/nl901874j. URL: http://dx.doi.org/10.1021/nl901874j.

[15] L. Guckert and E. E. Swartzlander. "Dadda Multiplier designs using memristors". In: *2017 IEEE International Conference on IC Design and Technology (ICICDT)*. 2017.

[16] J. Sandrini, M. Thammasack, T. Demirci, P.-E. Gaillardon, D. Sacchetto, G. De Micheli, and Y. Leblebici. "Heterogeneous integration of ReRAM crossbars in 180 nm CMOS BEoL process". In: 145 (Sept. 2015).

[17] A. A. El-Slehdar, A. H. Fouad, and A. G. Radwan. "Memristor based N-bits redundant binary adder". In: *Microelectronics Journal* 46.3 (Mar. 2015), pp. 207–213. DOI: 10.1016/j.mejo.2014.12.005. URL: http://www.sciencedirect.com/science/article/pii/S0026269214003541.

[18] D. Fey. "Using the multi-bit feature of memristors for register files in signed-digit arithmetic units". In: *Semiconductor Science and Technology* 29.10 (2014), p. 104008. DOI: 10.1088/0268-1242/29/10/104008. URL: http://stacks.iop.org/0268-1242/29/i=10/a=104008.

[19] P. Junsangsri, F. Lombardi, and J. Han. "A memristor-based TCAM (Ternary Content Addressable Memory) cell". In: *2014 IEEE/ACM International Symposium on Nanoscale Architectures (NANOARCH)*. July 2014, pp. 1–6. DOI: 10.1109/NANOARCH.2014.6880478.

[20] S. Masui, W. Yokozeki, M. Oura, T. Ninomiya, K. Mukaida, Y. Takayama, and T. Teramoto. "Design and applications of ferroelectric nonvolatile SRAM and flip-flop with unlimited read/program cycles and stable recall". In: *Proceedings of the IEEE 2003 Custom Integrated Circuits Conference, 2003.* Sept. 2003, pp. 403–406. DOI: 10.1109/CICC.2003.1249428.

[21] W. Zhao, E. Belhaire, and C. Chappert. "Spin-MTJ based Non-volatile Flip-Flop". In: *2007 7th IEEE Conference on Nanotechnology (IEEE NANO)*. Aug. 2007, pp. 399–402. DOI: 10.1109/NANO.2007.4601218.

[22] T. Aoki et al. "30.9 Normally-off computing with crystalline InGaZnO-based FPGA". In: *2014 IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC)*. Feb. 2014, pp. 502–503. DOI: 10.1109/ISSCC.2014.6757531.

[23] I. Kazi, P. Meinerzhagen, P. E. Gaillardon, D. Sacchetto, A. Burg, and G. D. Micheli. "A ReRAM-based non-volatile flip-flop with sub-VT read and CMOS voltage-compatible write". In: *2013 IEEE 11th International New Circuits and Systems Conference (NEWCAS)*. June 2013, pp. 1–4. DOI: 10.1109/NEWCAS.2013.6573586.

[24] A. Lee et al. "A ReRAM-Based Nonvolatile Flip-Flop With Self-Write-Termination Scheme for Frequent-OFF Fast-Wake-Up Nonvolatile Processors". In: *IEEE Journal of Solid-State Circuits* 52.8 (Aug. 2017), pp. 2194–2207. DOI: 10.1109/JSSC.2017.2700788.

[25] F. Su, Z. Wang, J. Li, M. F. Chang, and Y. Liu. "Design of nonvolatile processors and applications". In: *2016 IFIP/IEEE International Conference on Very Large Scale Integration (VLSI-SoC)*. Sept. 2016, pp. 1–6. DOI: 10.1109/VLSI-SoC.2016.7753543.

[26] J. Zheng, Z. Zeng, and Y. Zhu. "Memristor-based nonvolatile synchronous flip-flop circuits". In: *2017 Seventh International Conference on Information Science and Technology (ICIST)*. Apr. 2017, pp. 504–508. DOI: 10.1109/ICIST.2017.7926812.

[27] S. Pal, V. Gupta, and A. Islam. "Variation resilient low-power memristor-based synchronous flip-flops: design and analysis". In: *Microsystem Technologies* (July 2018). DOI: 10.1007/s00542-018-4044-6. URL: https://doi.org/10.1007/s00542-018-4044-6.

9 Approximate, Stochastic and Neuromorphic Computing Based on Memristors

Emerging architectural paradigms that have received substantial attention in the last few years are approximate, stochastic and neuromorphic architectures. *Approximate architectures* [1] realize some major advantages in circuit area and/or power consumption by tolerating a certain degree of inaccuracy [2]. *Stochastic computing* uses unary representation of numbers, which leads to very compact and power-efficient circuit realizations of basic functions [3]. *Neuromorphic and neuro-inspired approaches* mimic the functioning of human brain (or our understanding of its functioning) to efficiently perform computations that are difficult or impractical for conventional computer architectures [4] [5].

9.1 Memristors in Neuromorphic and Neuro-Inspired Computing

One extremely useful property of memristors in the context of neuromorphic is their biorealism, i.e., the ability to mimic behavior of elements found in human brain [6] and vision system [7]. Some of the early neuromorphic systems used capacitors to represent weights in the analog domain [4], and memristance can assume its role [6]. Well-known learning concepts, including spike-timing-dependent plasticity (STDP), can be mapped to memristive components in a natural way [8]. A recent example of a biorealistic hardware model is [9], which reports the manufacturing of a larger-scale network of artificial memristive neurons and synapses capable of learning. The memristive functionality is achieved by precisely controlling silver nanoparticles in a dielectric film such that their electrical properties closely matches ion channels in a biological neuron.

Biorealistic models are not the only application of memristors in neuromorphic or neuro-inspired architectures. Such architectures realize neural networks (NNs) with a vast amount of weights, which are determined, or learned, during the training phase, and then used without modification for an extended period of time, during the inference phase. After some time, when the relevant conditions have changed, it may become necessary to re-train the NN and replace its weights by new values. Memristive NVMs are an attractive, lightweight and low-power option for storing these weights. The circuit, once trained, can be activated and deactivated flexibly while retaining its learned knowledge. A number of neuromorphic accelerators based on memristive NVMs have been proposed in the last few years. For example, IBM developed a neuromorphic core with a 64-K-PCM-cell as "synaptic array" with 256 axons × 256 dendrite to implement spiking neural networks [10].

9.2 Memristors in Approximate Computing

Approximate architectures are inherently resilient against imperfections in their underlying manufacturing technologies [11], and some of the architectures have been specifically designed to work on unreliable substrates [12]. This robustness allows the designer to use emerging memristive technologies even before they have reached reliability levels comparable with their conventional counterparts. For example, an early study [13] showed that failures of certain locations within a JPEG compression circuit result in image modifications that a human viewer would not recognize. One approach to leverage the early benefits of memristive technologies would be to mix memristive and conventional CMOS circuitry, using more compact and power-efficient memristive devices in locations where ultimate reliability is not essential and sticking to reliable CMOS for critical parts, e.g, the system's control automaton.

Some approximate architectures incorporate on-chip infrastructures to monitor the system's state (e.g., the currently observed error levels) [14] and automatically switch between system modes. For example, near-threshold voltage architectures [15] support a number of voltage-frequency operating points, some of which are associated with a non-trivial error rate. Memristive devices have a strong application potential in such on-chip infrastructures. Memristive nonvolatile memories are a logical choice to store, e.g., the current values of voltage and frequency and also the values produced by on-chip sensors, which are

updated relatively infrequently but must be available permanently.

A further interesting possible application of memristive memories in approximate architectures is low-cost checkpointing. An approximate system may decide that the observed error rate is too high and the last produced computational results are worthless and should be recalculated. The system would then roll back to its last consistent state (checkpoint) stored in a memristive memory. Checkpointing is currently restricted to larger systems, like database servers [16], but non-volatile memristive memories would make it feasible even for small systems with no hard drive. Moreover, memristive memories are much faster than today's permament storage and using it for checkpointing would make this technique sufficiently fast for on-the-fly operation (the usual application of checkpointing is typically associated with very large delays).

9.3 Memristors in Stochastic Computing

Memristors have two main applications in stochastic computing: realization of actual computing elements (logic gates) and provision of high-quality random bit sequences [17]. Current stochastic circuit approaches use optimized pseudo-random number generators to produce sequences with certain entropy and correlation properties [18]. The inherent non-determinism of memristive devices suggests their use as a source of true physical randomness. For example, random fluctiations in read-out time of a memristive memory were employed in [17] for this purpose. An attractive property of this solution is its easy compatibility with the computational blocks in the remainder of the circuit (logic gates) which can be implemented in the same memristive technology as the true random number generator. Stochastic circuits tend to be quite compact, and this makes them a good testbed for an early implementation of a fully-memristive application.

9.4 Perspective

These radically new architectural concepts are proliferating due to two main factors. First, the applications that benefit from such architectures are increasing in importance. This chiefly refers to various kinds of machine learning, and in particular, classification approaches based on neural networks [19]. Especially important is the need for advanced machine learning and classification in extremely small, cheap and resource-constrained embedded IoT devices. For example, environmental monitoring and surveillance nodes should be able to classify sensor data directly at the node in order to avoid communication of vast amounts of data to a central server, but such nodes cannot incorporate high-performance microprocessors to run the necessary classification software. Dedicated hardware based on the above-mentioned architectural principles can be a key solution for such "near-sensor computing" scenarios [20, 21].

A second factor which fosters emerging architectures is the availability of new technologies with properties directly beneficial to these architectures. Memristors and especially memristor-based non-volatile memories (NVMs) are one important foundation for approximate, stochastic and neuromorphic computing [22]. The specific potentials and applications are discussed below for each of these three classes of architectures.

References

[1] S. Mittal. "A Survey of Techniques for Approximate Computing". In: *ACM Comput. Surv.* 48.4 (2016), 62:1–62:33.

[2] P. Gupta et al. "Underdesigned and Opportunistic Computing in Presence of Hardware Variability". In: *IEEE Trans. on CAD of Integrated Circuits and Systems* 32.1 (2013), pp. 8–23.

[3] A. Alaghi, W. Qian, and J. P. Hayes. "The Promise and Challenge of Stochastic Computing". In: *IEEE Trans. on CAD of Integrated Circuits and Systems* 37.8 (2018), pp. 1515–1531.

[4] C. Mead. "Neuromorphic electronic systems". In: *Proceedings of the IEEE* 78.10 (Oct. 1990), pp. 1629–1636.

[5] W. Wen, C. Wu, X. Hu, B. Liu, T. Ho, X. Li, and Y. Chen. "An EDA framework for large scale hybrid neuromorphic computing systems". In: *DAC*. 2015, 12:1–12:6.

[6] I. E. Ebong and P. Mazumder. "CMOS and Memristor-Based Neural Network Design for Position Detection". In: *Proceedings of the IEEE* 100.6 (2012), pp. 2050–2060.

[7] C. K. K. Lim, A. Gelencser, and T. Prodromakis. "Computing Image and Motion with 3-D Memristive Grids". In: *Memristor Networks*. 2014, pp. 553–583. DOI: 10.1007/978-3-319-02630-5_25.

[8] T. Serrano-Gotarredona, T. Masquelier, T. Prodromakis, G. Indiveri, and B. Linares-Barranco. "STDP and STDP variations with memristors for spiking neuromorphic learning systems". In: *Frontiers in Neuroscience* 7 (2013).

[9] X. Wang, S. Joshi, S. Savel'ev, et al. "Fully memristive neural networks for pattern classification with unsupervised learning". In: *Nature Electronics* 1.2 (2018), pp. 137–145.

[10] G. Hilson. *IBM Tackles Phase-Change Memory Drift, Resistance*. EETimes. 2015. URL: http://www.eetimes.com/document.asp?doc_id=1326477.

[11] X. Li and D. Yeung. "Application-Level Correctness and its Impact on Fault Tolerance". In: *HPCA*. 2007, pp. 181–192.

[12] H. Cho, L. Leem, and S. Mitra. "ERSA: Error Resilient System Architecture for Probabilistic Applications". In: *IEEE Trans. on CAD of Integrated Circuits and Systems* 31.4 (2012), pp. 546–558.

[13] D. Nowroth, I. Polian, and B. Becker. "A study of cognitive resilience in a JPEG compressor". In: *DSN*. 2008, pp. 32–41.

[14] N. P. Carter et al. "Runnemede: An architecture for Ubiquitous High-Performance Computing". In: *HPCA*. 2013, pp. 198–209.

[15] K. A. Bowman et al. "A 45 nm Resilient Microprocessor Core for Dynamic Variation Tolerance". In: *J. Solid-State Circuits* 46.1 (2011), pp. 194–208.

[16] E. N. Elnozahy, L. Alvisi, Y. Wang, and D. B. Johnson. "A survey of rollback-recovery protocols in message-passing systems". In: *ACM Comput. Surv.* 34.3 (2002), pp. 375–408.

[17] S. Gaba, P. Knag, Z. Zhang, and W. Lu. "Memristive devices for stochastic computing". In: *ISCAS*. 2014, pp. 2592–2595.

[18] F. Neugebauer, I. Polian, and J. P. Hayes. "S-box-based random number generation for stochastic computing". In: *Microprocessors and Microsystems - Embedded Hardware Design* 61 (2018), pp. 316–326.

[19] M. Rastegari, V. Ordonez, J. Redmon, and A. Farhadi. "XNOR-Net: ImageNet Classification Using Binary Convolutional Neural Networks". In: *ECCV (4)*. Vol. 9908. Lecture Notes in Computer Science. 2016, pp. 525–542.

[20] Z. Du et al. "ShiDianNao: Shifting vision processing closer to the sensor". In: *2015 ACM/IEEE 42nd Annual International Symposium on Computer Architecture (ISCA)*. June 2015, pp. 92–104.

[21] V. T. Lee, A. Alaghi, J. P. Hayes, V. Sathe, and L. Ceze. "Energy-efficient hybrid stochastic-binary neural networks for near-sensor computing". In: *DATE*. 2017, pp. 13–18.

[22] B. Rajendran and F. Alibart. "Neuromorphic Computing Based on Emerging Memory Technologies". In: *IEEE J. Emerg. Sel. Topics Circuits Syst.* 6.2 (2016), pp. 198–211.

10 Perspectives of Memristor Technologies for Application Domains

Due to the characteristics of memristors, namely the non-volatility and low-energy demand compared to conventional memories as well as their compatibility to analog and bio-medical sensors, many new application opportunities arise with the use of memristors. A promising solution for various application domains is a well-tailored partitioning into memristor-based and classical CMOS-based components to cover dedicated requirements of each application domain. In many application domains, the computation can be partitioned into edge computing (near-sensor computing) and central processing, which can be implemented very differently when using memristor devices.

Another important aspect is caused by the recent breakthroughs in training large neural networks ("deep learning"), which advances the applicability of artificial intelligence in many domains with unprecedented pace. In contrast to its biological origin, however, high performance of artificial neural networks critically relies on much higher energy demands. While the average energy demand of the entire human brain is about 20W, artificial intelligence often resorts to large HPCs with several orders of magnitude higher energy demand of around 20MW. Therefore, a low-energy AI hardware is needed as a core building block for a large variety of application domains (e.g. smart sensor systems, autonomous vehicles, human-centric intelligent assistant systems, robots, IoT devices, etc.).

10.1 Memristors in HPC, Servers, and HPDA

Most obvious and already in progress is usage of memristor technology in the memory hierarchy of supercomputers, high-performance servers, and potentially also HPDA farms (see chapter 4). As recent example for the server domain, Dell EMC announced upgrades to its PowerEdge server line with with adoption of the new Intel Xeon Scalable processors and Intel Optane DC persistent memory [1].

Applications for memristor-based near- and in-memory computing accelerators could be data intensive applications often categorized as "Big Data" workloads. Such accelerators are especially fitting for data analytics, as they provide immense bandwidth to memory-resident data and dramatically reduce data movement, the main source of energy consumption. Analytic engines for business intelligence are increasingly memory resident to minimize query response time [2]. Graph traversal applications are fitting well due to their unpredictable memory access patterns and high ratio of memory access to computation. Such algorithms are common in social network analysis as e.g. Average Teenage Follower (ATF) that counts for each vertex the number of its teenage followers by iterating over all teenager, in Breadth-First Search (BFS), PageRank (PR, and Bellman-Ford Shortest Path (BF) [3].

Memristors in the memory hierarchy may also be applied for checkpointing of application software or as memristor memories/storage for financial applications where "Doing a transaction quickly is important, but having a record is just as important" [4].

10.2 Non-Volatile Processors

Non-Volatile-Processors (NVPs) apply memristors not only to render main memory, caches, and register files *non-volatile* but, in principle, also every Flip-Flop (FF) within the electronic circuitry. NVPs are an important research topic in the field of IoT, Wearable Healthcare Devices (WHDs), and other small embedded devices exploiting energy harvesting. Their main advantage is the possibility to continue computation after no-power standby or a power outage without loss of data and computation time.

As memristors are at a very early research state, large improvements for non-volatile FlipFlops (nvFFs) are also expected. These include faster access times, lower power consumption, and smaller fabrication sizes. Each improvement alone is going to be a game changer for all devices employing energy harvesting or depending on minimal power consumption like IoT, health-care, building automation, and wearables. It is even possible that some ideas of NVPs are going to

find their way into other application areas like the mobile-phone, automotive, or space area.

If the access times of memristor-based nvFFs decreases sufficiently, *real* nvFFs without store and restore technology would become possible (see chapter 8). This will decrease the size of NVPs even more and simplify the integration of Non-Volatile (NV) features in all different kinds of devices. When real nvFFs are applied, a nvFF controller is superfluous and can completely be removed.

If every component of a device, e.g. a Central-Processing-Unit (CPU), is NV, these components can independently be powered off or on according to the current computation and combine power consumption with computational progress. For this scenario new resource scheduling algorithms are required. It is also necessary to make sure data is actually deleted, for example on a pipeline flush or a system reset.

The possible improvements described so far also have an impact on Operating-System (OS) development. OSs running on a NVP can rely on the NV nature of all system resources. Therefore, the OS can support the hardware resource manager. Saving system data to a hard-disk is not required anymore. A large problem to solve is the deletion or storage of cryptographic keys (see chapter 5). Current OSs rely on the deletion of all cryptographic data upon system reset, this is not the case for NVPs. Therefore, cryptographic keys or plain text can be read from memory or even from CPU registers after a reset.

Research in the field of NVPs include processor architecture issues and power management. The focus of NVPs is computational progress under difficult power situations, not performance as is the case for standard processor cores.

10.3 IoT and Wireless Sensors

A broad area of applications which usually has a great need for ultra-low energy demand at moderate computation speed is the Internet of Things (IoT) domain, which can be applied e.g. in the field of smart buildings, industry automation, smart grids, and wearable electronics for sports and health care. IoT devices are equipped with sensors and actuators and need wireless interfaces to connect the world of "things" with the digital world of the Internet.

Wireless sensors suffer from very strong energy requirements, especially in case of energy harvesting applications. Ultra-low power IoT devices can be supported easily by applying the event-triggered computing paradigm. These systems consume no power during stand-by (or sleep mode) and only the necessary processing elements are activated instantaneously when an input event occurs. This is supported inherently by memristors since their last state is kept during power-off, which facilitates a low-power microcontroller implementation when using nvFF (see chapter 8).

- A sensor, collecting data from the environment
- A sensor hub, performing intelligent sensor fusion from multiple sensor sources,
- A micro-controller, interpreting the collected data and deciding what to do with it
- A protocol processing unit, creating data packets to be sent to other nodes
- A transceiver, sending and receiving data packets
- A power management unit, controlling sleep mode and wake-up times of the device
- RAM/ROM/Flash, storing data for local computation, e.g., to identify trends in the data
- An interconnect structure, connecting all the other components

The sensor unit itself can in many cases make use of memristor functionality, e.g., if a value is to be measured over time and can therefore be accumulated. The RAM/ROM/Flash part can be replaced by memristor memory, merging these kinds of memory into one only. Therefore, the power management unit can be shrunk or even omitted. Even the transceiver part can make use of memristor technology [5] and thereby enhancing energy efficiency.

The trend towards smart IoT devices needs to utilize modern deep learning techniques for sensor data processing which in turn require ultra-low power machine learning approaches to be implemented in the device. Memristor-based architectures to build neuromemristive circuits or entire neuromorphic architectures will facilitate the specific requirements for edge computing in IoT devices. An overview of a range of neuromemristive circuits and architectures suitable to be developed as integrated circuit chips in edge computing devices is discussed in [6].

Furthermore, memristors will provide new interfaces to analog sensor components and are well-suited to

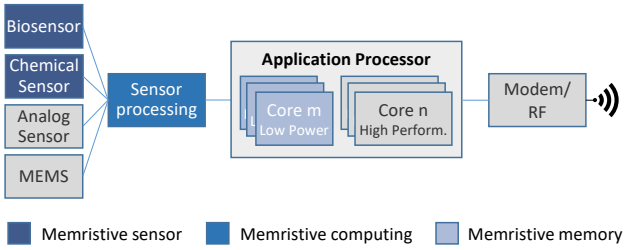


Figure 10.1: IoT platform for smart semsor processing

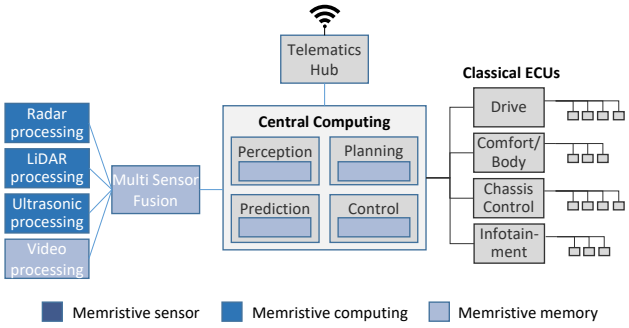


Figure 10.2: E/E architecture of modern autonomous vehicles and potential usage scenarios for memristors

interact with bio-medical interfaces. There already exist approaches to exploit the memristive behavior in freestanding nanowires which are functionalized by use of antibodies populating all-around the free-standing nanowire [7]. The antibodies act as a virtual all-around bio-gate in the memristive biosensor e.g. for drug monitoring or cancer detection.

A generic IoT platform for smart semsor processing is depicted in Figure 10.1. Application scenarios for memristors in sensor, memory, and computation blocks are highlighted.

10.4 Automotive, Avionics, and Space Technologies

In the automotive area, the computing power and the number of sensors per vehicle is steadily increasing. Many sensors are installed far-out from its processing nodes. The overall efficiency of the sensors and devices has to be improved to prevent a decrease in driving range. Therefore, new approaches leading to lower energy demand in local computation as well as in distributed computation via a vehicle network are desired. Local non-volatile memory removes the necessity of a global storage unit. Near-sensor computing avoids unnecessary interconnection activities. Even low power, high performance machine learning capabilities seem possible using ReRAM based SNNs. (Analog) neuromorphic processing also shows a promising robustness against noise.

Current estimates show that the growing number of sensors required for the next stages of autonomous driving will lead to an increase in power consumption of almost 1000W. For instance, state-of-the-art autonomous test vehicles have a power consumption of more than 3000W [8]. Despite major developments in deep learning, energy efficiency is still some orders of magnitude away from feasibility. Current research deals with the transfer of deep learning into the embedded domain. In particular, spiking neural networks and binary neural nets show promising results. Neuromorphic computing will move closer to its biological origin. ADCs will become ASCs (analog spike converter) and near-sensor computing will become “in-sensor computing”. Memristor-based analog SNNs will process sensor data event-based, i.e. partial changes of the sensor data only cause changes in the corresponding part of the network. This will increase energy-efficiency tremendously.

Furthermore, processor in memory (PIM) architectures enable fast and energy-efficient processing of memory-bound applications. The use of only one type of memory that is already non-volatile and supports low-level lock stepping can simplify device architecture and ease the implementation and verification of functional safety requirements. Active standby enables instant wake-up capabilities, which can be used to reduce the power requirements of safety critical systems. Non-volatile registers boost autonomous vehicles by transforming all computing devices into a kind of flight recorder.

Figure 10.2 shows an E/E architecture of modern autonomous vehicles. In recent years, the concept of the central computing cluster has been established [9]. Today, the central computing cluster exists of multiple high-performance multi-core CPUs and GPUs. The surround sensors are connected via a sensor hub. The idea is to use memristive computing for sensor and sensor hub processing, especially the usage of Memristor-based analog SNNs. The central computing cluster can benefit from the memristive memory architectures. Only the classic ECUs will stay with standard CMOS technology.

Since the car should have a low energy consumption each device added to it is supposed to only need the bare minimum of energy, adding to the overall energy

demand (which is reflected in the gas consumption for the owner). Therefore, approaches leading to low energy demand either (or both) in local activities or in the interconnection of devices are desired. Furthermore, the space needed for common storage devices keeping track of engine optimization parameters, user convenience settings, start and goal of a tour including tracking, etc. could be lessened by use of only one type of memory that is already non-volatile.

A similar trend can be observed in the avionics domain by migrating from a federated architecture to a Integrated Modular Avionics (IMA) architecture [10]. IMA allows to integrate multiple software functions with possibly different criticality levels on single avionic computing resources in order to keep the weight, volume and cost of the avionic architecture within reasonable limits. To ensure high safety requirements, IMA is composed of a set of redundant computing processing modules. Furthermore, in avionics and even more for space applications, radiation-hardened electronics play an important role. Latest research shows, that memristor-based non-volatile SRAM will provide a superior radiation hardness, which may recommend this technology to be used in the avionics and space domain [11].

10.5 Memristors for Applications Requiring Back-Up Memory

In common devices any information which is to be used later, i.e., after a restart of the system, is typically stored in conventional non-volatile memories such as FlashROMs or EEPROMs, acting as backup memories. Examples could be BIOS configuration data on computer main boards or programs loaded into microcontrollers. Another example is the configuration of FPGAs (see chapter 7).

However, the time needed to access (read or write) these memories is longer than for memristors. Data can be saved at specific points in time on purpose, such as checkpointing. Furthermore, if a power outage would occur exactly when the data is written it might be corrupted and thereby unusable.

What would be intended is to have a non-volatile memory that holds any data at all times, enabling the device to be unaffected by sudden power outages at any time. It would further be helpful to have the actual working memory replaced by memristors, which would make these backup memories obsolete, since the current status of any memory would stay the same even if power fails and the current activity could be resumed after power is available again.

There are a few exceptions from lossless power outages, though. Usually protocol-driven connections to other devices will lose this connection after a certain time-out and will have to reconnect or re-establish. Thus in these cases the operation cannot seamlessly be resumed after power outages. Worse, the local device would not even recognize the power outage and assume the connection is still valid. Therefore it might be useful to add some kind of interruption detection method into devices which could then calculate the duration of the interruption (e.g., by comparing the current time to the last time stamps) and deducing whether connections have to be re-established.

Since it is unlikely that in the near future full systems are to be equipped with memristor memory only, a promising approach may be a hybrid memory composed of memristive and conventional memory. It would then become important to identify different criticality level of data processed by the applications. If some data are more valuable (i.e., in case of a data loss it would take a long time to recalculate the data) it should be stored in the memristor memory part, being safe against power outages at any time. Less valuable data (i.e., data that can quickly be reproduced or which represents a snapshot on current information which will be outdated shortly thereafter, such as sensor values) would then mapped into the conventional memory. Algorithms have to be analyzed in order to identify different data criticality. This is another auspicious field of research for the next years to come.

References

[1] D. Black. “Systems Vendors Refresh Product Lines as Intel Launches New Xeon, Optane”. In: *HPCwire, April 2, 2019* (2019). URL: https://www.hpcwire.com/2019/04/02/systems-vendors-refresh-product-lines-as-intel-launches-new-xeon-optane/.

[2] M. Drumond, A. Daglis, N. Mirzadeh, D. Ustiugov, J. Picorel, B. Falsafi, B. Grot, and D. Pnevmatikatos. “The Mondrian Data Engine”. In: *Proceedings of the 44th Annual International Symposium on Computer Architecture*. ACM. 2017, pp. 639–651.

[3] E. Azarkhish, D. Rossi, I. Loi, and L. Benini. “Design and Evaluation of a Processing-in-Memory Architecture for the Smart Memory Cube”. In: *Proceedings of the Architecture of Computing Systems – ARCS 2016, Lecture Notes in Computer Science, vol 9637. Springer*. 2016.

[4] G. Hilson. *Everspin Targets Niches for MRAM*. URL: https://www.eetimes.com/document.asp?doc_id=1332871.

[5] N. Wainstein and S. Kvatinsky. "TIME;Tunable Inductors Using Memristors". In: *IEEE Transactions on Circuits and Systems I: Regular Papers* 65.5 (May 2018), pp. 1505–1515. DOI: 10.1109/TCSI.2017.2760625.

[6] O. Krestinskaya, A. P. James, and L. O. Chua. "Neuromemristive Circuits for Edge Computing: A review". In: *IEEE transactions on neural networks and learning systems* (2019).

[7] S. Carrara, D. Sacchetto, M.-A. Doucey, C. Baj-Rossi, G. D. Micheli, and Y. Leblebici. "Memristive-biosensors: A new detection method by using nanofabricated memristors". In: *Sensors and Actuators B: Chemical* 171-172 (2012), pp. 449–457. DOI: https://doi.org/10.1016/j.snb.2012.04.089. URL: http://www.sciencedirect.com/science/article/pii/S0925400512004583.

[8] S.-C. Lin, Y. Zhang, C.-H. Hsu, M. Skach, M. E. Haque, L. Tang, and J. Mars. "The Architectural Implications of Autonomous Driving: Constraints and Acceleration". In: *Proceedings of the Twenty-Third International Conference on Architectural Support for Programming Languages and Operating Systems*. ASPLOS '18. Williamsburg, VA, USA, 2018, pp. 751–766. DOI: 10.1145/3173162.3173191. URL: http://doi.acm.org/10.1145/3173162.3173191.

[9] M. Mody et al. "Understanding Vehicle E/E Architecture Topologies for Automated Driving: System Partitioning and Tradeoff Parameters". In: *Electronic Imaging* 2018.17 (2018), pp. 358–1.

[10] P. Bieber, F. Boniol, M. Boyer, E. Noulard, and C. Pagetti. "New Challenges for Future Avionic Architectures." In: *AerospaceLab* 4 (2012), p–1.

[11] H. M. Vijay and V. N. Ramakrishnan. "Radiation Effects on Memristor-based Non-volatile SRAM Cells". In: *J. Comput. Electron.* 17.1 (Mar. 2018), pp. 279–287. DOI: 10.1007/s10825-017-1080-x. URL: https://doi.org/10.1007/s10825-017-1080-x.