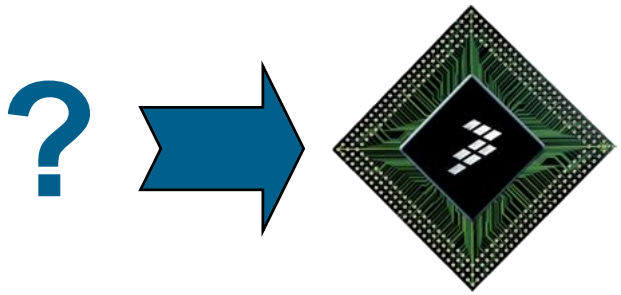
Embedding Functional Safety –
by Hardware or by Software?

Markus Baumeister

Sep. 2009

freescale™ semiconductor

The problem of functional safety



freescale™ semiconductor

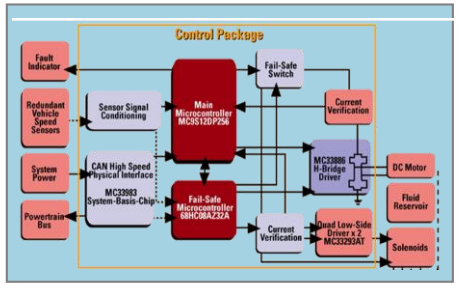
Automotive Safety and Functional Safety

Functional Safety Systems

Automotive Safety Systems

Active Safety Systems

Passive Safety Systems



"Safety of persons in respect to accidents"

"Safety of persons in respect to (E/E/PES) system failures"

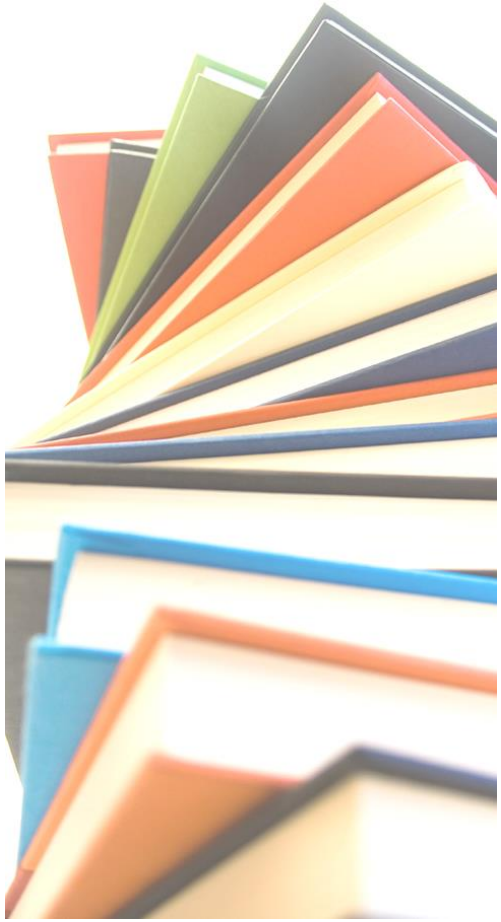
Safety is freedom from unacceptable risk"

(IEC 61508)

freescale™ semiconductor

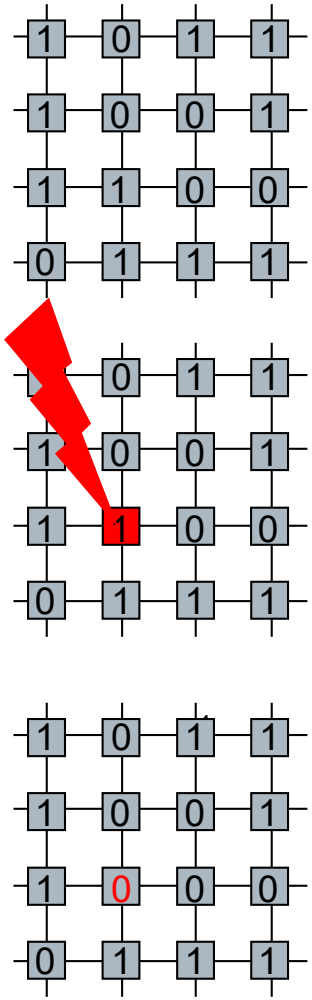
Safety Standards

- Standards establish metrics and development process recommendations
- IEC 61508
  - General industry standard for functional safety in electrical and electronic systems
  - V1 since late 1990s, V2 announced
- ISO 26262
  - Adaptation of IEC 61508 to comply with needs specific to the application sector of E/E systems within road vehicles
  - Current draft, release expected ~2010



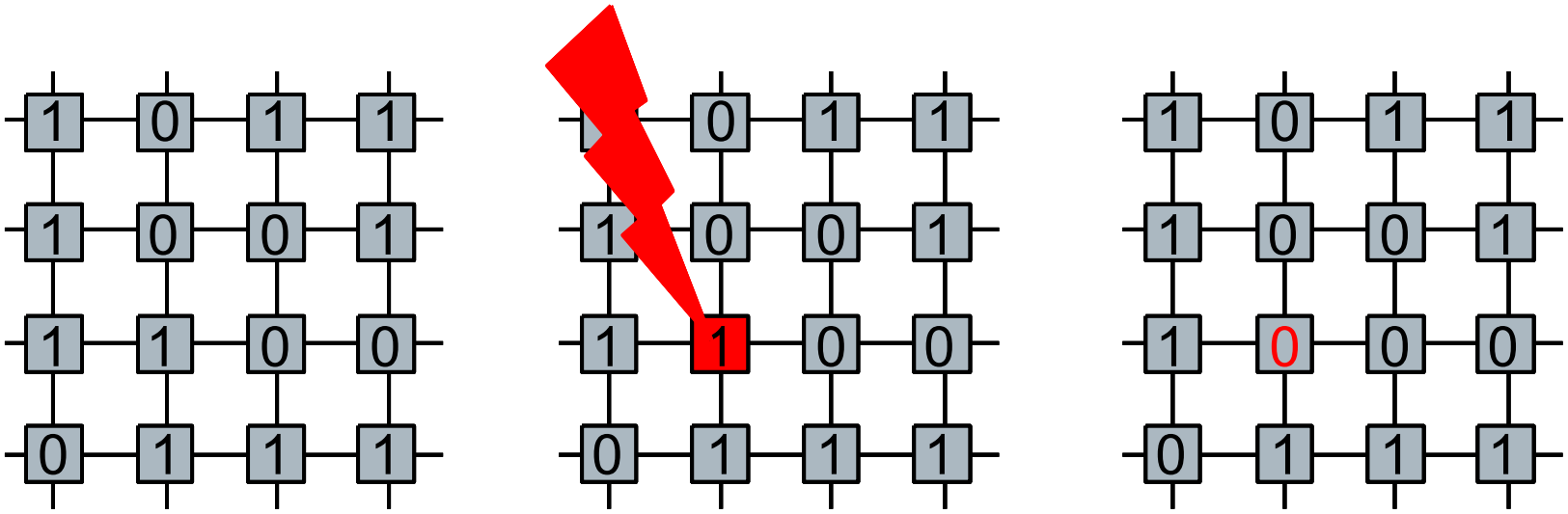
Contents

- Example: Memory ECC for functional safety
  - Additional failure modes
  - Probabilities
  - Countermeasures
- Cost for safety
  - Hardware
  - Software
- Other SW vs. HW examples
- Discussion



freescale™ semiconductor

Primary error: Bit flip in System RAM



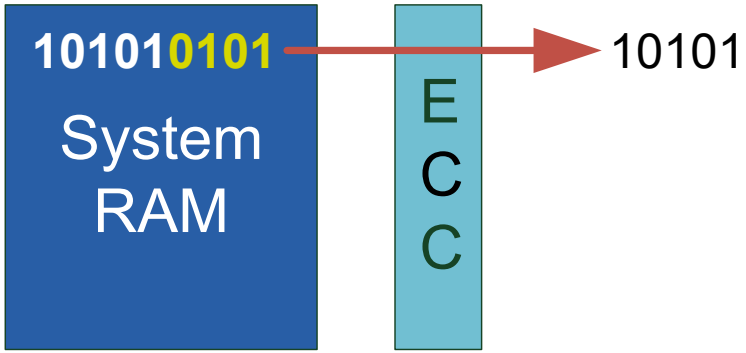
- Transient (soft) fault
  - Alpha particles
  - Neutrons
- Permanent (hard) fault
  - Stuck-at
  - Bridging
  - Delay

In current technology transient faults are more common

freescale semiconductor

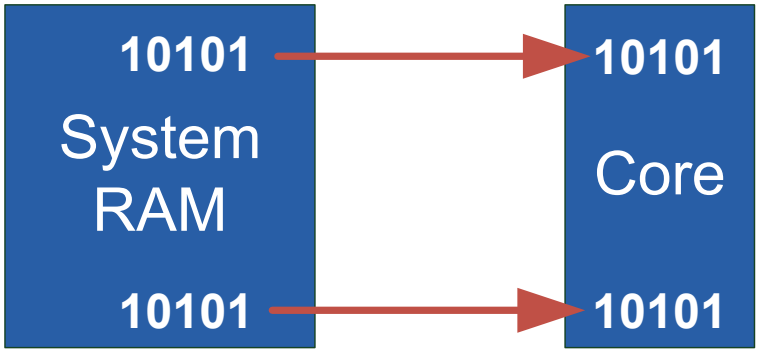
Countermeasures

- Hardware (ECC)
  - Store additional parity bits
  - 8 parity bits for 64 data bits
  - Additional logic to encode/decode



- Alternatives
  - “Hardened” RAM
  - Parity

- Software (Replicated Storage)
  - Store twice in RAM
  - Write twice
  - Read twice and compare
  - Operations by MCU core

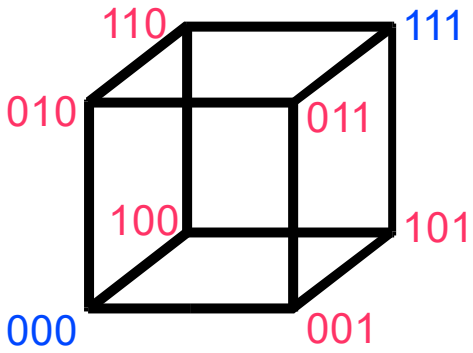


- Alternatives
  - Checksum (Application CRC)

freescale semiconductor

Typical **E**rror **C**orrecting **C**ode:

- Single Error Correction/Double Error Detection
- Hamming Code with Hamming Distance 4
  - combination of multiple parity codes
- Notation: (#overall bits,#data bits) Code



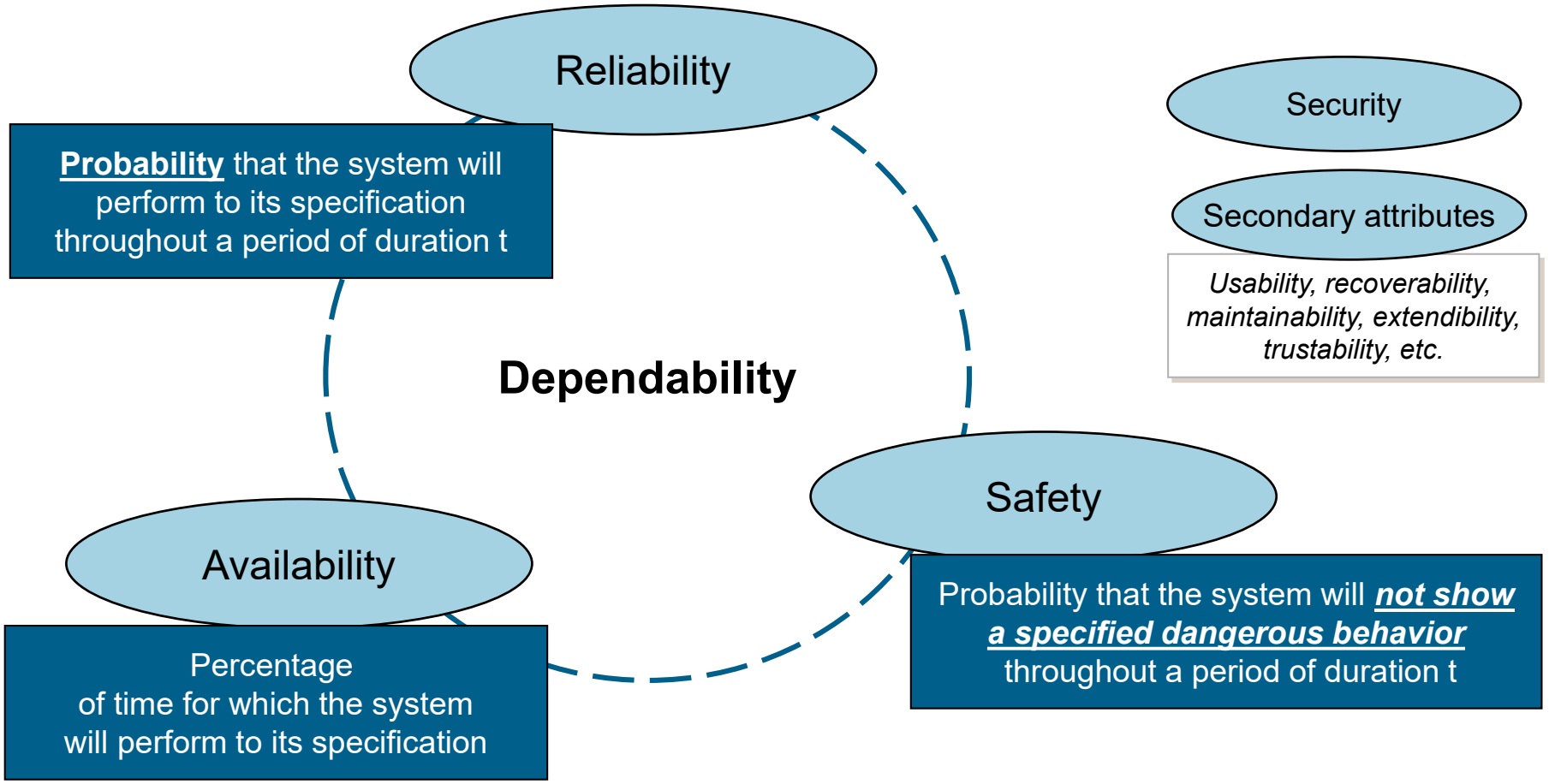
(3,1) Hamming Code, Hamming distance 3

Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1		X		X		X		X		X		X		X		X
2			X	X			X	X			X	X			X	X
4					X	X	X	X					X	X	X	X
8									X	X	X	X	X	X	X	X

(16,11) Hamming code, distance 4

- Has unused bits: E.g. (72,64)-code, potentially protects 120 data bits, actually protects 64

Safety in the Context of Dependability (IFIP WG10.4)



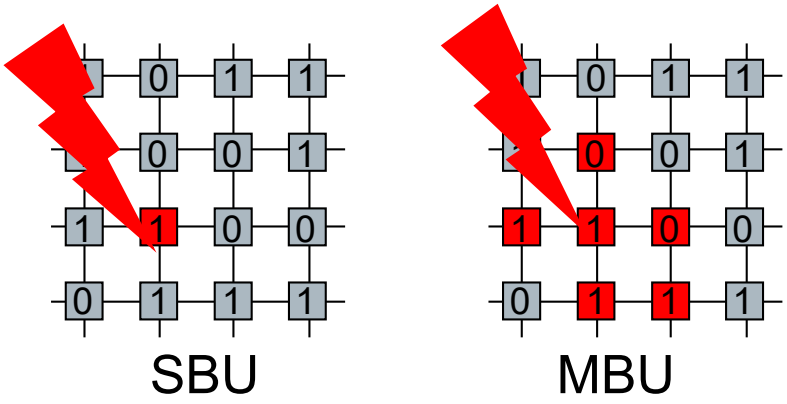
freescale semiconductor

Contents

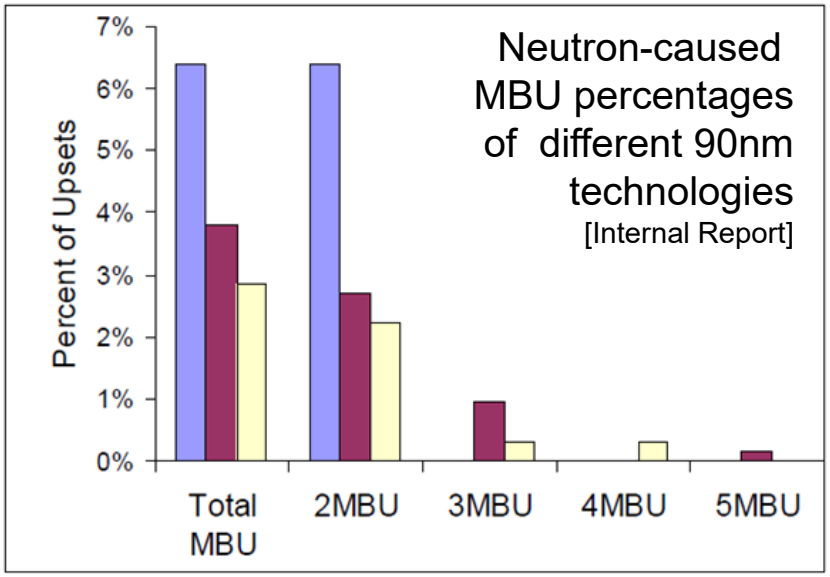
- Example: ECC for functional safety
  - Additional failure modes
  - Probabilities
  - Countermeasures
- Cost for safety
  - Hardware
  - Software
- Other SW vs. HW examples

Additional failure mode: MBUs

- Multi Bit Upsets flip more than one bit



… and are not that seldom

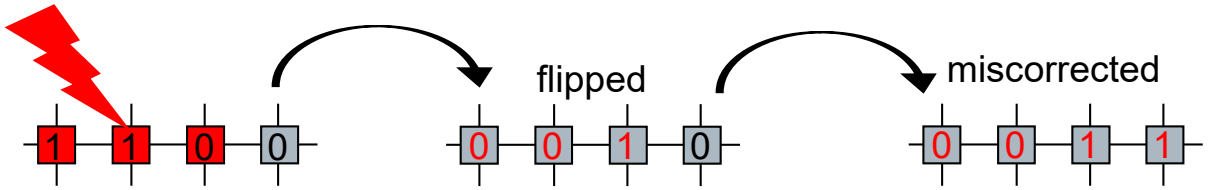


freescale™ semiconductor

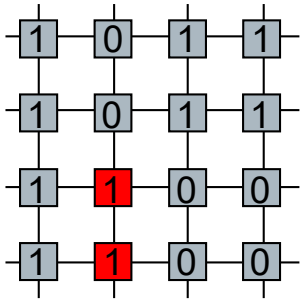
Effects on primary countermeasures

- HW
  - SECDED can miscorrect if 3 or more bits are flipped
  - E.g. (72,64) Hamming code, 3 different flipped bits

  ⇨ ≥ 56% Chance for miscorrection



- SW
  - Replicated storage
    - depends on physical placement of data
  - Application CRC
    - Resistant to bursts

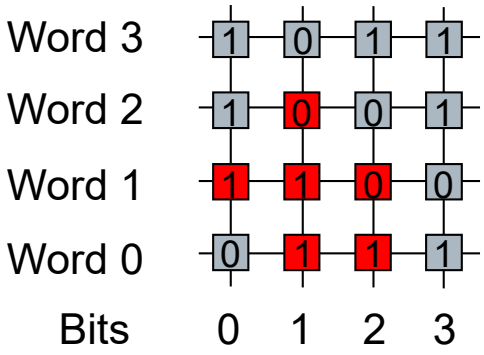


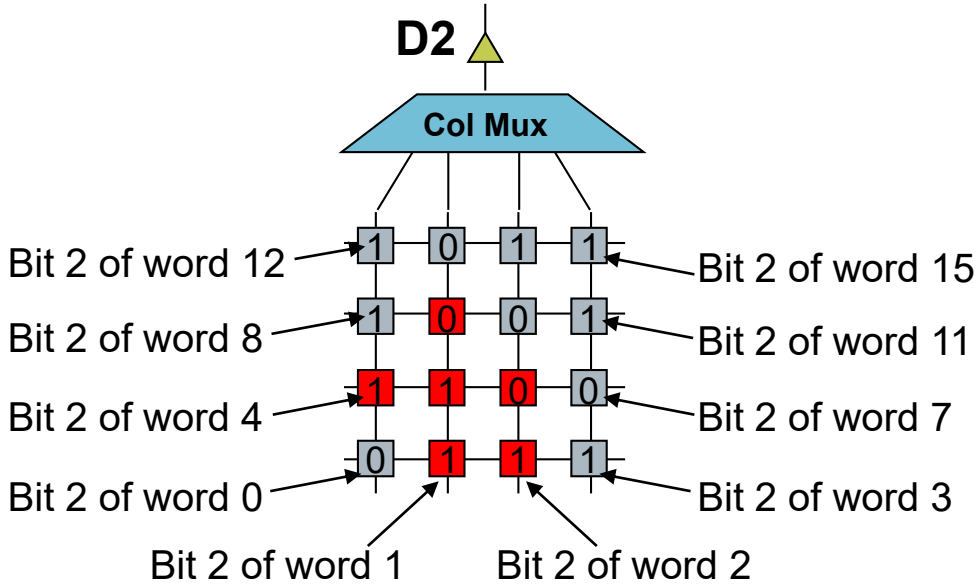
MBU and
neighbouring duplicates

freescale™ semiconductor

Additional Countermeasures against MBUs

- Hardware countermeasures
  - Column multiplexing

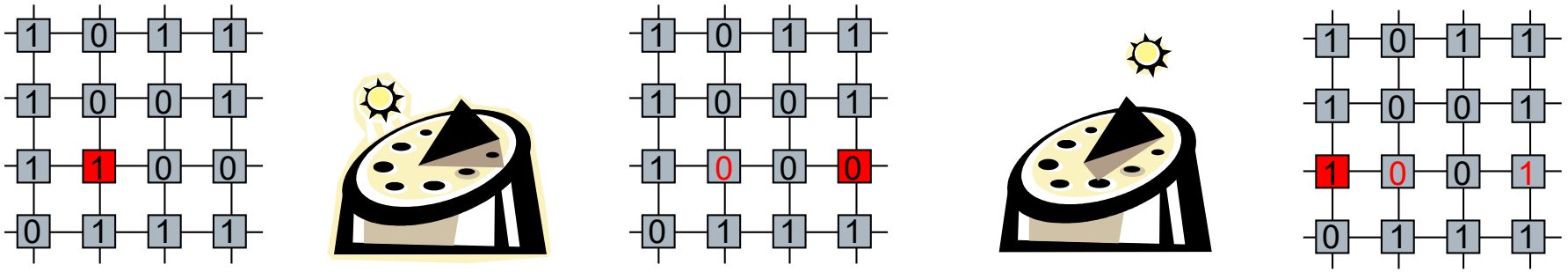




freescale™ semiconductor

Additional failure mode: Fault accumulation

- Over time, several bits can flip

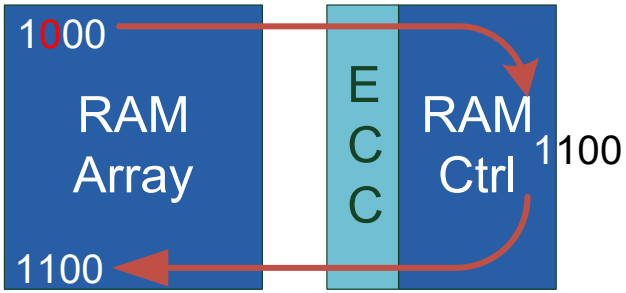


- Same effect as MBU

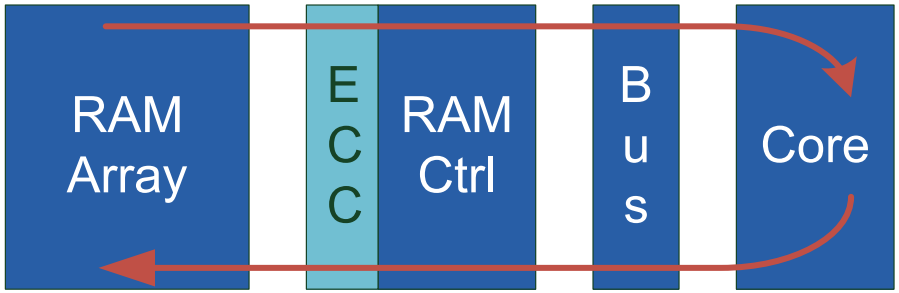
freescale™ semiconductor

Additional Countermeasures against fault accumulation

- Countermeasure: “Scrubbing”
  - Read-Correct-Write back



HW-Scrubbing



SW-Scrubbing

- Alternative: Checking
  - Read and Fail

freescale™ semiconductor

Probability of fault accumulation

HW – ECC:

- At least 3 independent faults
- MBUs affect several words
- Example:
  - Transient fault rate 2000 FIT/MBit
  - Max. operation time: 10 hours
  - (72,64) ECC
  - 65536 words (=512K) RAM

SW – Repl. Storage:

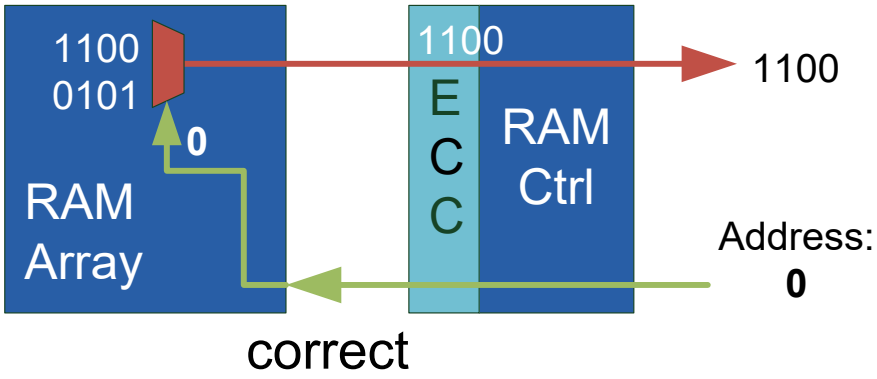
- Fault at bit and replicated bit
- MBUs ignored

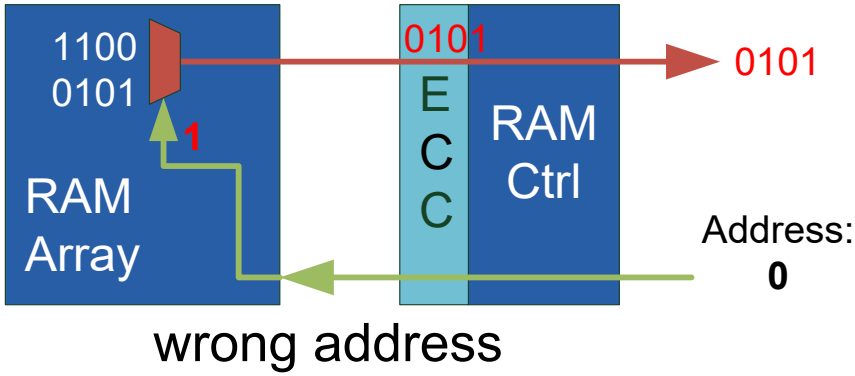
HW – ECC: Approx. probability of 3 or more flipped bits in any one word	IEC 61508: Typically allowed overall failure rate for SIL 3 MCU	SW – Repl. Storage: Approx. probability of any bit & its corresponding replicated bit to flip
Ca. $2.3*10^{-23}$	$1*10^{-10}$/h	Ca. $8.8*10^{-16}$

- Periodic reset is an effective countermeasure for safety

Additional failure mode: Addressing error

- Failure: Wrong memory row accessed



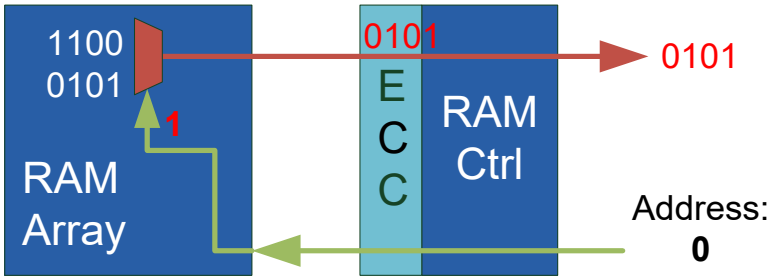


- ECC taken from same row ⇨ correct
- Failure variants possible

Probability of addressing errors

HW probability:

- Remaining ECC diagnostic coverage: Est. 35%
- Failure rate in RAM array: Ca. $2*10^{-11}/h * 0.65 = 1.3*10^{-11}/h$
- Failure rate in RAM Ctrl: Ca. $1.5*10^{-10} * 0.65 = 1.0*10^{-10}/h$



SW probability:

Address Replica 1	Address Replica 2
10000010	11000010
10000001	11000001
…	…

Remaining diagnostic coverage wrong address: Est. 12.5%
Overall est. 38%
but ca. 99% against transient faults

Address Replica 1	Address Replica 2
10000000	11001010
10000001	11011010
…	…

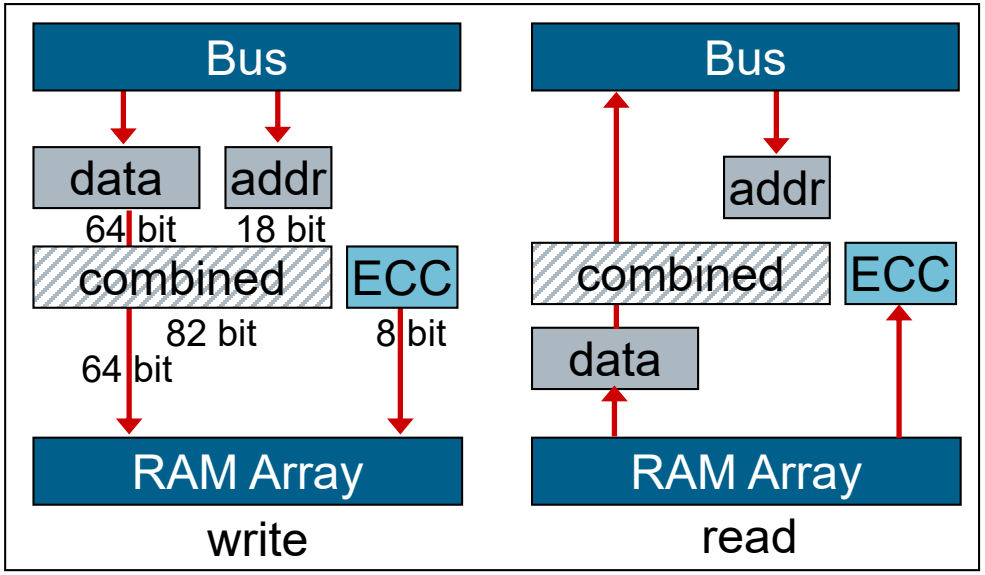
Remaining diagnostic coverage wrong address: > 90%
Overall > 90%

Values are not generally applicable

freescale™ semiconductor

Addressing errors & Countermeasures

- HW countermeasures
  - Address in ECC calculation



    - Ca. 60% diagnostic coverage
  - Local Checks
    - E.g. Address feedback
    - Read after write

- SW countermeasures
  - Known reads/writes on specifically selected addresses
  - Read after write

Values are not generally applicable

Contents

- Example: ECC for functional safety
  - Additional failure modes
  - Probabilities
  - Countermeasures
- Cost for safety
  - Hardware
  - Software
- Other SW vs. HW examples

Effectiveness of safety measures

Measure	Single Bit flip	MBU	Fault accumu-lation	Logic error	Addressing error
SW: Replicated storage	Detectable	No effect if storage positions chosen well	Negligible probability	N/A	High diag. coverage if storage positions scrambled
HW: ECC	Correctable	No effect if column multiplexing	Very negligible probability	Additional countermea sure might be needed	Additional countermea sures are needed

- So why use ECC?

Cost of safety measures

Measure	HW effort	HW performance	SW effort	Availability
SW: Replicated storage	2 * RAM for safety-relevant data	2 * storage access for safety-relevant reads/writes	Double reads/writes & compares in safety-relevant program	Not improved
HW: ECC	1.1 * RAM for all data	Slightly prolonged critical path for all writes	Possibly some separated SW safety measures	1 correctable error per RAM word

- ECC dominates non-safety criteria

Influence factors for HW vs. SW safety measures

HW	SW
Diagnostic coverage	
HW-, SW-, Performance cost	

freescale™ semiconductor

Influence factors for HW vs. SW safety measures

HW	SW
Diagnostic coverage	
+ Non-filtered HW access (RAM test)	– HW failure mode analysis + Handling of non-standard/ seldom errors (Scrubbing)
HW-, SW-, Performance cost	
+ Local actions without CPU involvement (Scrubbing) + Exploitation of synergies Column Muxing) + Solution developed once	+ Use of non-safe hardware + Application tuned measures

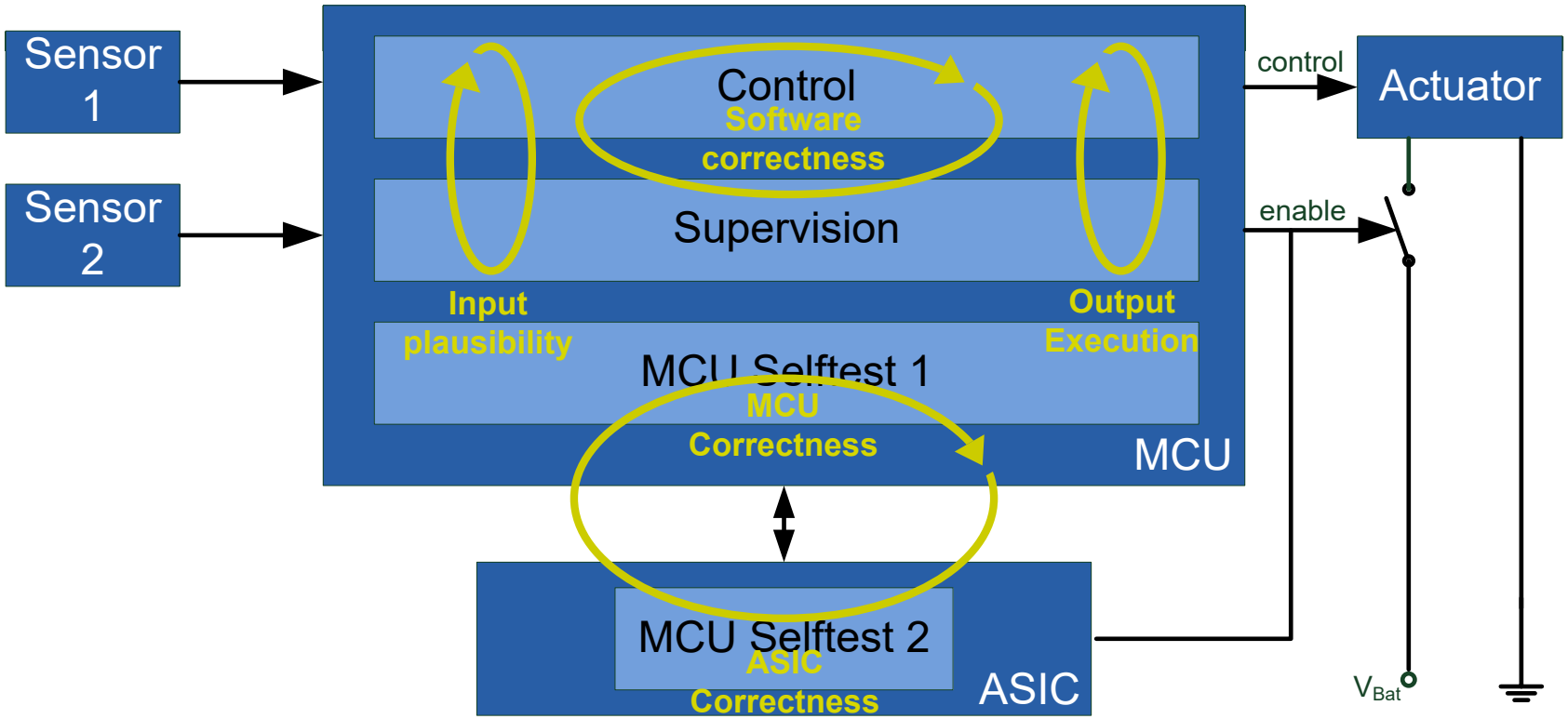
Influence factors for HW vs. SW safety measures

HW	SW
Diagnostic coverage	
+ Non-filtered HW access	– HW failure mode analysis + Handling of non-standard/ seldom errors
HW-, SW-, Performance cost	
+ Local actions without CPU involvement + Exploitation of synergies + Solution developed once	+ Use of non-safe hardware + Application tuned measures

Generic vs. application specific

How far can HW measures go? (1)

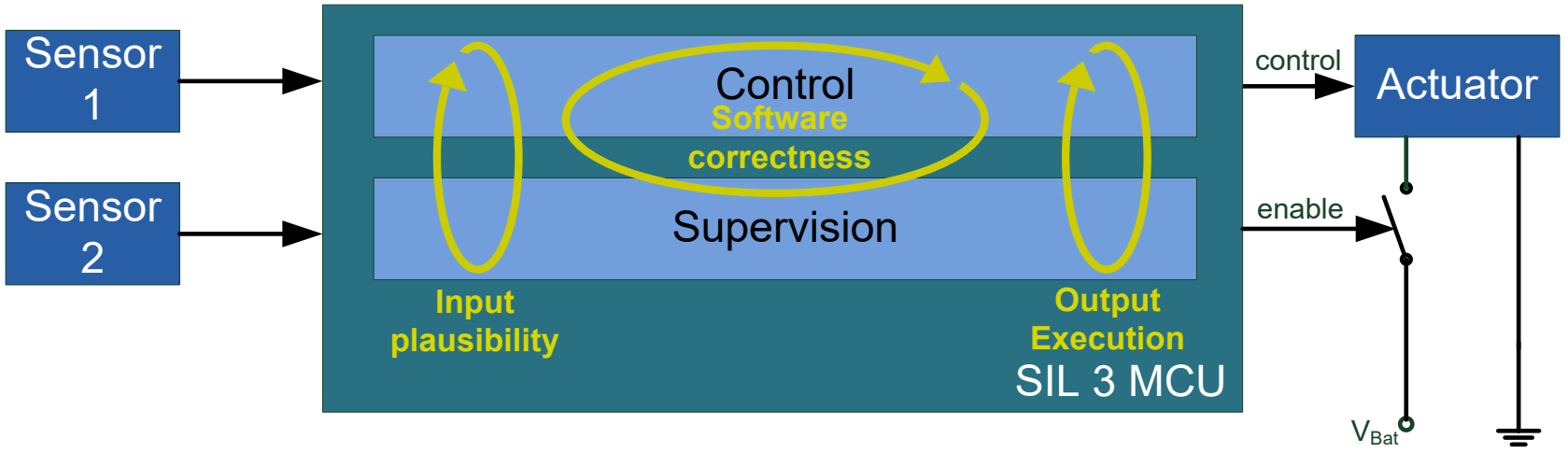
- Example: Electronic throttle control



freescale™ semiconductor

How far can HW measures go? (2)

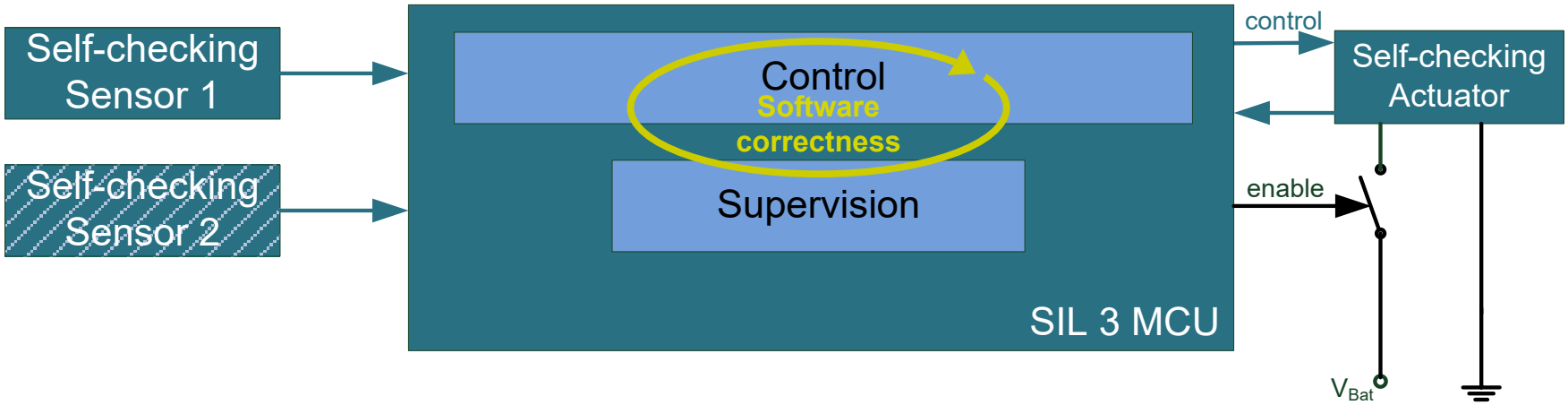
- SIL 3 capable MCU instead of self tests



freescale™ semiconductor

How far can HW measures go? (3)

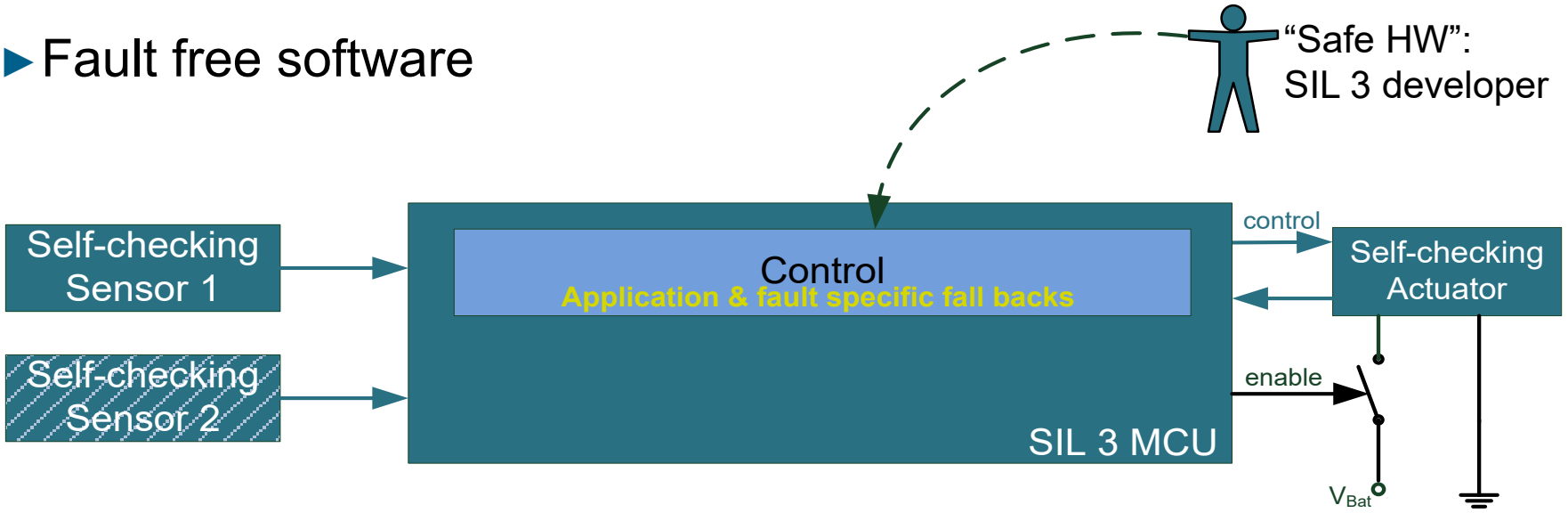
- Self-checking periphery and safe communication



freescale semiconductor

How far can HW measures go? (4)

- Fault free software



⇨ HW and SW safety measures will probably co-exist

Contents

- Example: ECC for functional safety
  - Additional failure modes
  - Probabilities
  - Countermeasures
- Cost for safety
  - Hardware
  - Software
- Other SW vs. HW examples

HW-SW balance for safety measures

More HW-oriented:

- Control flow watchdog
- HW self-test
- Lockstep MCU
- Diverse cores (forcing somewhat diverse software)
- Temperature throttling

More SW-oriented:

- SW flow control with simple watchdog
- SW self-test using minor HW extensions
- Decoupled dual core MCU with SW synchronization & comparison
- n-Version programming
- SW readable temperature sensor

Summary

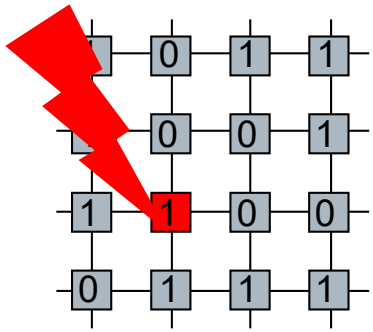
- Evaluation of software measures against hardware failures requires detailed hardware failure knowledge
- Important decision criteria
  - Diagnostic coverage
  - Costs
  - Generic vs. application-specific solution
- Costs difficult to quantify
  - Development vs. recurrent costs
  - System complexity
- Coexistence/Cooperation of hardware and software measures

⇨ Semiconductor vendors need customer feedback


freescale™
semiconductor

Countermeasures

- Software
  - Replicated Storage
  - Checksum calculation (e.g. Application CRC)
- Hardware
  - “Hardening” of RAM
  - Checksum (e.g. Parity)
  - Error correcting codes (e.g. Hamming)



Diagnostic technique/measure	See IEC 61508-7	Maximum diagnostic coverage considered achievable
RAM test "checkerboard" or "march"	A.5.1	Low
…	…	…
Parity-bit for RAM	A.5.5	Low
RAM monitoring with a modified Hamming code, or detection of data failures with error-detection-correction codes (EDC)	A.5.6	High

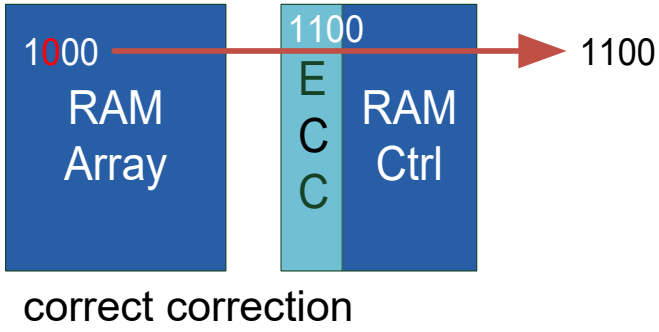
IEC 61508-2 table A.6: Variable Memory

Low = 60% diagnostic coverage
High = 99% diagnostic coverage
Diagnostic coverage: Percentage of dangerous failures detected

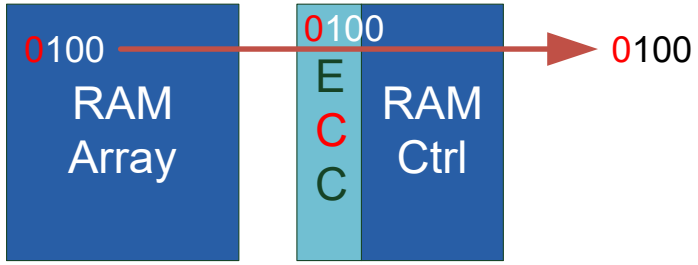
freescale™ semiconductor

Additional failure mode: ECC logic error

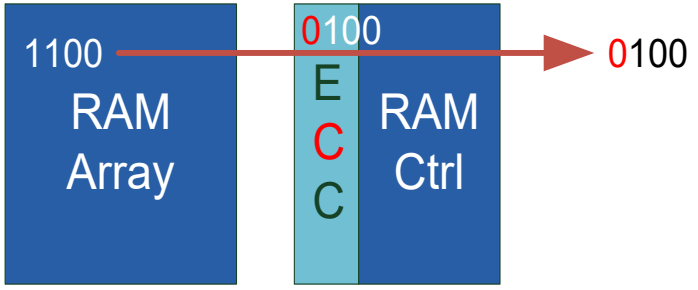
- ECC logic can change the data



correct correction



no correction



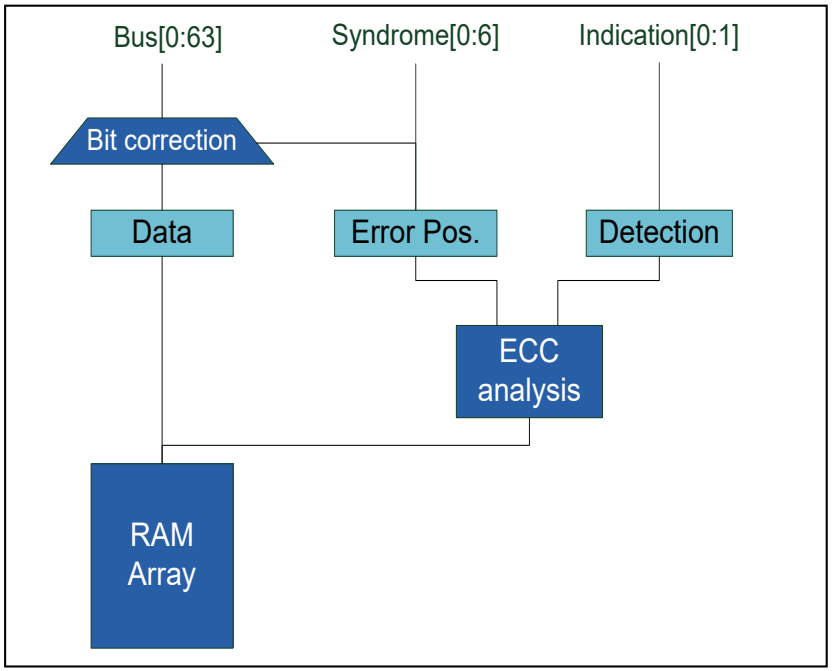
corruption

freescale™ semiconductor

Probability of ECC logic error

- Approach A: Rough estimation
  - Overall permanent failure rate of MCU[1/h]: $2*10^{-9}$
  - Fraction of MCU logic dedicated to ECC: ca. 0.04%
  - Approx. permanent failure rate [1/h]: $2*10^{-9}*0.0004 = 8*10^{-13}$

- Approach B: FMEDA
  - Register bits: 71
  - Gates: ca. 1000
  - Transient dangerous error rate [1/h]: $<0.12 *10^{-9}$
  - Permanent dangerous error rate [1/h]: $<5*10^{-13}$



Values are not generally applicable

freescale™ semiconductor

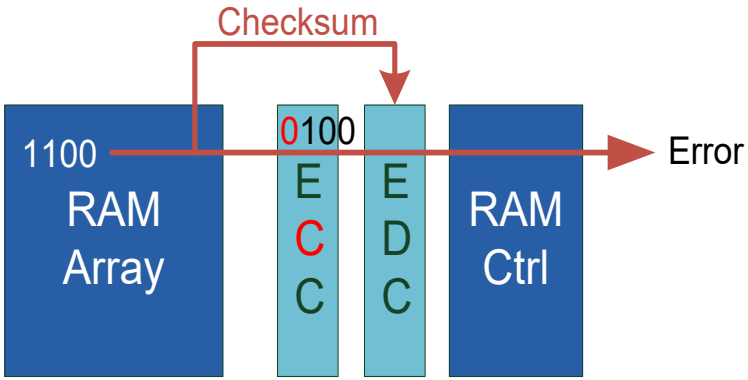
Additional countermeasures against logic errors

HW-measures

- Only no correction
  - Startup selftest
- Both
  - Replication
  - Checker

SW-measures

- Only corruption
  - Correction indicator
  - Inherent detection
- Both
  - Known reads



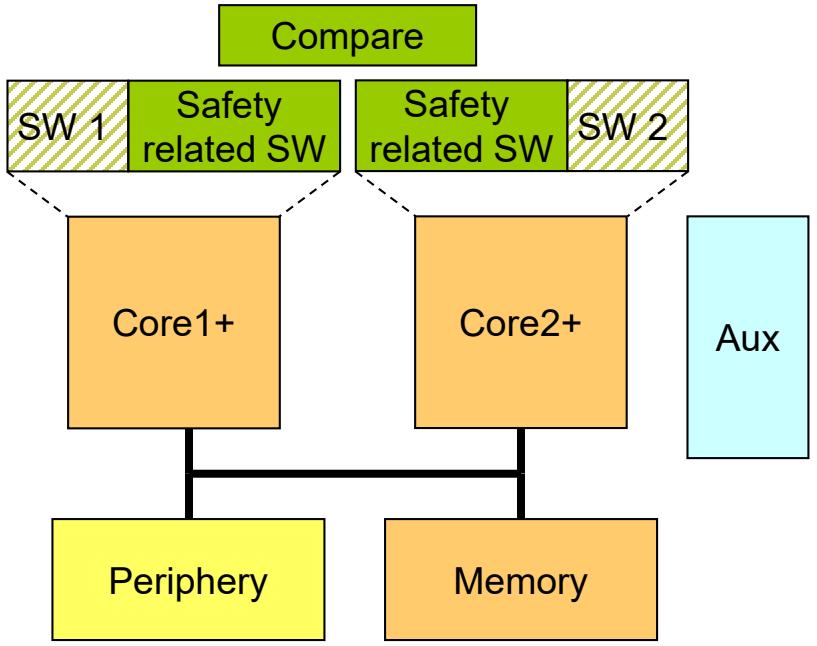
freescale semiconductor

Safety measures against additional ECC failure modes

Measure	MBUs	Fault accumulation	ECC logic error	Addressing Error
SW:	-	Scrubbing Cost: RAM, Bus & Core bandwidth Effect: Error prevented	Known Reads (self test) Cost: RAM, Bus & Core bandwidth Effect: Diagnose some errors	Read known addresses (self test) Cost: Spread out variables in RAM Effect: Diagnose some errors
HW:	Column Multiplexing Cost: Very slight increase in critical path Effect: Prevents error	Scrubbing Cost: RAM bandwidth Effect: Error prevented	Replication Cost: Duplicated RAM Ctrl & ECC logic Effect: Error prevented	Address in ECC Cost: Increase in critical path if no SEC wait state Effect: Diagnose some errors

Two non-locked Cores with SW-based Comparison

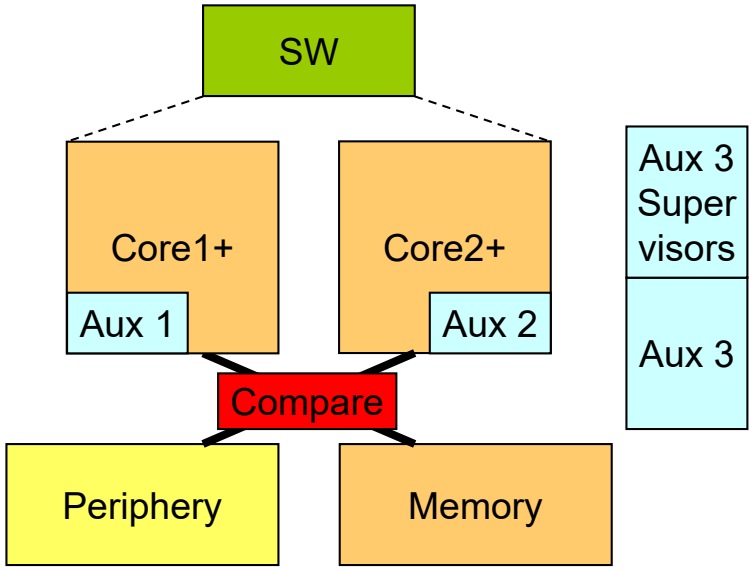
Approach	▶ Two independent cores with (nearly) same capabilities ▶ Non safety-related SW executed on only one core ▶ Safety related SW executed on both cores • Software diversity can be used • Final/intermediate results stored • Compared by further SW on both cores • Verification of peripheral control by write & read back (by other core) ▶ HW measures against Common Cause Failures
SIL Capability	▶ SIL 3 with system safety time in range of calculation cycle time
Distinct Benefits	▶ Low overhead for non-SR SW ▶ Diverse software capability ▶ More cores could be added



freescale™ semiconductor

Two Cores in Lockstep

Approach	▶ Full replication of parts of computational and auxiliary subsystem • Software executed on both cores synchronized in parallel • Outputs of both replicas are compared each execution step • (Comparators themselves replicated) • Failures are detected immediately • No software adaptations necessary ▶ Non replicated components • Supervised (Auxiliaries) • Self-tests (Peripherals) • Dual use (Peripherals) ▶ Cores can be switched into dual configuration (see previous slide) at boot
SIL Capability	▶ SIL 3 & very short system safety times
Distinct Benefits	▶ Highest coverage of faults ▶ Ease of use



freescale™ semiconductor

Comparison of Architectures

		HW Architectures					
		Single Core	Asymmetric Cores	Dual Core	Dual Core Lockstep	SW Effort	Detection of SW Errors
SW Approach	Single SW Instance	-	-	-	ASIL D	-	-
	Replicated SW Instances	ASIL A (Time redundancy)	Not common	ASIL D	Makes little sense	+	-
	Diverse SW Instances	ASIL A-B (Time redundancy)	ASIL C-D	ASIL D	ASIL D (Time redundancy)	++	+
Self-Test	Startup Self-Test	Makes little sense	Required for sufficient LFM	Required for sufficient LFM	Required for sufficient LFM		
	Online Self-Test	ASIL A	Not common	Not common	Not common		
	HW Effort	-	+	+	++		

freescale™ semiconductor

Additional failure mode: Permanent errors in RAM array

- Memory self-test needed?