« Normally off » computing for smart systems

Cache and main memory architecture based on MRAM: application to High Performance Computing and Embedded Systems

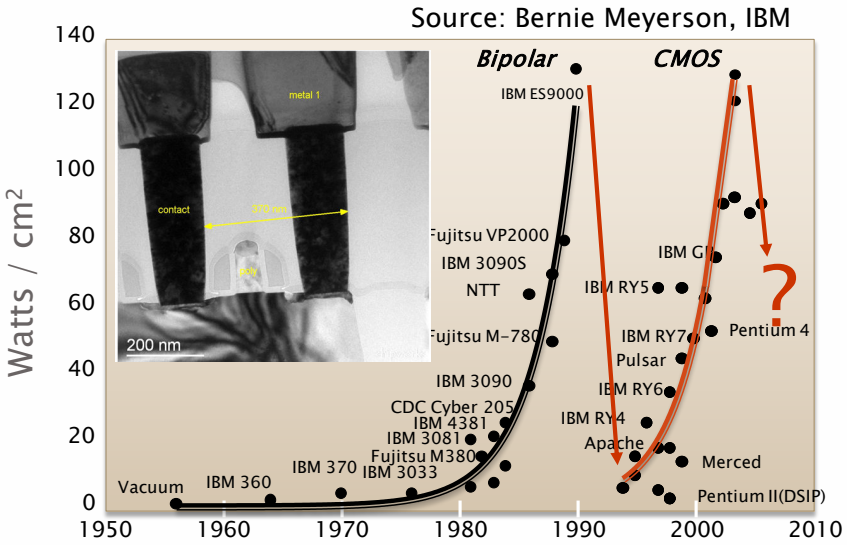
Lionel Torres – Univ. Montpellier, France

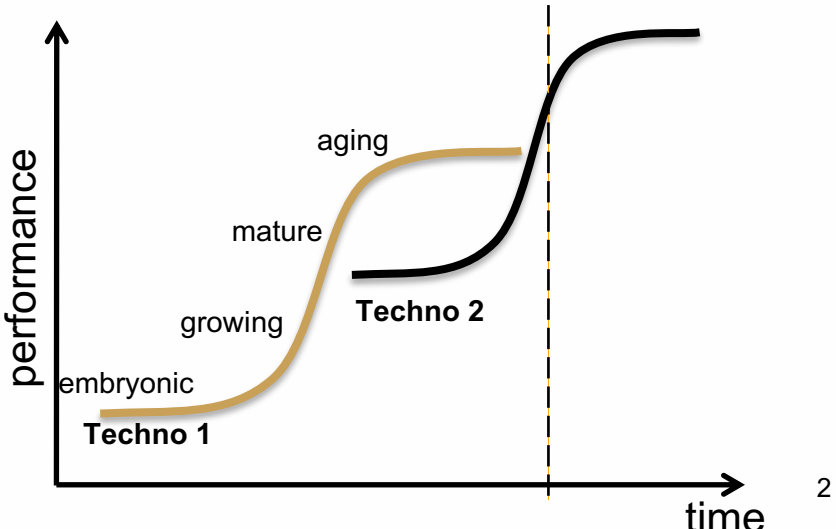
Merci à : G. Sassatelli, A. Gamatie, P. Benoit, P. Nouet, D. Novo, G. Dinatale, A. Todri, A. Virazel, L. Latorre, M. Robert, G. Patrigeon, P.Y. Peneau, J. Modad, F. Ouattara, J. Lopes, O. Coi, K. Sevin

General context

Current IC Integration Challenges

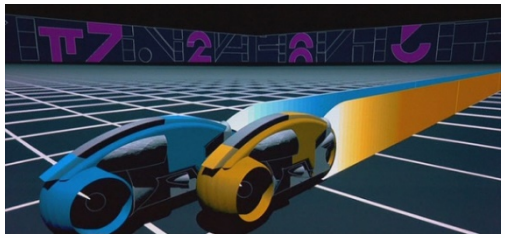
- Energy is **critical**
- We need more and more **Performances** for applications
- Actual technology limitations (CMOS) – **Integration** is more and more complex
  10^9 transistors / cm2
- **Actual Reliability** is a problem– X% of the systems encounter an uncorrectable error per year (X ranging from 1 to 5%)





General context

Technology target : CMOS < 20 nm

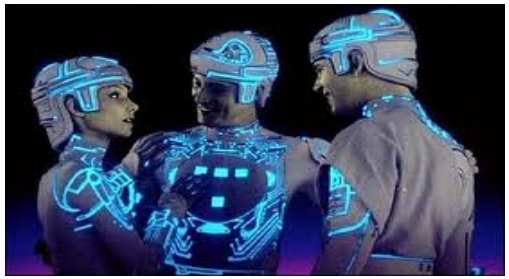


To Transport 1 bit → 1pJ / mm

To transport 10^9 data – 1s (1Ghz) → 1pJ / mm x 10^9 = 1mw / mm

64 Bits Bus → 64mw / mm

On real IC → several W / cm2



Calcul, Bit transition → 1 aJ

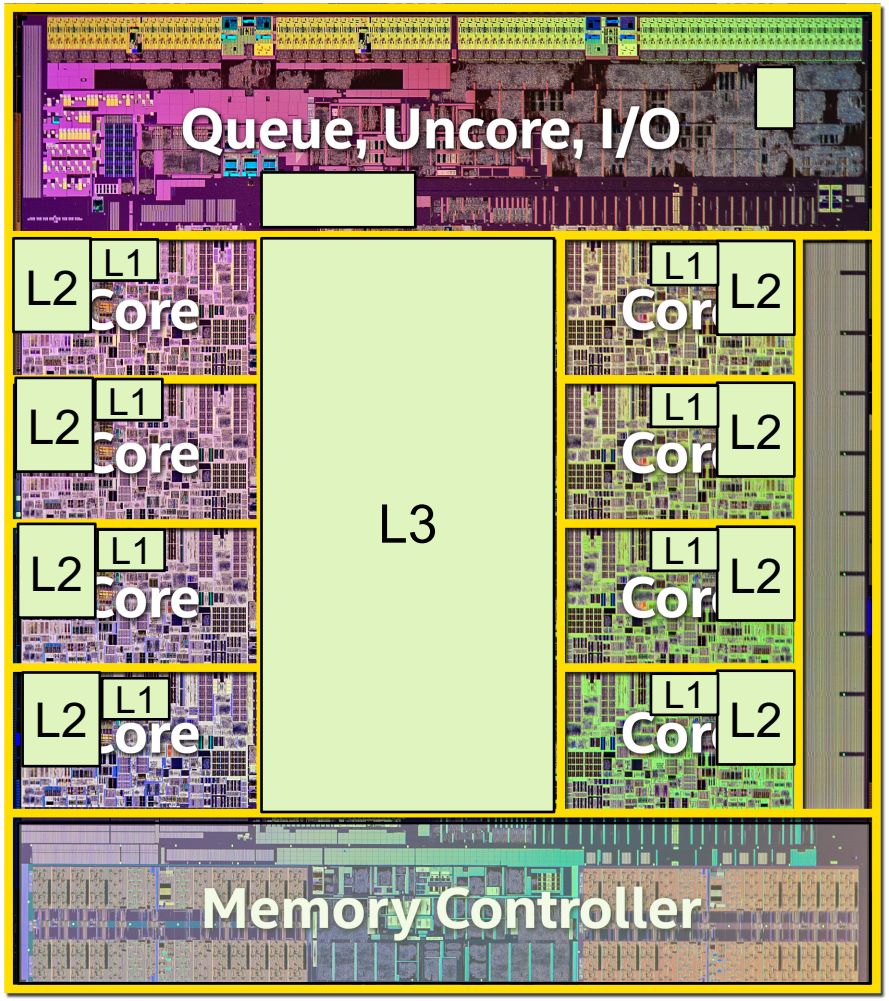
Calcul, 10^9 data transition – 1s → 1aJ x 10^9 = 1nw

→ It is better to ”calculate” than to “transport”’ the information

→ *In computing* memory is certainly interesting

→ Reminder : minimal energy to change 1 bit d’information – K. T Ln2 → 2,85 zJ

One challenge: the memory



- Today, 50% of the silicon area of IC is memory
- Take care to energy (static) !

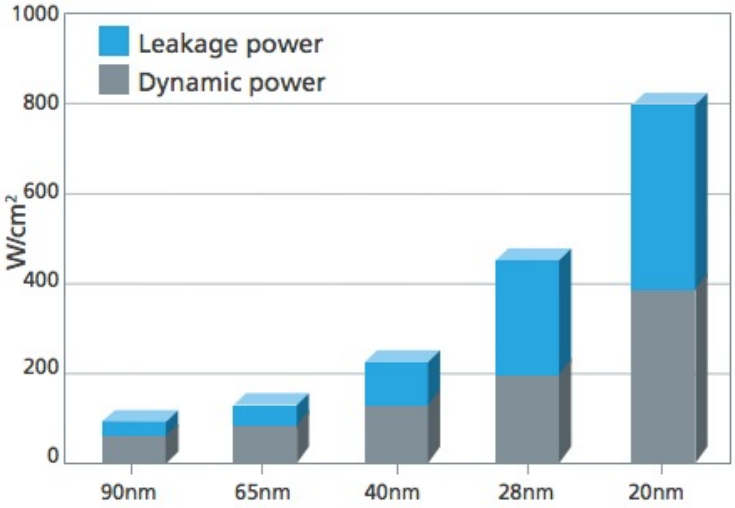
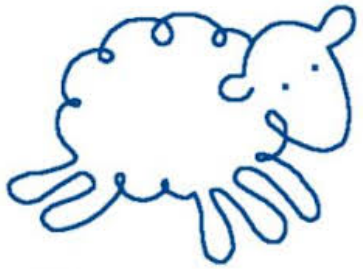


Figure 1: Leakage power becomes a growing problem as demands for more performance and functionality drive chipmakers to nanometer-scale process nodes (Source: IBS).

Technology evolution

Actual memories:

- SRAM for fast access
- DRAM for applications
- Flash (mass storage)
- …



Emerging memories

- Magnetic tunneling junctions
- Phase change memory
- Programmable metallization cells
- OxRRAM
- …

Universal memory:
"Non-volatile memory"

- SRAM performance
- Size of DRAM/Flash
- Non-volatility
- Scalibility

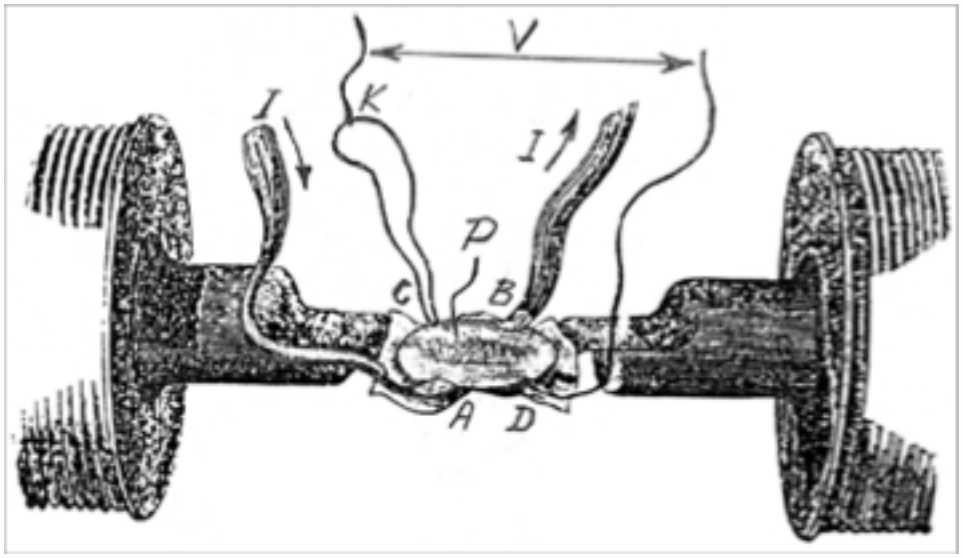
Resistance Switching Memory

Emerging memories offer non-volatility, speed and endurance => disruption of the memory hierarchy?

Spin Technology

Conductance of magnetic metal plates is larger in the presence of a magnetic field perpendicular to the current flow

William Thomson
1824-1907



Currently known as **Anisotropic Magnetoresistance (AMR)**

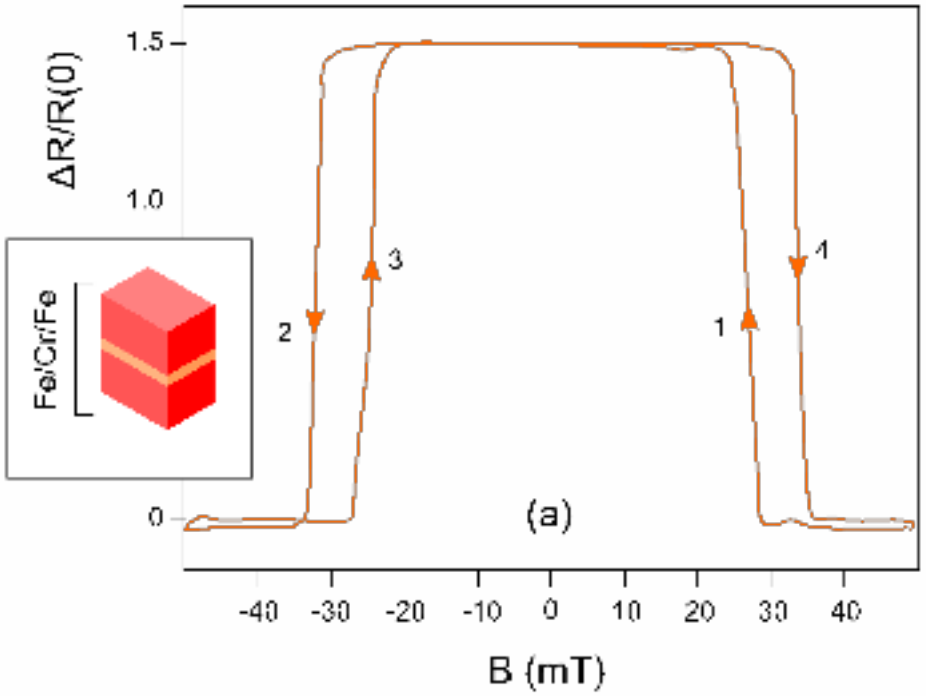
Resistance variation attained: 2%-5% in RT

Spin Technology

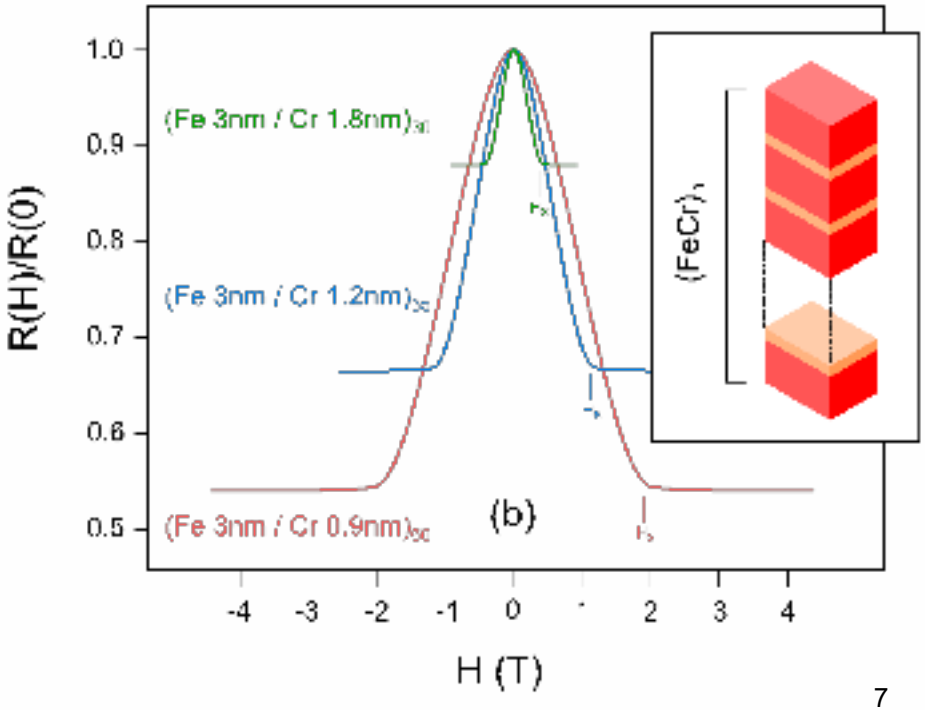
Peter Grünberg and Albert Fert
2007 Nobel Prize in Physics

- Thin stacks of FM/NM metals have seen a conductance increase of up to 100% when subjected to a magnetic field

B. Guinasch et al., 1989

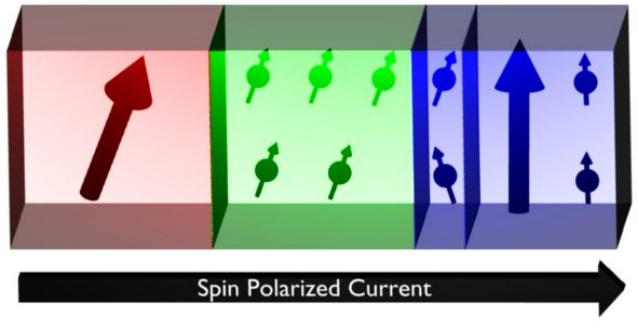


M. N. Baibich et al., 1988



Spin Technology

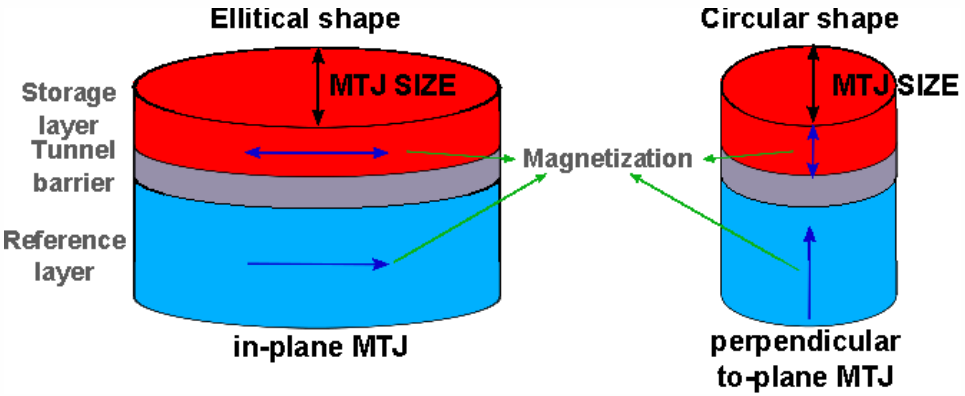
Magnetoresistance Tunnel:



$$R(\theta) = RP + \Delta_R \frac{(1 - cos(\theta))}{2}$$

$$\theta = 0 \rightarrow R = RP$$

$$\theta = 180 \rightarrow R = RP + \Delta_R = RAP$$

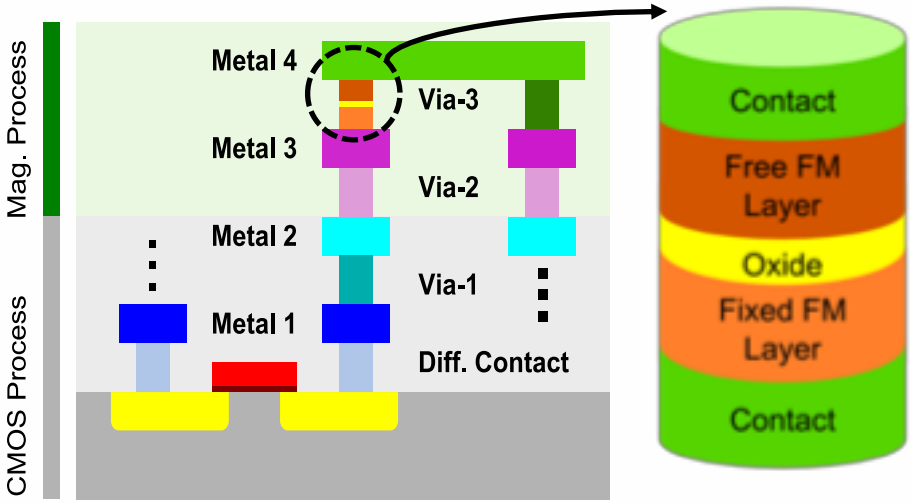


$$TMR(\%) = \frac{\Delta_R}{R_P} = \frac{R_{AP} - RP}{R_P}$$

TMR Classique between 150% et 250% (or more)

M. Bowen et al. Nearly total spin polarization…

Spin Technology



Compatible with CMOS

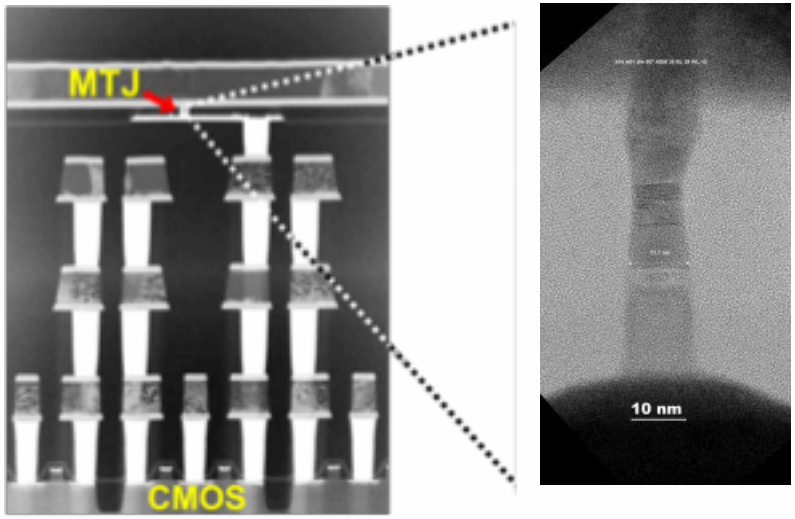
Non–volatile memory

Switching time < 1ns

writing current < 10uA-100uA

density x4 vs SRAM

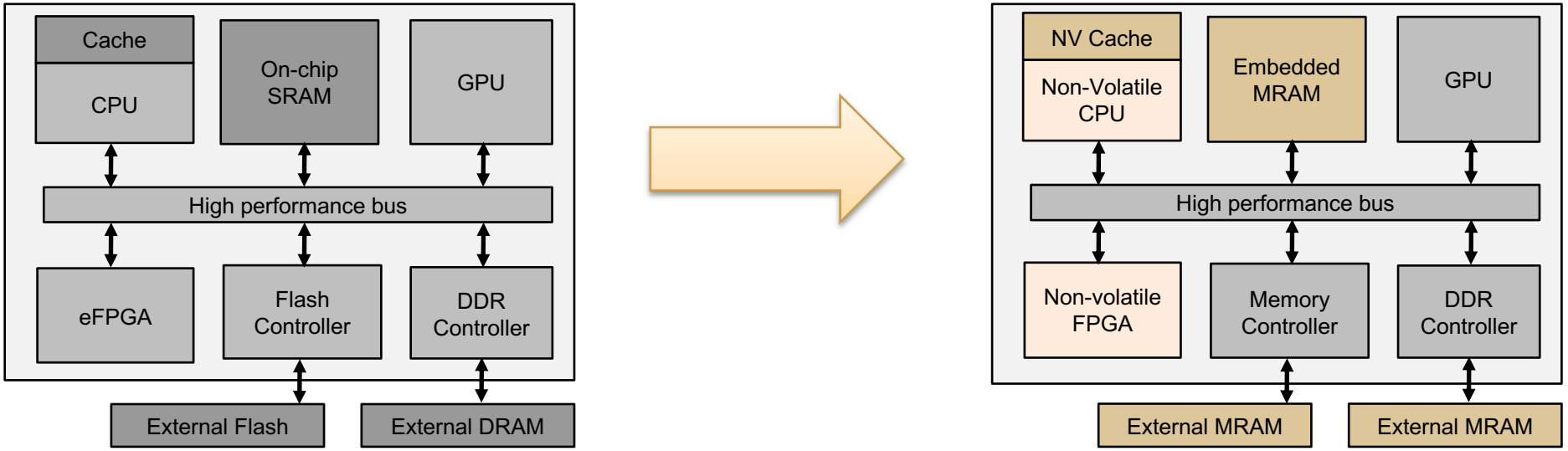
Immune to radiations



Samsung demonstrator
(8 Mbit STT–MRAM) – 2016

Motivations

- **A way**
  - Go towards non-volatile systems using emerging NVMs
  - Current NVMs issues : Speed, Dynamic energy, Reliability



Where and how to place MRAM to:

reduce total power consumption ?

keep same or get better performance ?

Contributions

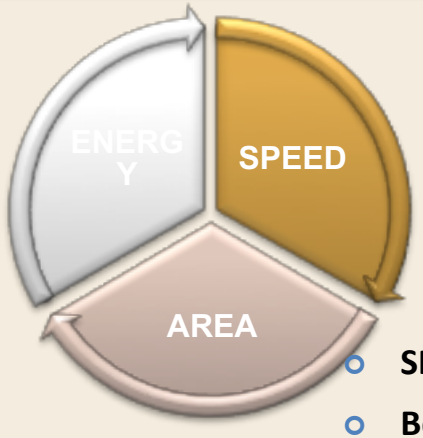
1. Evaluation of MRAM-based cache memory hierarchy:
   - Exploration flow and extraction of memory activity
   - L1 and L2 caches based on STT-MRAM and TAS-MRAM

2. Non-volatile computing
   - *Instant-on/off* capability for embedded processor
   - Analysis and validation of *Rollback* mechanism

MRAM applied to cache

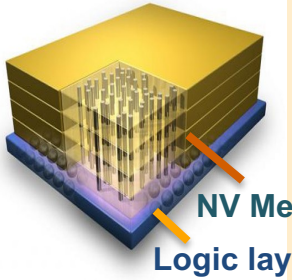
Possible studies

Performance comparison

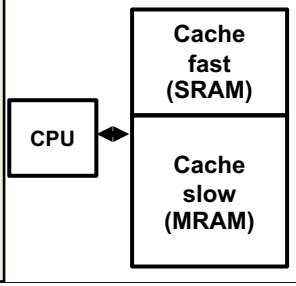


- **SRAM vs MRAM**
- **Benefits of MRAM**
  - Low leakage
  - High density
  - Non-volatility

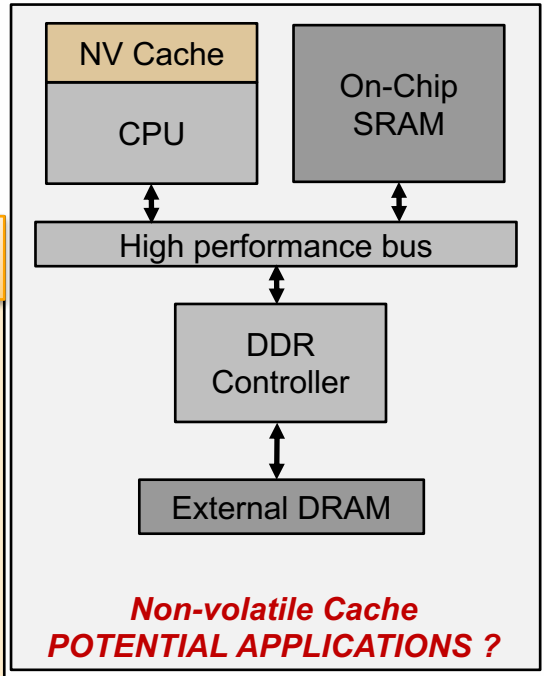
New architectures



3D-Stacking capability of MRAM



Hybrid SRAM/MRAM cache



Non-volatile Cache POTENTIAL APPLICATIONS ?

Take advantages of MRAM

Low leakage
High density
Non-volatility

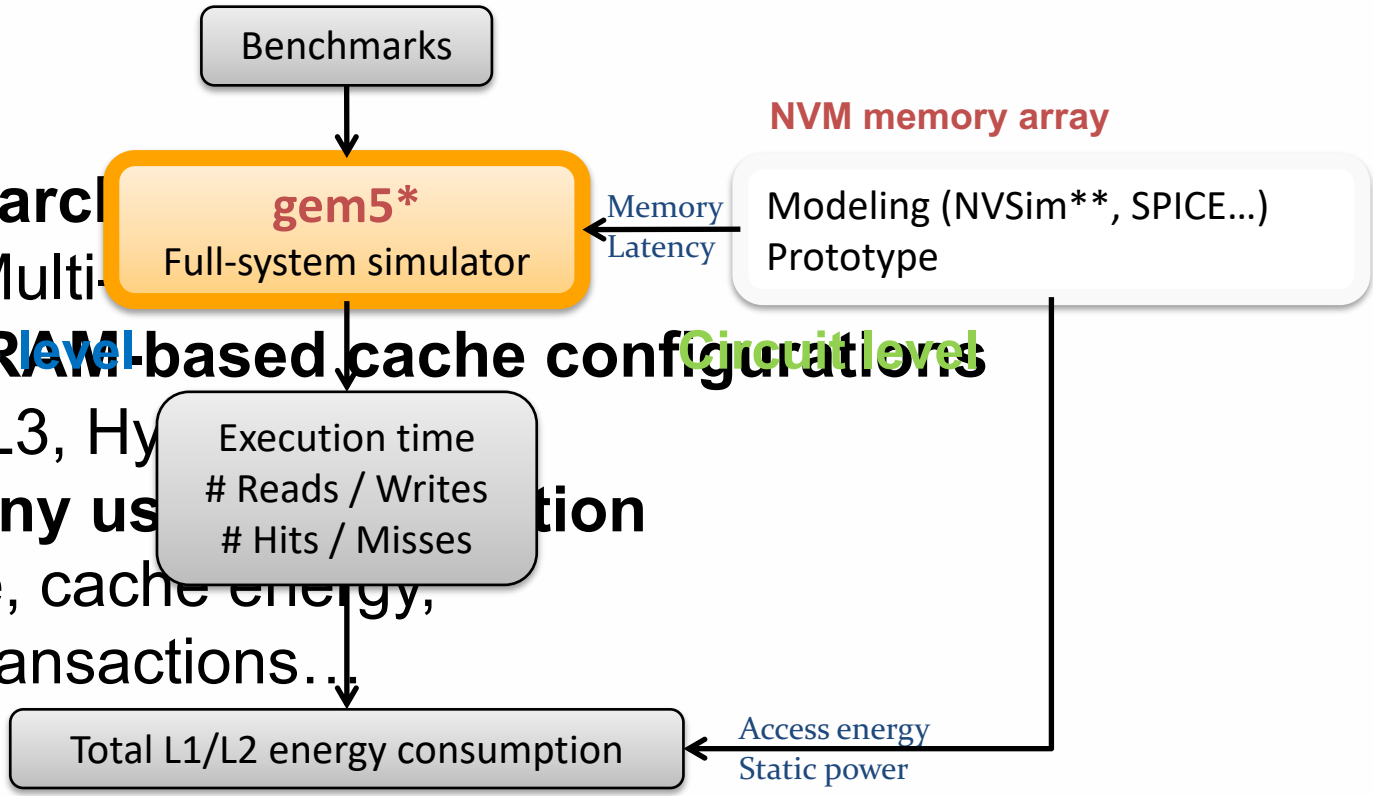
Mitigate drawbacks of MRAM

High write latency
High write energy

MRAM applied to cache

NVM exploration flow

1. **Define the arch**
   - Single/Multi-
2. **Explore MRAM-based cache configurations**
   - L1, L2, L3, Hy
3. **Extract many us tion**
   - Runtime, cache energy, cache transactions…



* N. Binkert et al., "**The gem5 simulator**," ACM SIGARCH Computer Architecture News, Aug. 2011.

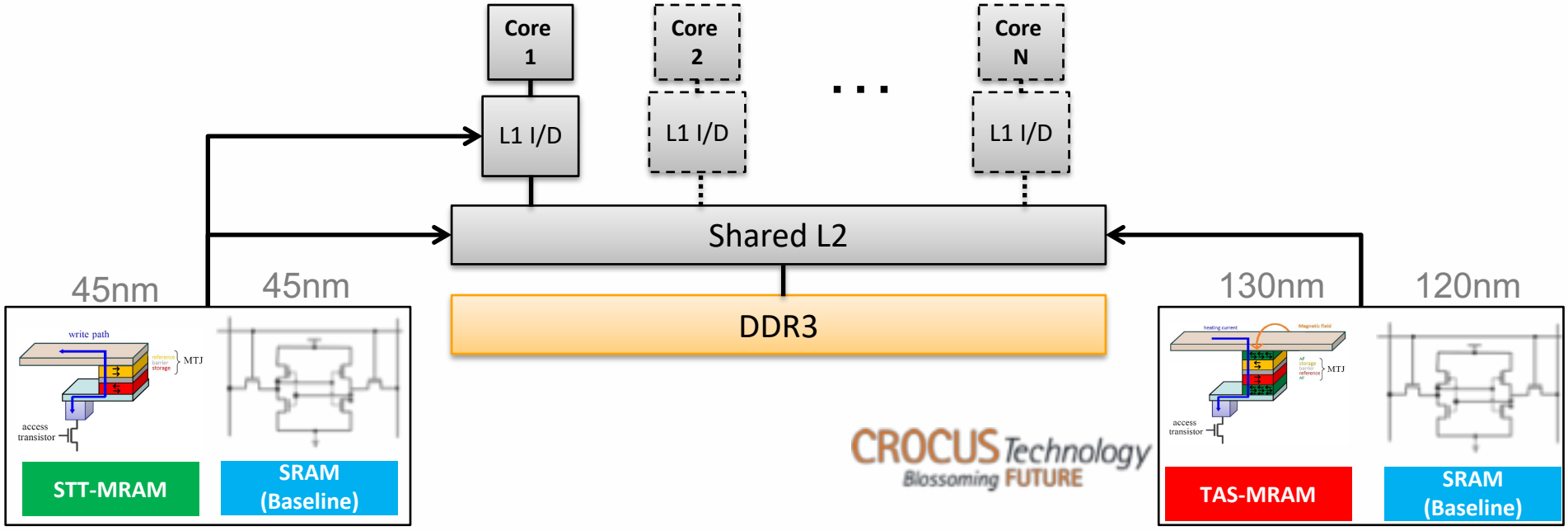
** X. Dong et al., "**NVSim: A Circuit-Level Performance, Energy, and Area Model for Emerging Nonvolatile Memory**," IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Jul. 2012 .

MRAM applied to cache

Experimental setup

From single to multi-core architecture

ARMv7 ISA
Private L1 instruction/Data
Shared L2
(Additional levels of caches possible)
Main Memory

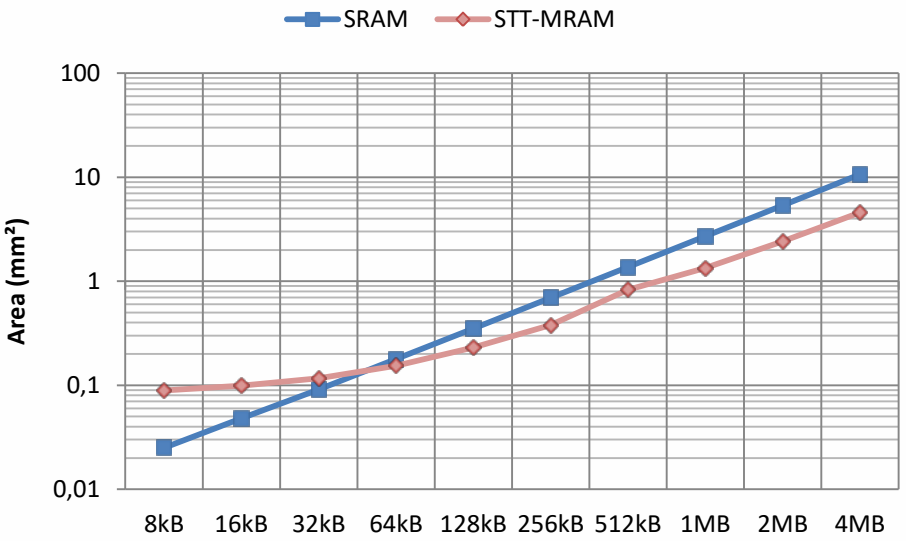


14

MRAM applied to cache

Circuit-level analysis:
Models (NVSim) & Prototype

Area



Node	Technology	512kB L2 (mm²)	32kB L1 (mm²)
45nm	SRAM	1.36	0.091
	STT-MRAM	0.82	0.117
120nm	SRAM	9.7	-
	TAS-MRAM	11.7	

- MRAM is denser for large cache capacity
- MRAM cell size smaller than that of SRAM
- MRAM needs large transistors for write
- TAS-MRAM cache larger due to field lines

MRAM applied to cache

Circuit-level analysis: Models (NVSim) & Prototype

512kB L2 cache

		Read		Write		Standby
Node	Technology	Latency (ns)	Energy (nJ)	Latency (ns)	Energy (nJ)	Leakage (mW)
45nm	SRAM	4.28	0.27	2.87	0.02	320
	STT-MRAM	2.61	0.28	6.25	0.05	23
120nm	SRAM	5.95	1.05	4.14	0.08	82
	TAS-MRAM	35	1.96	35	4.62	10

/2.2 ≈ | 2.1 | 2.5 | /14 | /8

STT-MRAM ≈ SRAM
TAS-MRAM > SRAM

MRAM > SRAM

MRAM << SRAM

32kB L1 cache

Node	Technology	Latency (ns)	Energy (nJ)	Latency (ns)	Energy (nJ)	Leakage (mW)
45nm	SRAM	1.25	0.024	1.05	0.006	22
	STT-MRAM	1.94	0.095	5.94	0.04	3.3

/7

MRAM > SRAM

MRAM > SRAM

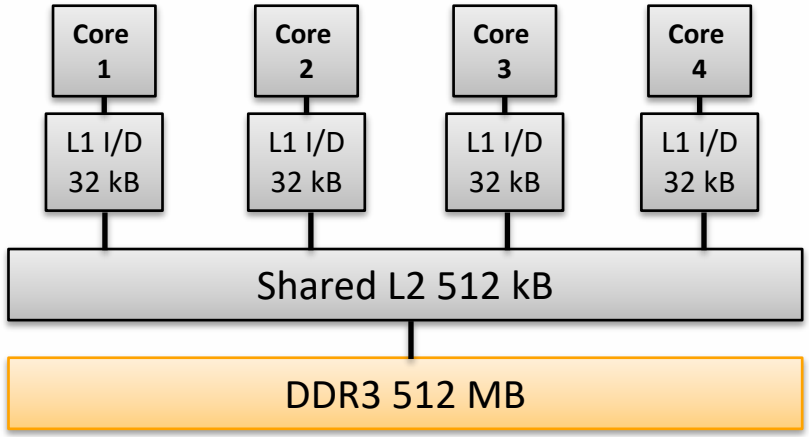
MRAM << SRAM

MRAM applied to cache

Case study

Quad-core architecture:

- Frequency 1GHz
- ARMv7 ISA
- Private L1 I/D
- Shared L2
- DDR3 Main memory



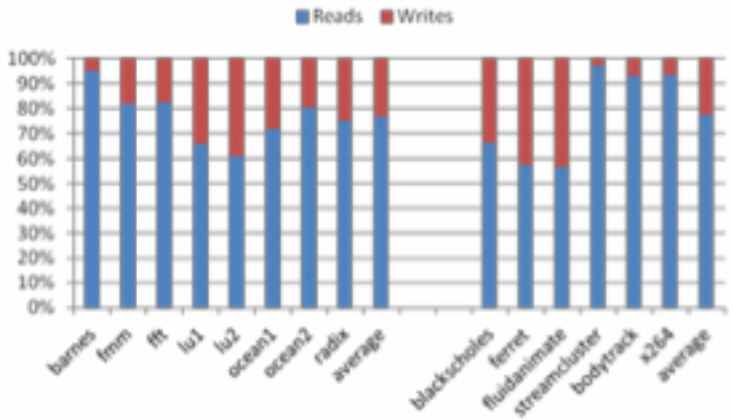
Benchmarks

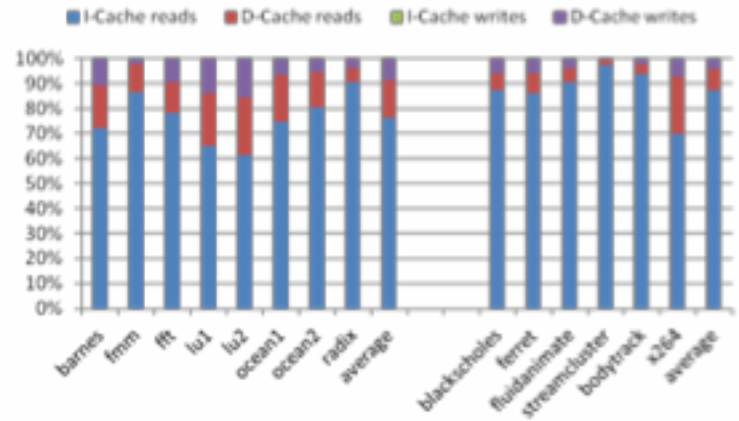
- SPLASH-2
  - Mostly high performance computing
- PARSEC
  - Animation, data mining, computer vision, media processing

MRAM applied to cache

Architecture-level analysis:
gem5

Read/Write ratio





L2/L1 access ratio

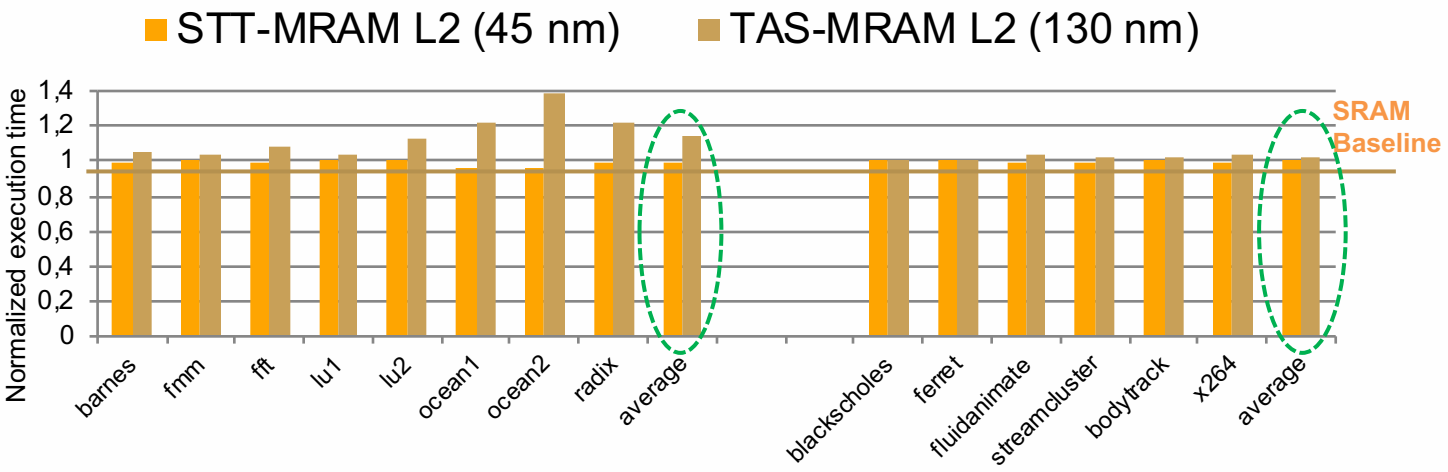
Benchmark	Number of accesses	
	L1 cache	L2 cache
SPLASH-2	~2 billions (0.5 billions/CPU)	~26 millions
PARSEC	~12 billions (3 billions /CPU)	~16 millions

Static/Dynamic energy ratio

Static energy

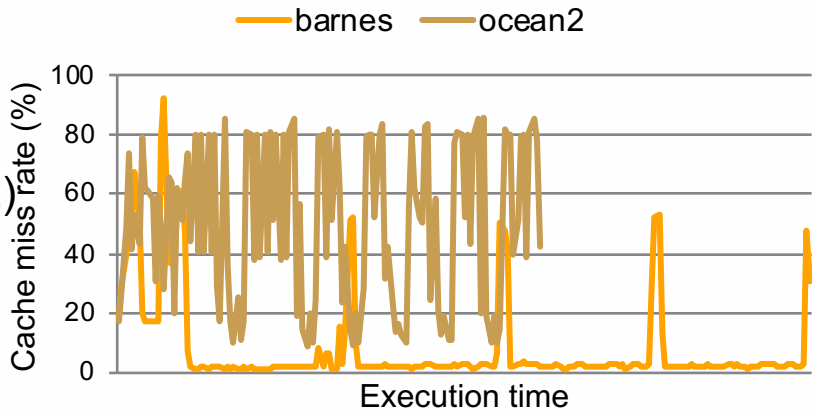
L2 → 90%
L1 → 80%

Execution time



Observations:

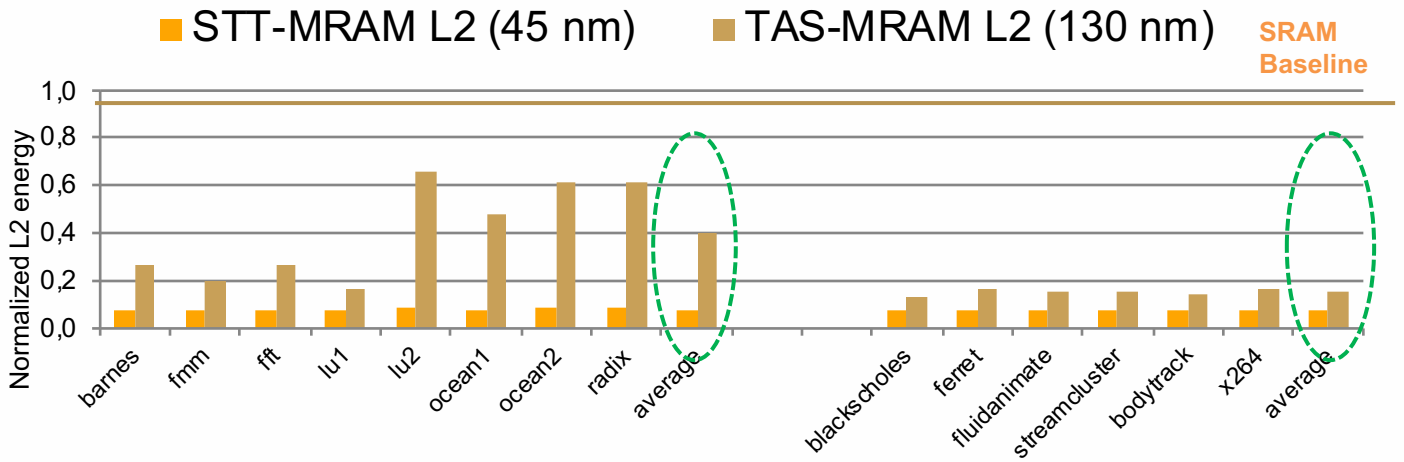
- STT shows good performance
  - L2 has small impact in overall performance
- For TAS, 14% of penalty in average (SPLASH-2)
  - Depends on applications
    (Cache miss rate, L1/L2 access ratio)



19

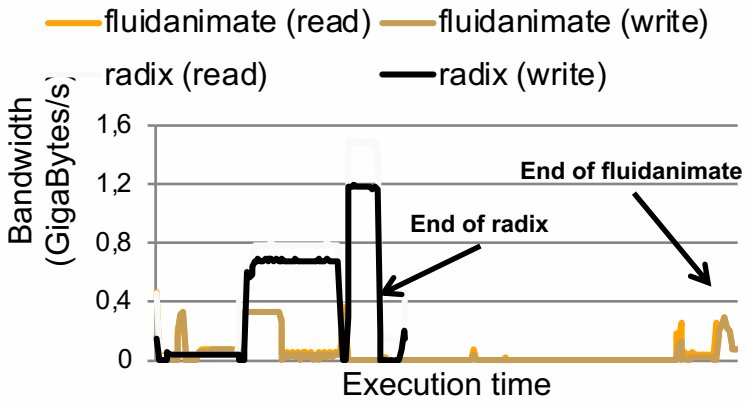
MRAM-based L2

Total L2 cache energy consumption



Observations:

- Up to 90% of gain for STT
- From 40% to 90% for TAS
  - Due to the very low leakage of MRAM-based cache



20

Extension to this work

STT-RAM designs with **different data retention times** [1]

	Design 1	Design 2	Design 3
Cell size (F^2)	23	22	27.3
MTJ sw time (ns)	10	5	1.5
Retention Time	4.27yr	3.24s	26.5μs
Write Latency (ns)	10.378	5.370	1.500
Write Dyn. Eng(nJ)	0.958	0.466	0.187

Given a multi-bank STT-RAM memory, where each bank has customized retention time, how to suitably allocate data in the memory?

=> lifetime analysis for program variables to decide their mapping (**see talks in «Timing Analysis» Session** @ RTNS'2018) - *Rabab Bouziane, Erven Rohou and Abdoulaye Gamatie*

[1] Q. Li et al. "Compiler--Assisted Refresh Minimization for Volatile STT-RAM Cache," in *IEEE Transactions on Computers*, vol. 64, no. 8, 2015.

Is MRAM suitable for cache ?

- Good candidate for lower level of cache (L2 or last level cache)
  - Up to 90% of energy gain
  - No or small performance penalty
  - More memory capacity using MRAM
  - Cache L2 is up of 20% energy consumption of overall system
- Not suitable for upper level of cache (L1) for high performance – but depending of the application some gain in energy
  - Micro-architectural modifications required to mask latency
  - Not detailed in this presentation but full evaluation of cache L1 done too

22

Contributions

1. Evaluation of MRAM-based cache memory hierarchy:
   - Exploration flow and extraction of memory activity
   - L1 and L2 caches based on STT-MRAM and TAS-MRAM

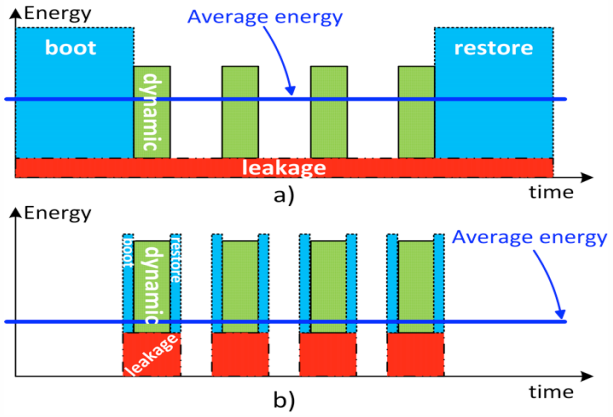
2. Non-volatile computing
   - *Instant-on/off* capability for embedded processor
   - Analysis and validation of *Rollback* mechanism

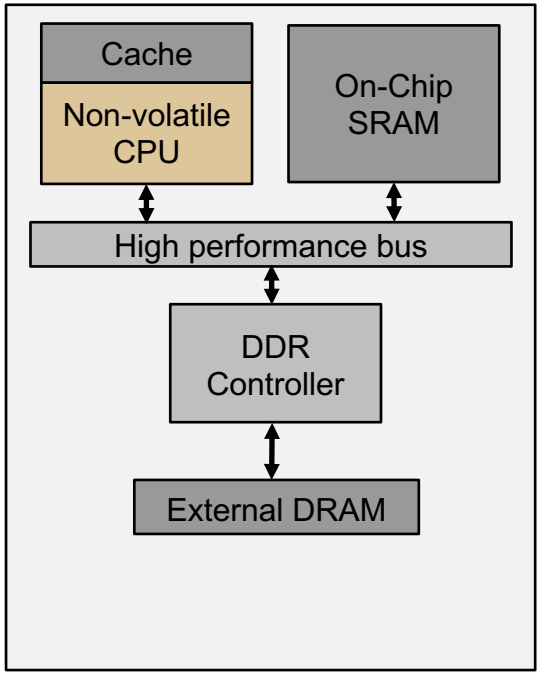
23

Normally-off computing

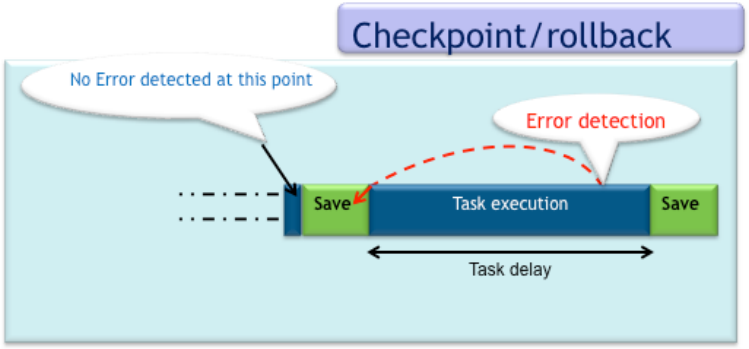
Two concepts:

- Instant on/off
  - Restore processor state





- Backward error recovery (Rollback)
  - Restore previous valid state

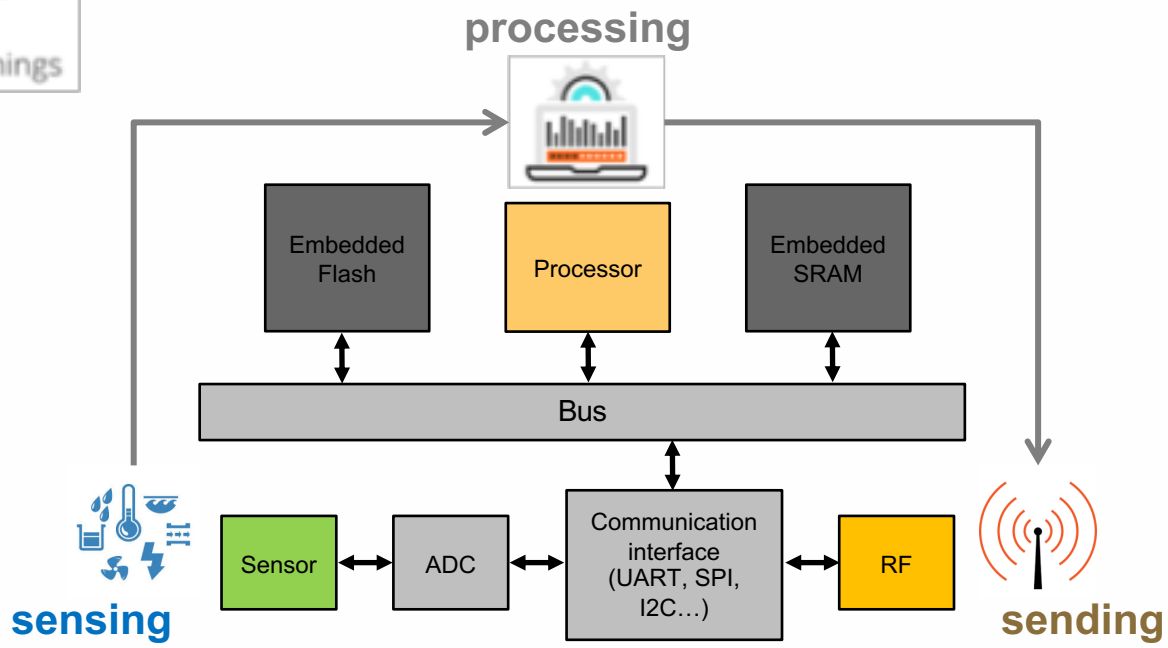


24

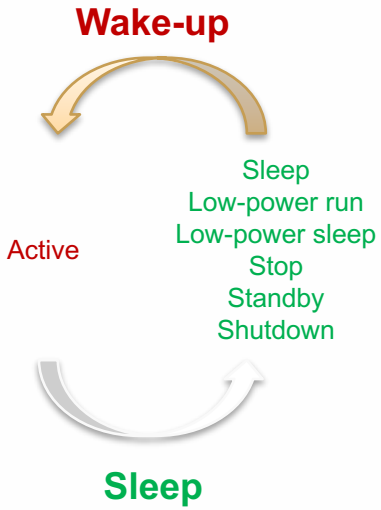
Traditional microcontrollers (MCU)

Internet of Things

Architecture



Power modes



Many power modes
(Up to nine for some MCUs)

Two memories
(SRAM & Flash)

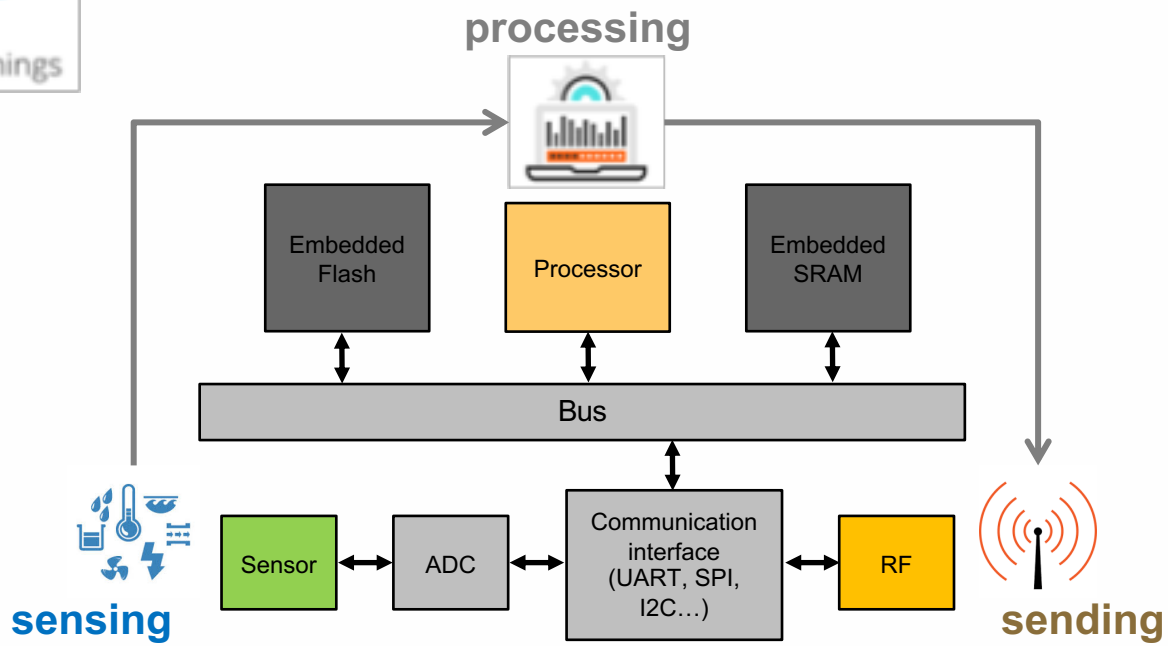
Energy / Wake-up time tradeoff
(Low leakage or fast wake-up ?)

Volatility issue
(Execution state loss)

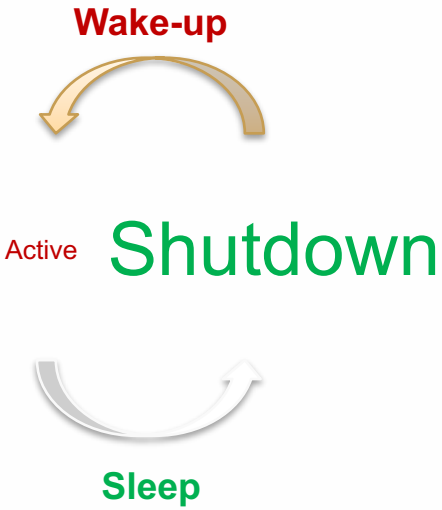
Traditional microcontrollers (MCU)

Internet of Things

Architecture



Power modes



Many power modes
(Up to nine for some MCUs)

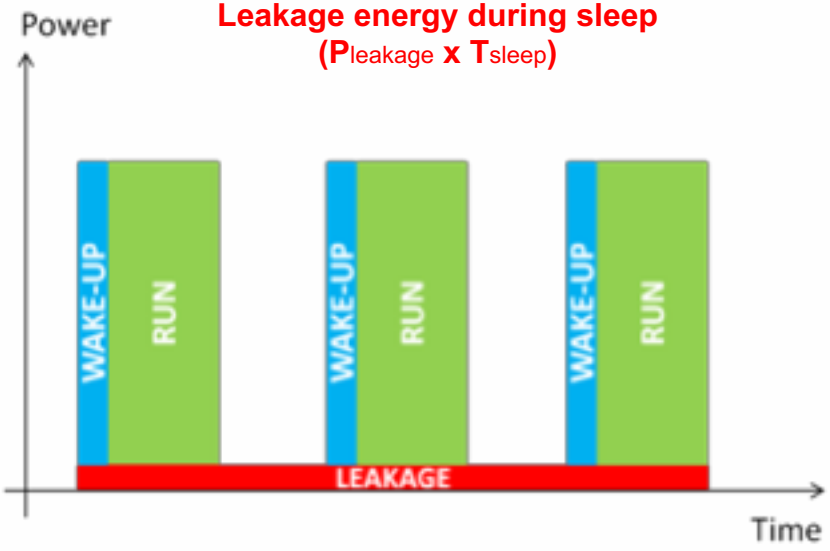
Two memories
(SRAM & Flash)

Energy / Wake-up time tradeoff
(Low leakage or fast wake-up ?)

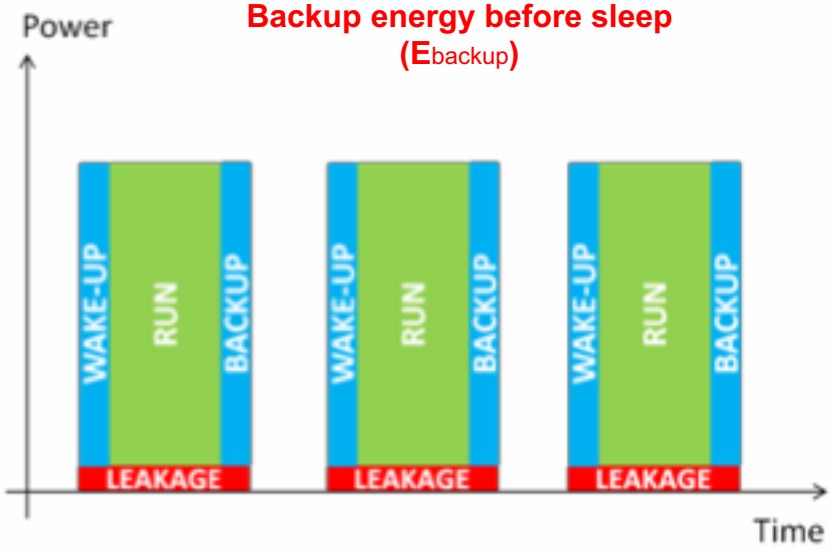
Volatility issue
(Execution state loss)

MCU Energy principle

Traditional MCU



Non-volatile MCU

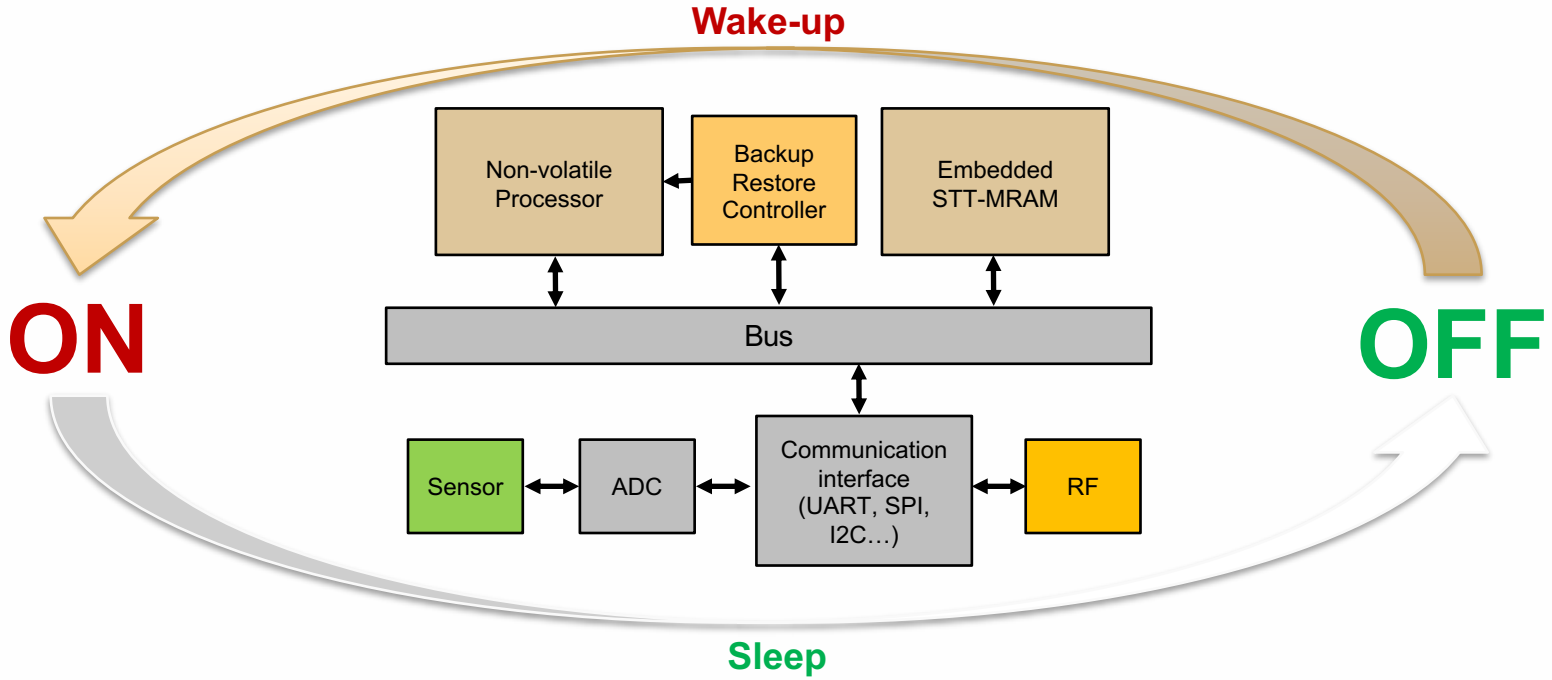


Minimum T_{sleep} required to be more energy efficient ?

$$E_{backup} < P_{leakage} \times T_{sleep}$$

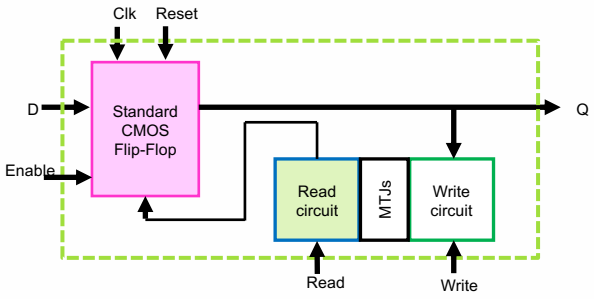
$$E_{backup} / P_{leakage} < T_{sleep}$$

MCU based on STT-MTJ



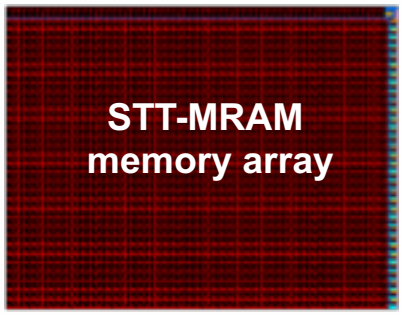
Non-volatile Flip-Flop (NVFF)

→ Store the processor state



STT-MRAM Memory

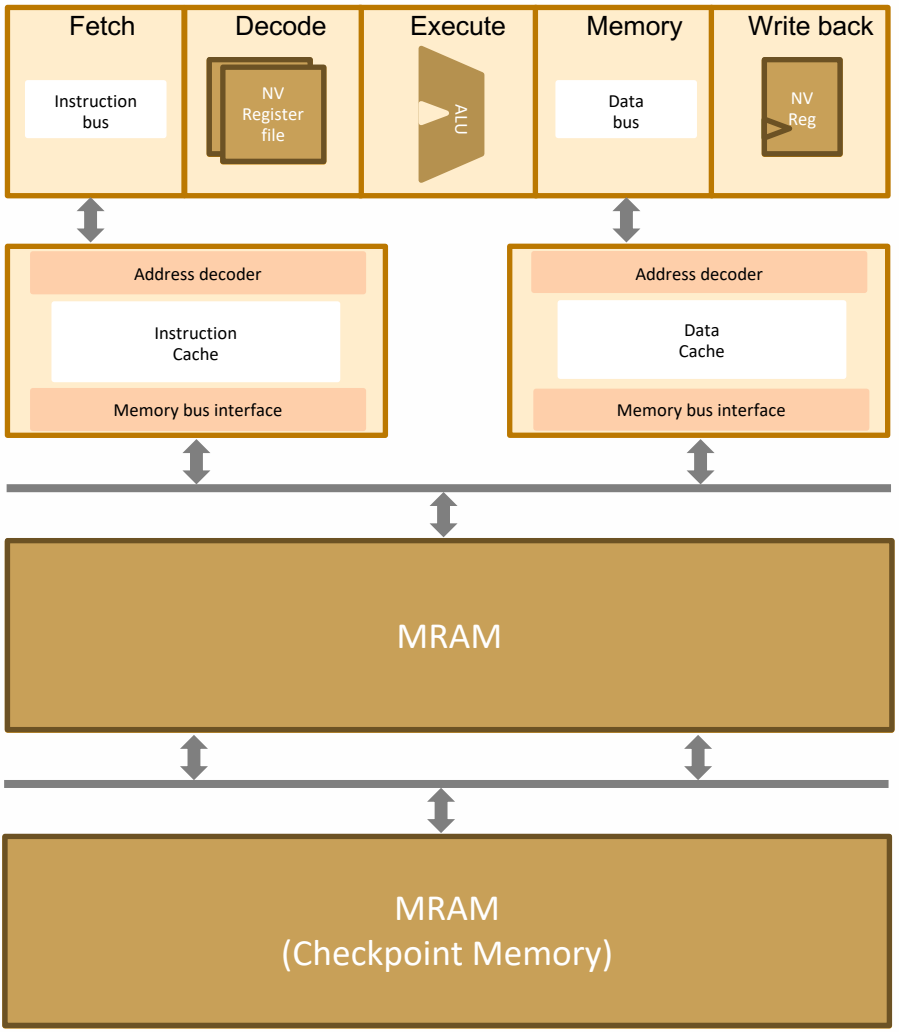
→ Store program & data



Backup/Restore Controller

→ Backup/Restore the FFs

MRAM-based processor



Instant on/off & Rollback

Non-volatile register

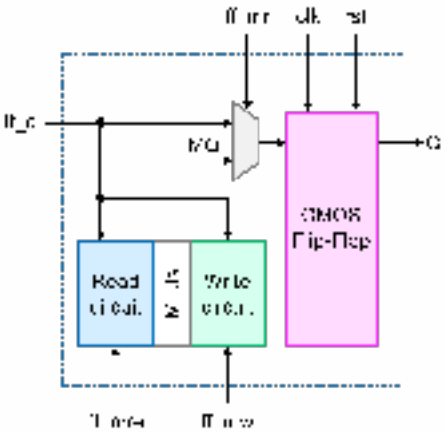
+

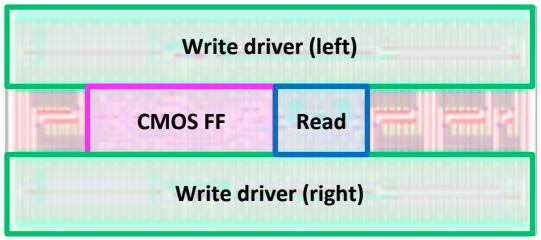
Non-volatile Memory

+

Checkpoint Memory (Rollback)

MRAM-based register



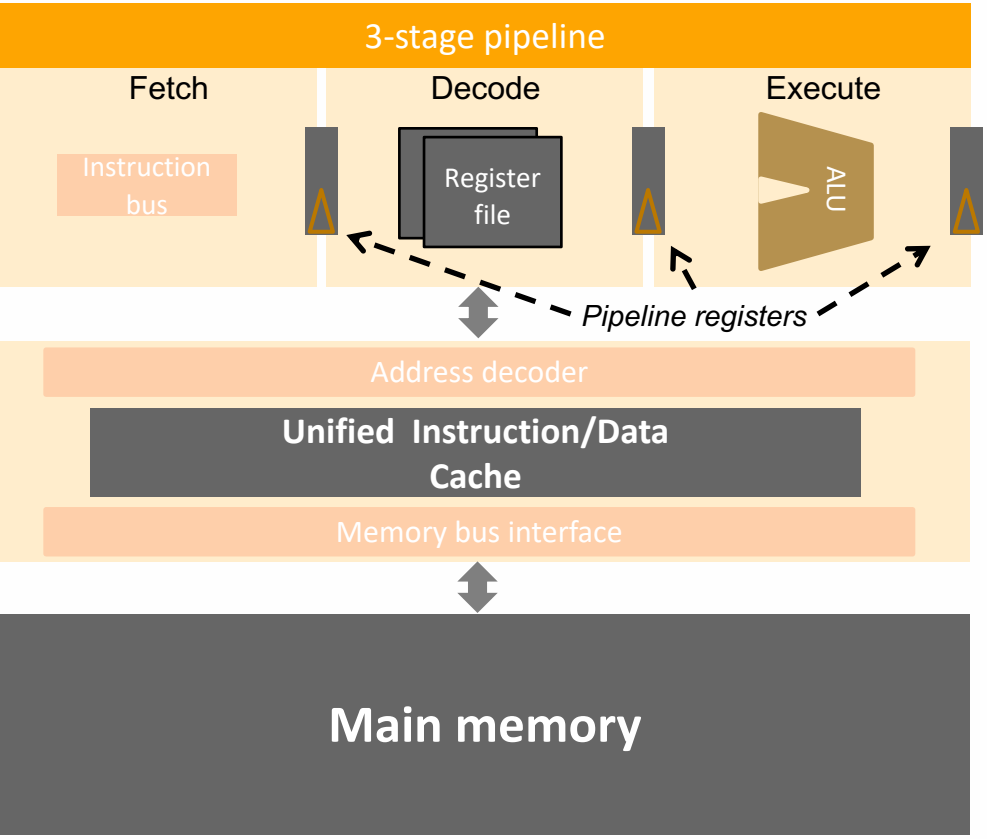


Layout of NV Flip-Flop
(28nm FDSOI, 90nm STT)

B. Jovanovic, R. Brum, L. Torres, *Comparative Analysis of MTJ/CMOS Hybrid Cells based on TAS and In-plane STT Magnetic Tunnel Junctions*, **IEEE Transactions on Magnetic**,, 2014.

MRAM-based processor

First Case study: Amber 23 processor (ARM based instruction)



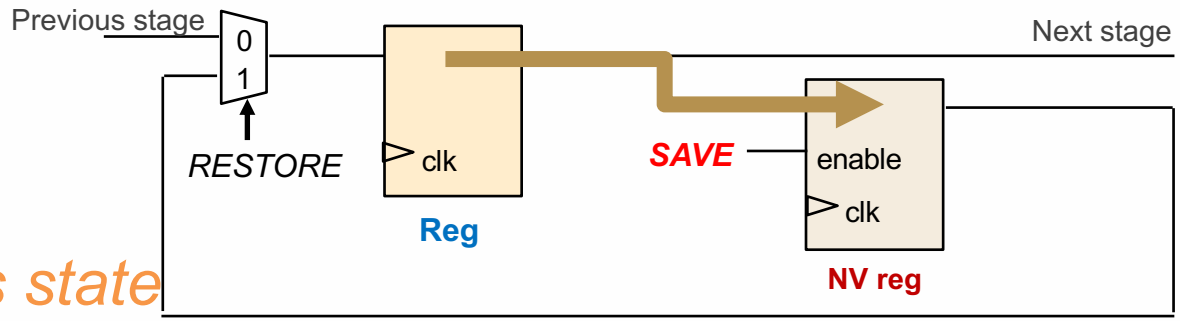
FEATURES

- 3-stage pipeline
- 16x32-bit register file
- 32-bit wishbone system bus
- Unified instruction/data cache (16 kBytes)
  - Write through
  - Read-miss replacement policy
- Main memory (> Mbytes)
- Multiply and multiply-accumulate operations

- Implementation of both instant-on/off and rollback (Verilog code modified)
- Duplication of the registers to emulate the non-volatility

Instant on/off

Instant on/off



1. *Save the register's state*

2. *POWER DOWN*

Main memory based on MRAM

↓

Data preserved

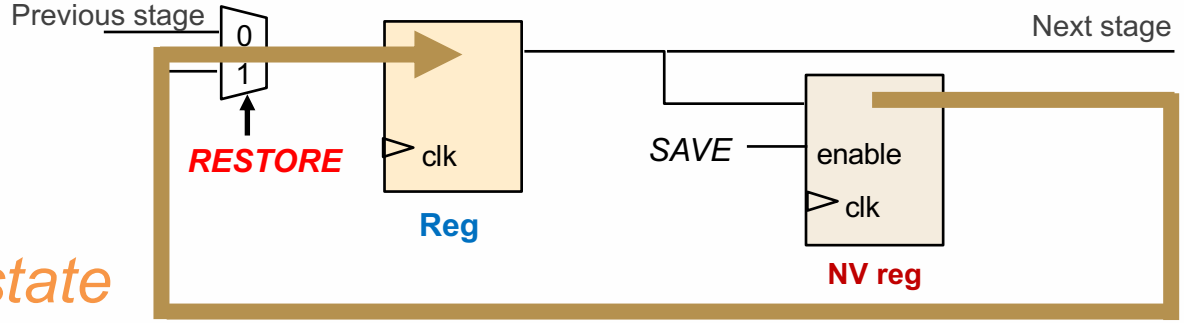
3. *POWER UP*

Main memory based on MRAM

↓

Data available

4. *Restore the register's state*



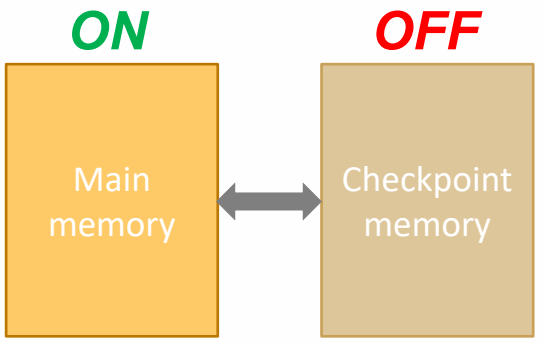
31

Rollback

Rollback

NORMAL EXECUTION

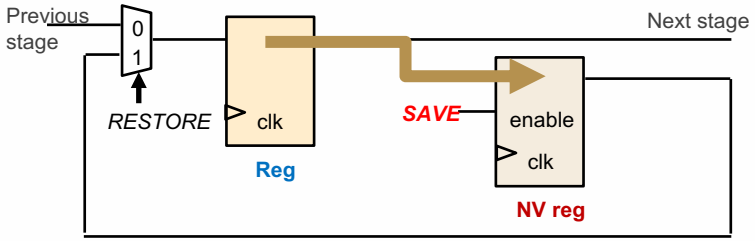
- ***Only the main memory contents are modified***
- ***The checkpoint memory is powered off***



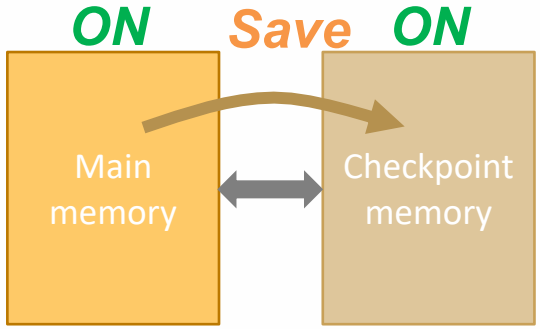
CHECKPOINT

- ***Save registers***
- ***Save memory***

=

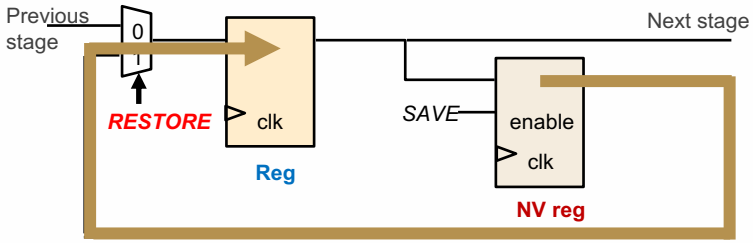


+

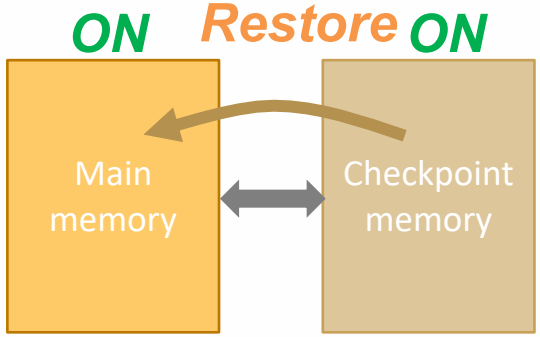


ROLLBACK

1. ***Stall the processor***
2. ***Restore checkpoint***
3. ***Execution***



+



SoC based on MSS

The goal of GREAT project is to co-integrate multiple functions like sensors, RF receivers and logic/memory together within CMOS by adapting STT-MTJs to a single baseline technology in the same System on Chip as the enabling platform for M2M and M2H IoT.

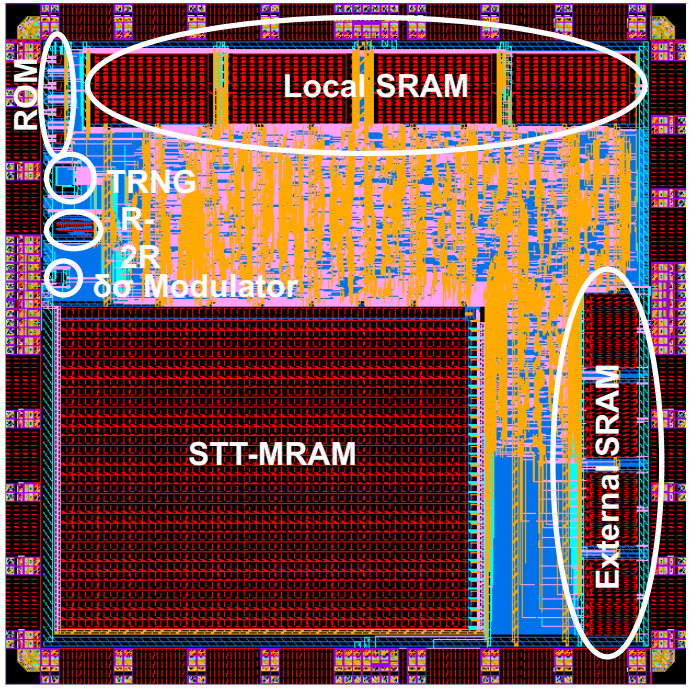
Proof of concept

- A full SoC based on STT-MTJ under fabrication

Full Layout

Process
180nm CMOS (TowerJazz)
200nm STT-MTJ (Spintec,Singulus)

Die area
~23mm²



Power supply
1.8V Core / 3.3V IO

Frequency
20 MHz

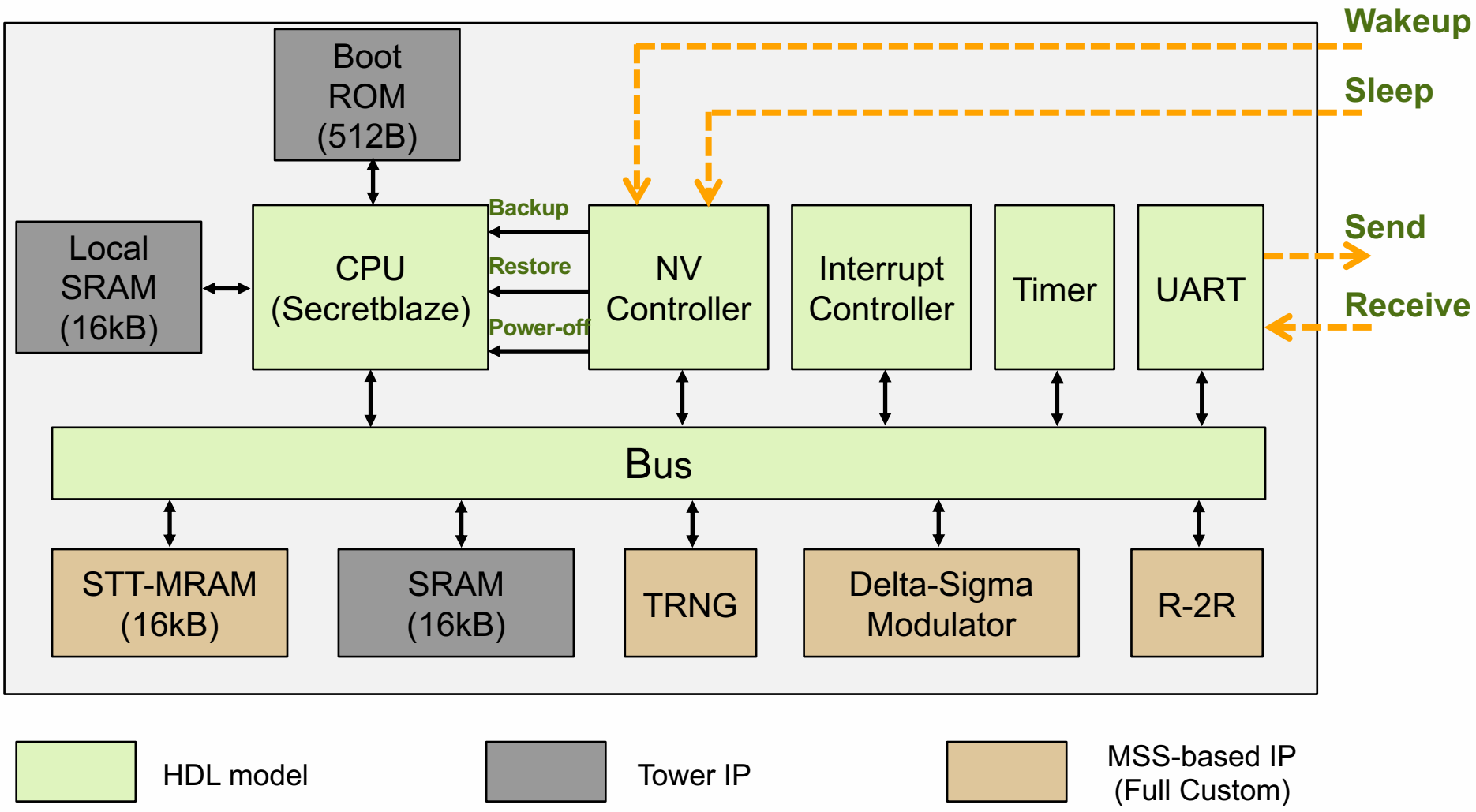
2126 NVFFs

GREAT

LIRMM

spintec
SPIN IN ELECTRONICS

Overall architecture



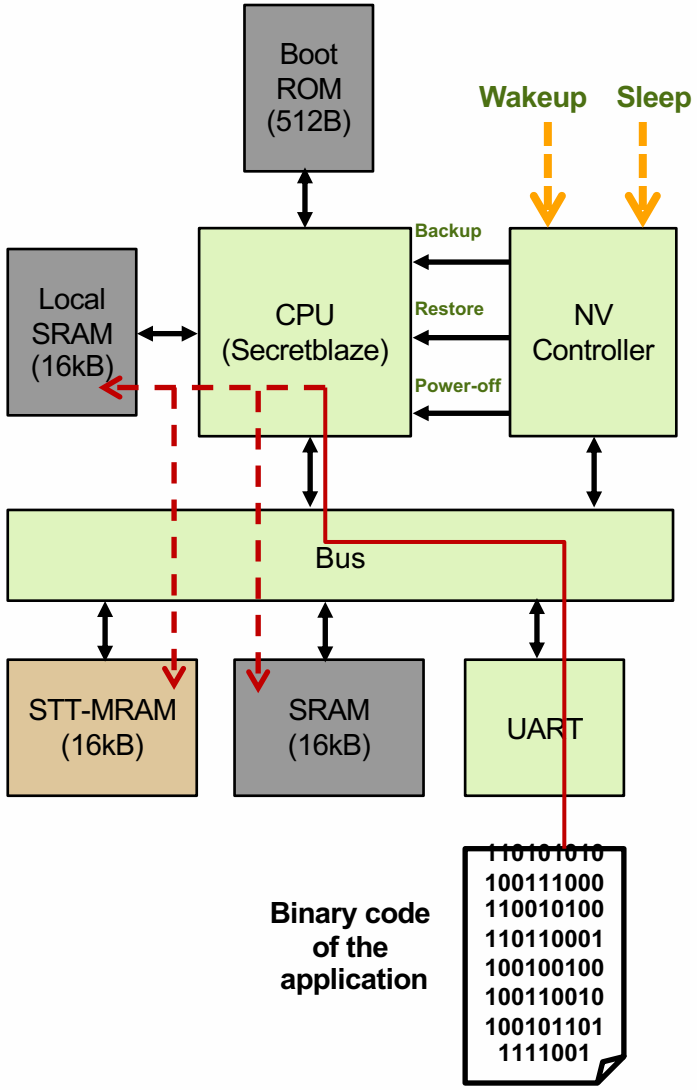
Operation

Normal execution

- Binary code loaded via UART
- Program executed from either local SRAM, or external SRAM or STT-MRAM

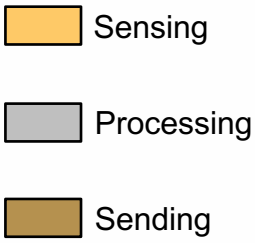
Active/Sleep modes management

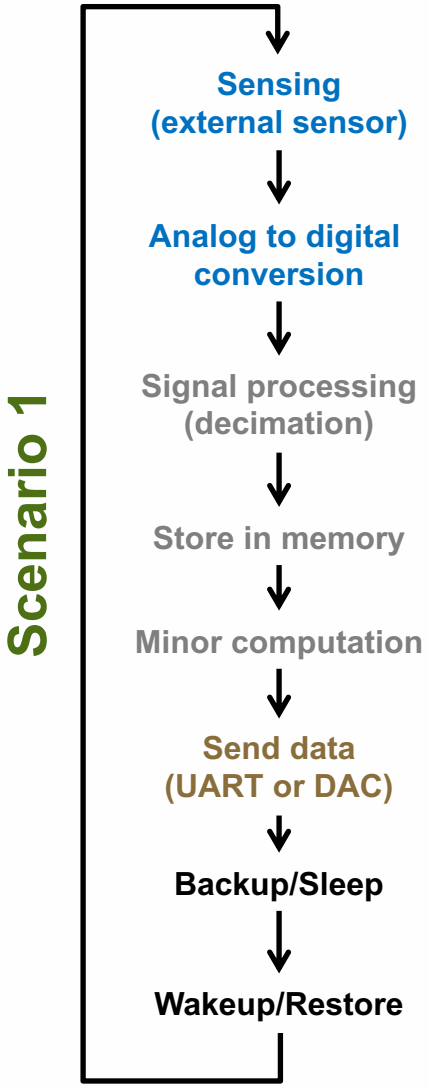
- Specific Controller
- Backup is initiated by software
- Power-off is triggered:
  - Either by an external signal « sleep »
  - Or by software
- Recovery is triggered:
  - Either by an external signal « wakeup »
  - Or futher to an event from the interrupt controller

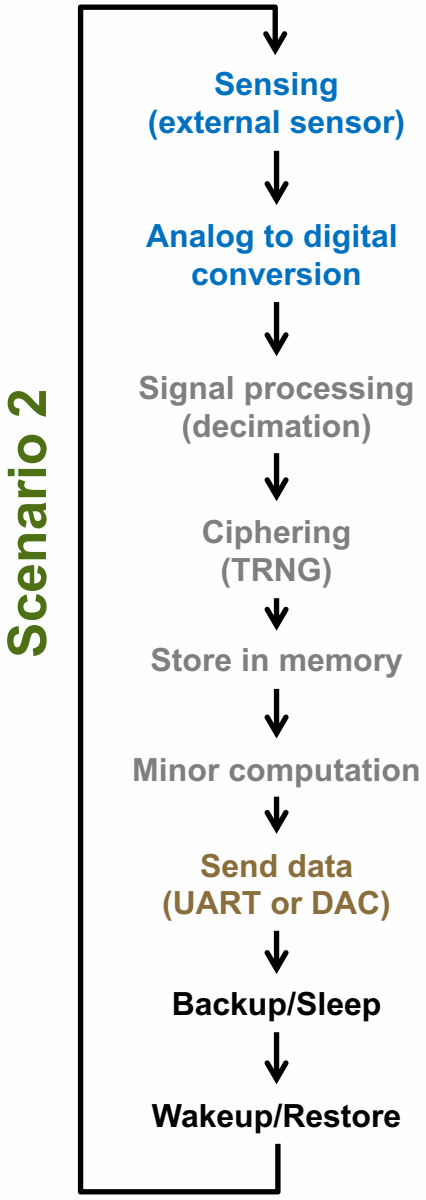


SoC based on MSS

Application scenarios



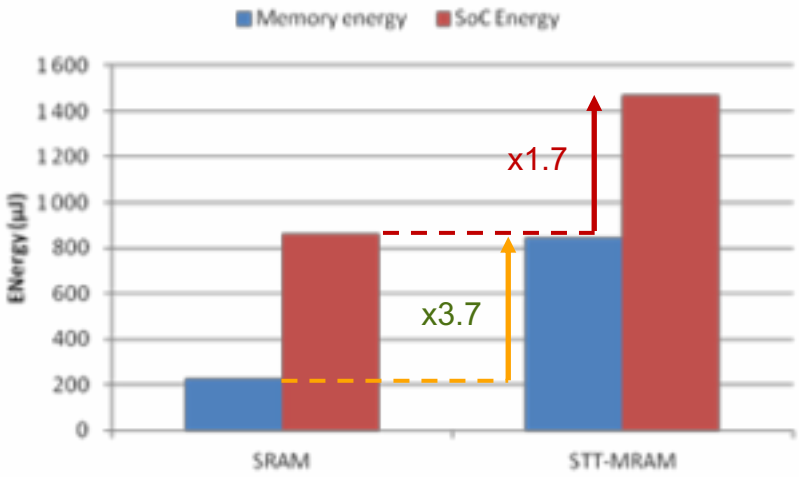




Active Energy @20MHz

Scenario 1

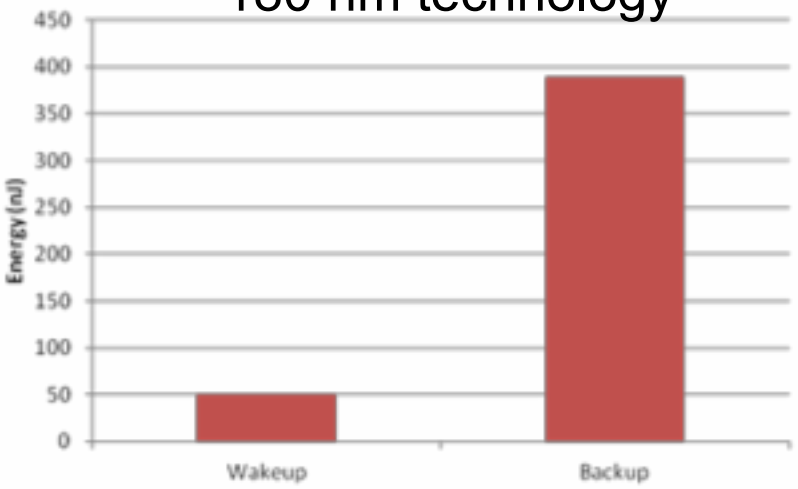
Scenario 2



Comparison between execution from SRAM and execution from STT-MRAM
(Post-layout simulations)

Backup/Wakeup Energy

180 nm technology



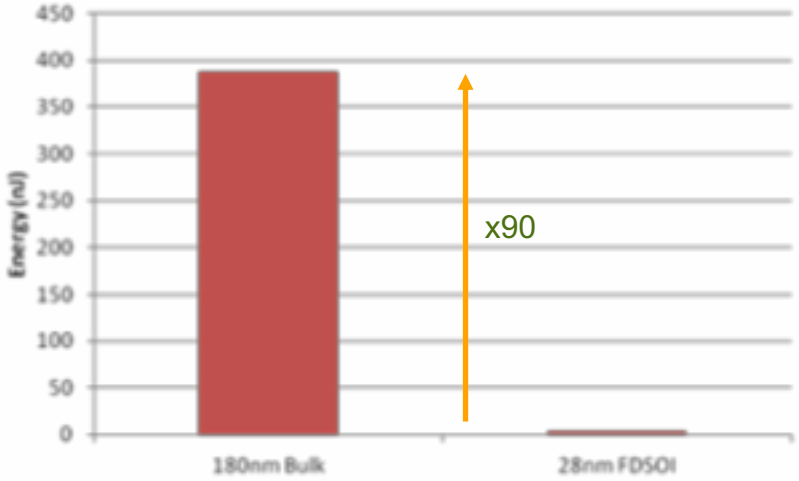
Wakeup time : 4.15 µs

Backup time : 4.15 µs

2126 NVFFs arranged by clusters to avoid electrical integrity issues

- 82 clusters

28nm projection



Number of clusters

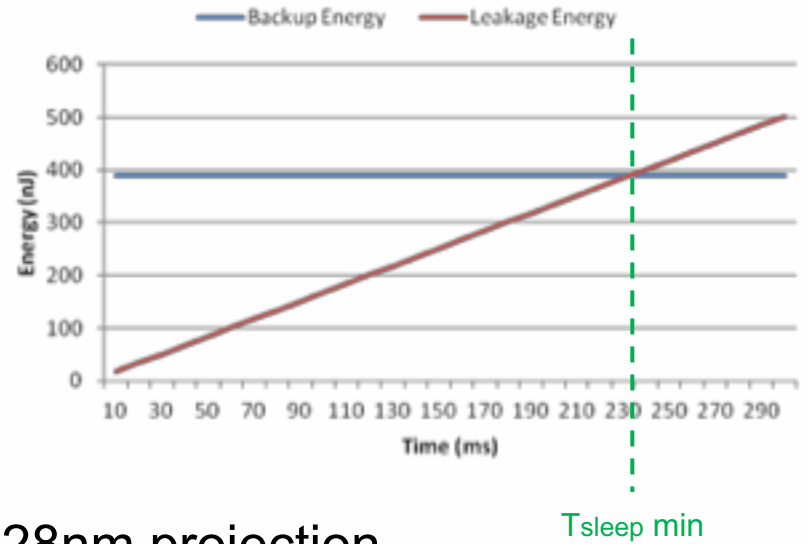
- 82 (for 180nm)
- 2 (for 28nm)

Backup time @20MHz

- 4.1µs (for 180nm)
- 100ns (for 28nm)

Minimum Tsleep

180 nm technology

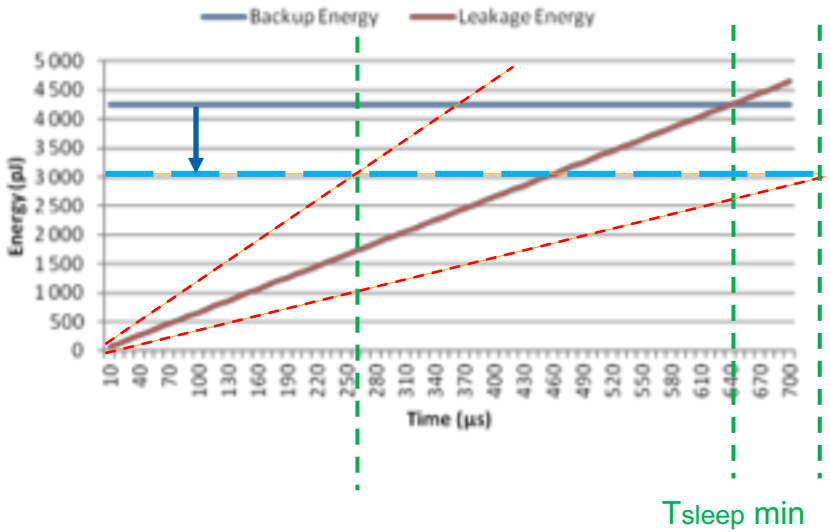


Backup energy is independent of the time spent in sleep mode

Leakage energy is dominated by SRAM

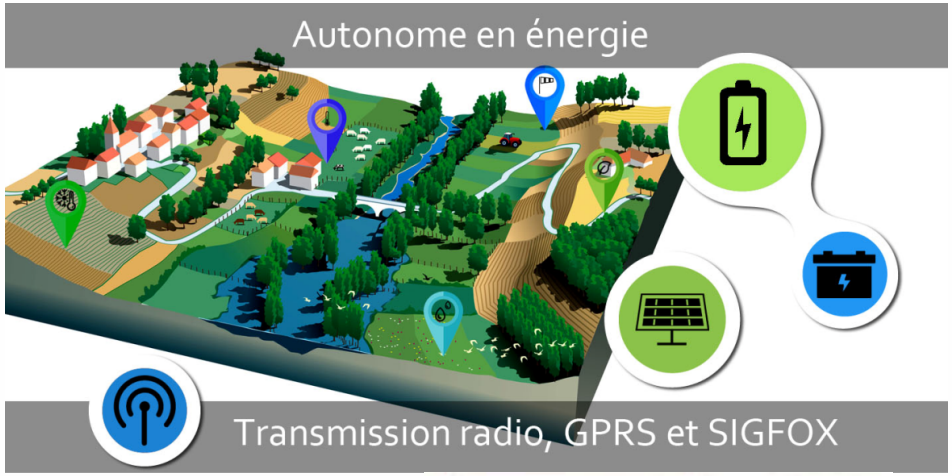
Minimum T_{sleep} to compensate the backup energy ≈ 65 ms

28nm projection



Minimum T_{sleep} to compensate the backup energy ≈ 641µs

Sensor node application (Agriscope)



- Agro monitoring : plant disease, temperature, irrigation, pesticide threshold, so on ..

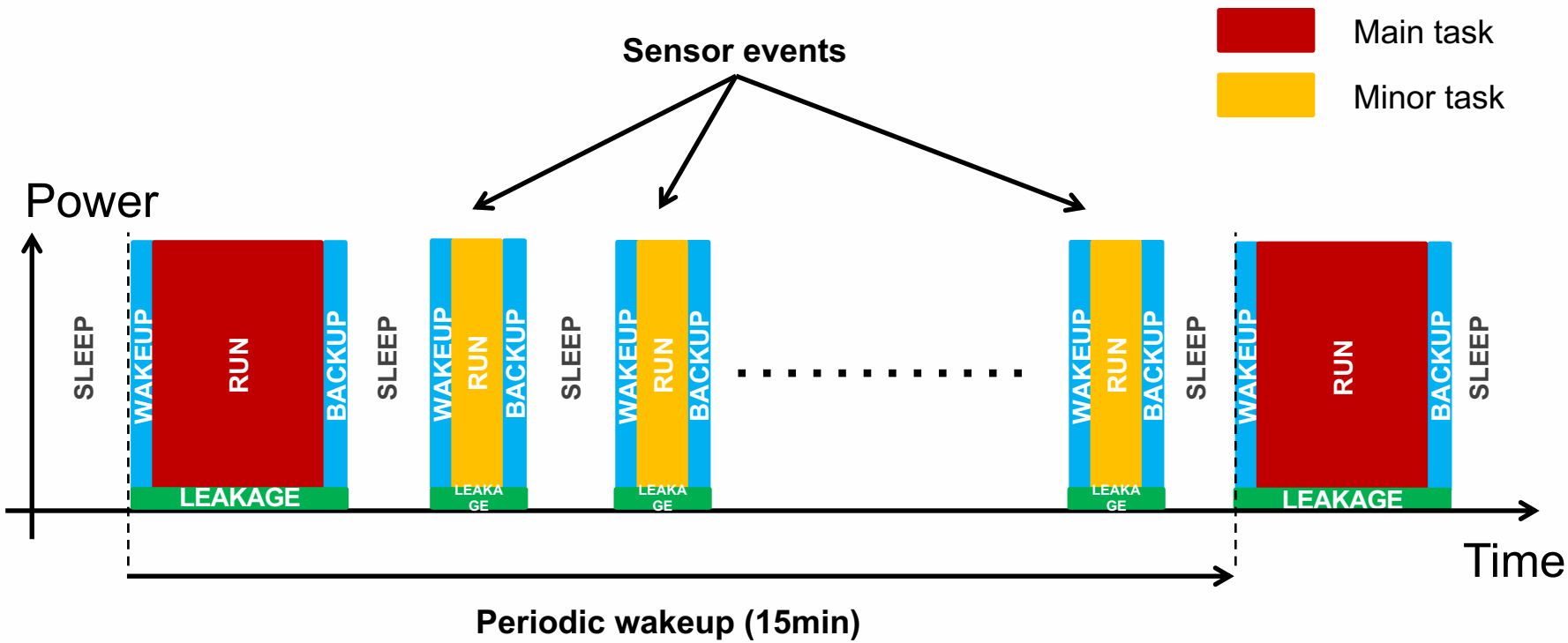
-Solutions are based on 32-bit processor + RF stack

- Targeted autonomy 10 years

- Applications fully compatible with our SoC

- **Agriscope's application has been ported to our SoC to compare with industrial use case**

Application to a sensor node



Energy consumption in the case of a non-volatile MCU - No leakage in sleep mode - Backup energy

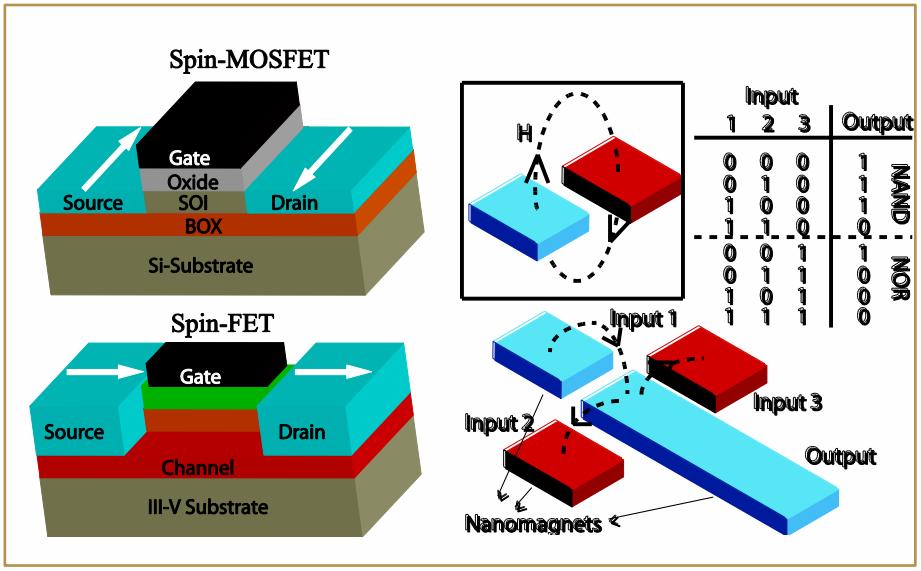
Sensor events	Traditional MCU	NV MCU	Factor
0	6.03 mJ	9.09 µJ	X 663
15 (1 per min) (rain gauge)	6.08 mJ	121 µJ	X 50
180 (1 per sec) (water meter)	6.81 mJ	1.5 mJ	X 4.5
9000 (1 per 0,1s sec) (anemometer)	41.4 mJ	67.2 mJ	/ 1.6

Conclusion

MRAM has a high potential to:

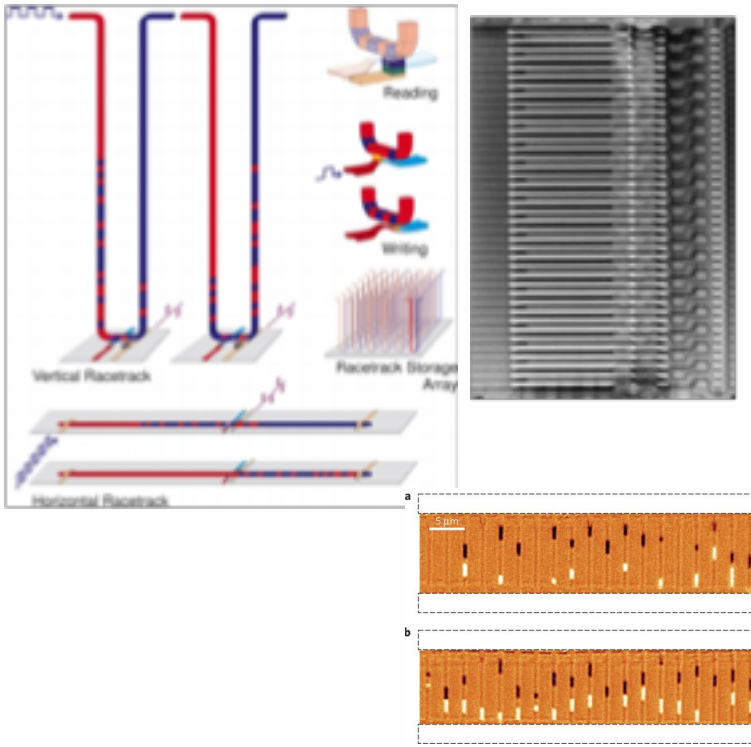
- **Certainly Reduce energy consumption**
  - At cache level (sure and proven)
  - Normally-off computing
- **Can facilitate some features**
  - Normally-off computing / Instant on-off
  - Backward error recovery (Rollback)
- **Results should be confirmed through measurements on silicon prototype !**
- **Important : Link with compilation and OS**
- **An open framework available to the community : MAGPIE**

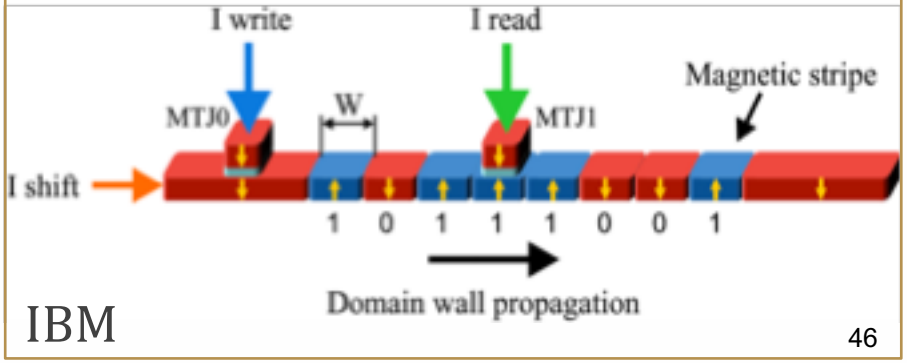
For the future….



THE ALL

Racetrack memory





THANKS !