FMEDA-Based Fault Injection and Data Analysis in Compliance with ISO-26262

Kuen-Long Lu[1, 2] ,Yung-Yuan Chen[1], and Li-Ren Huang[2]

[1]Dept. of Electrical Engineering, National Taipei University

[2]Dept. of Automotive Electronics Design & Application, Division of Biomedical & Industrial IC Technology, ICL, ITRI

2018/06/25

NTPU

工業技術研究院
Industrial Technology
Research Institute

Outline

- Introduction to ISO 26262
- Safety Analysis of ISO 26262 – FMEDA(Failure Modes, Effects, and Diagnostic Analysis)
- FIDA – **F**MEDA-based Fault **I**njection and **D**ata **A**nalysis
- Case Study
- Conclusion

Introduction to ISO 26262

- ISO 26262 is entitled with “Road vehicles – Functional safety”
  - Derived from IEC 61508
  - Focus on automotive functional safety
  - Include one or more electrical and/or electronic (E/E) systems
  - Passenger cars up to 3,500kg
  - Special purpose vehicles are excluded

- ISO 26262 was formally published on 14th Nov, 2011.

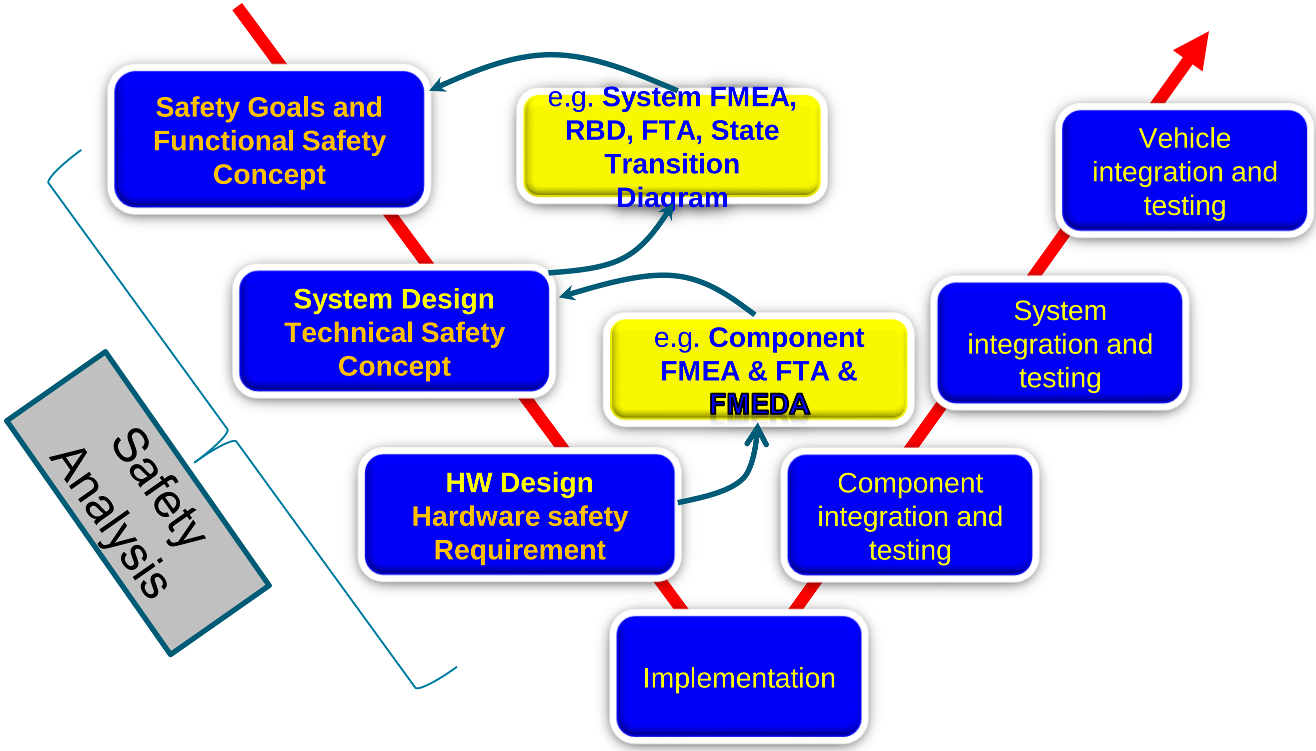
Introduction to ISO 26262

- 2009–11 Toyota vehicle recalls
  - https://en.wikipedia.org/wiki/2009%E2%80%9311_Toyota_vehicle_recalls
    - Michael Barr: Toyota did not follow best practices for real time life critical software, and that a single bit flip which can be caused by cosmic rays could cause unintended acceleration.
    - 37 people dead
    - 9 million vehicles were called back
    - Cost $2 billion

- 2018 May BMW recalls 300,000 cars that risk stalling completely
  - https://www.bbc.com/news/business-44050686
    - The BMW had suffered an electrical fault, causing its brake lights to fail and resulting in the vehicle stalling on a dark A-road.
    - Man was killed on Christmas Day in 2016 when his car swerved to avoid a stalled BMW

Introduction to ISO 26262

❖ Safety Analysis in ISO 26262



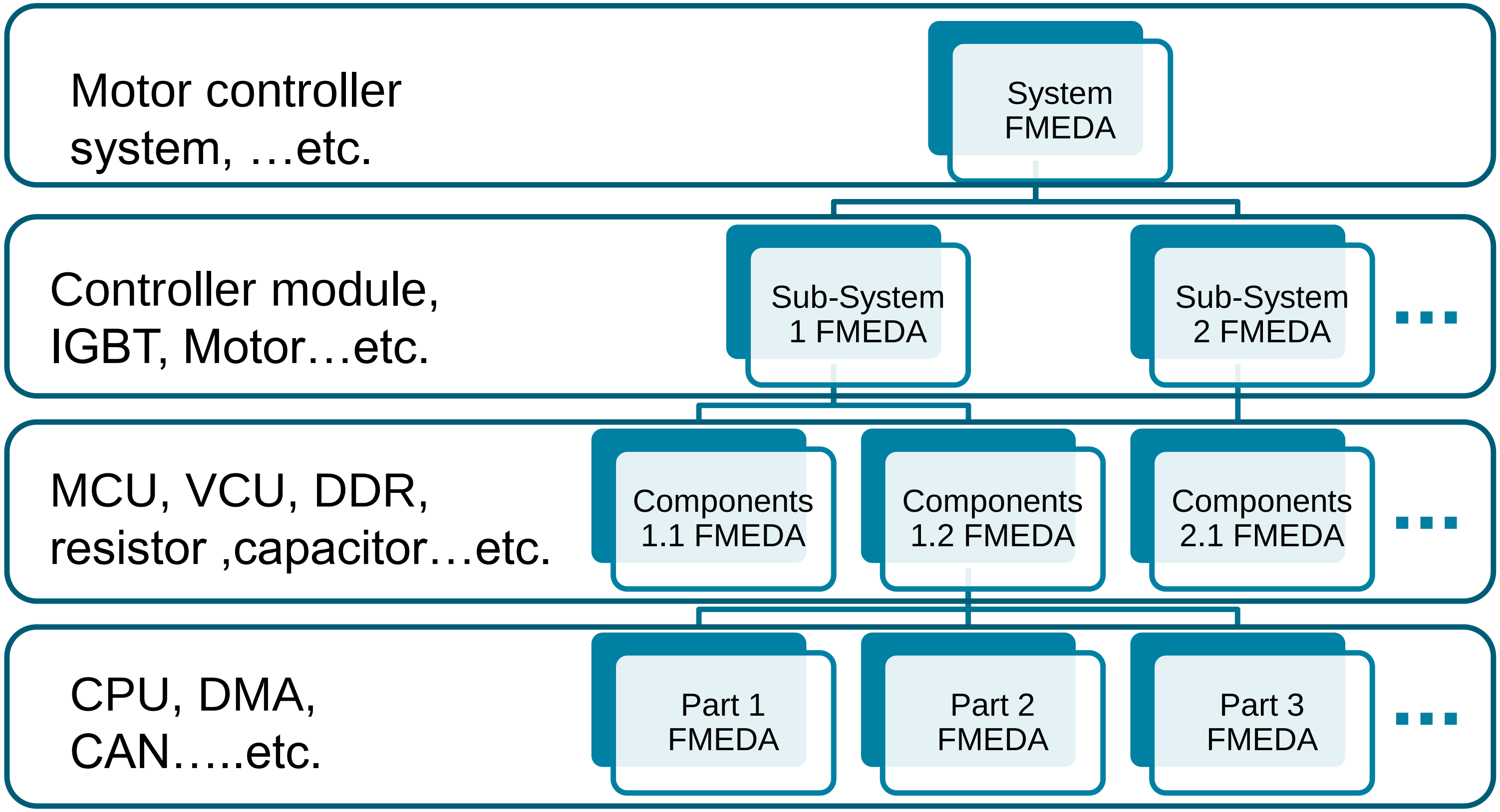
Safety Goals and Functional Safety Concept
e.g. System FMEA, RBD, FTA, State Transition Diagram
Vehicle integration and testing
System Design Technical Safety Concept
e.g. Component FMEA & FTA & FMEDA
System integration and testing
Safety Analysis
HW Design Hardware safety Requirement
Component integration and testing
Implementation

Introduction to ISO 26262

- FMEDA for ISO-26262
  - FMEDA is a systematic analysis technique to obtain subsystem / product level failure rates, failure modes and diagnostic capability, the followings are inputs for FMEDA:
    - All components of a design,
    - The functionality of each component,
    - The failure modes of each component,
    - The effect of each component failure mode on the product functionality,
    - The ability of any automatic diagnostics to detect the failure,
    - The design strength (de-rating, safety factors) and
    - The operational profile (environmental stress factors)

FMEDA for ISO-26262

❖ Hierarchical FMEDA



Motor controller system, …etc.

System FMEDA

Controller module, IGBT, Motor…etc.

Sub-System 1 FMEDA

Sub-System 2 FMEDA

…

MCU, VCU, DDR, resistor ,capacitor…etc.

Components 1.1 FMEDA

Components 1.2 FMEDA

Components 2.1 FMEDA

…

CPU, DMA, CAN…..etc.

Part 1 FMEDA

Part 2 FMEDA

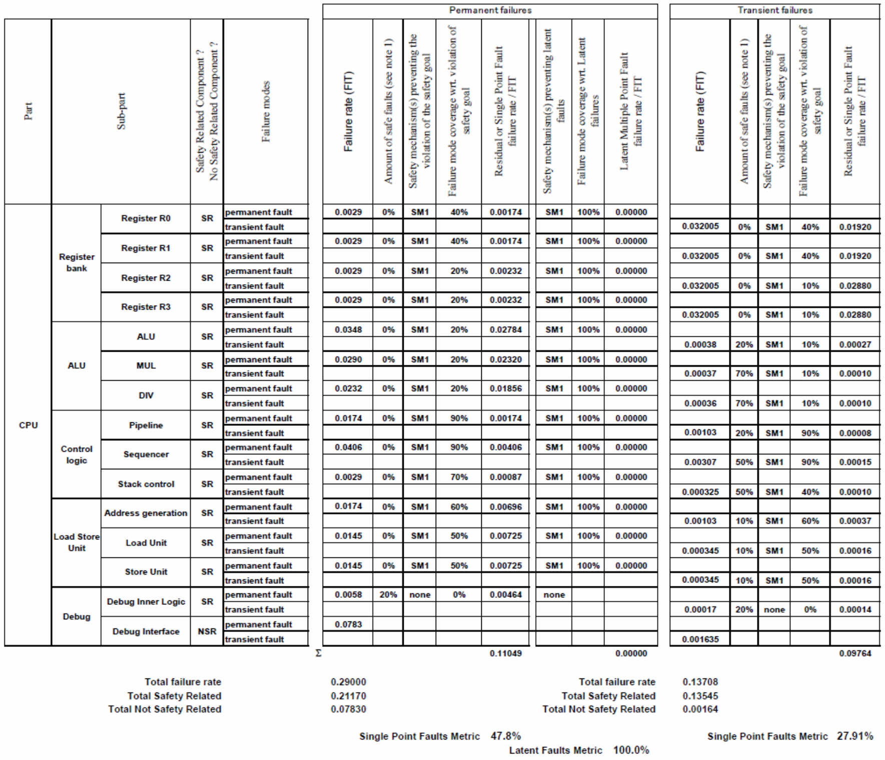
Part 3 FMEDA

…

FMEDA for ISO-26262

❖ Hardware architectural metrics calculation with FMEDA

FMEDA table



Part	Sub-part		Safety Related Component ? No Safety Related Component ?	Failure modes	Permanent failures								Transient failures				
					Failure rate (FIT)	Amount of safe faults (see note 1)	Safety mechanism(s) preventing the violation of the safety goal	Failure mode coverage wrt. violation of safety goal	Residual or Single Point Fault failure rate / FIT	Safety mechanism(s) preventing latent faults	Failure mode coverage wrt. Latent failures	Latent Multiple Point Fault failure rate / FIT	Failure rate (FIT)	Amount of safe faults (see note 1)	Safety mechanism(s) preventing the violation of the safety goal	Failure mode coverage wrt. violation of safety goal	Residual or Single Point Fault failure rate / FIT
CPU	Register bank	Register R0	SR	permanent fault	0.0029	0%	SM1	40%	0.00174	SM1	100%	0.00000					
				transient fault									0.032005	0%	SM1	40%	0.01920
		Register R1	SR	permanent fault	0.0029	0%	SM1	40%	0.00174	SM1	100%	0.00000					
				transient fault									0.032005	0%	SM1	40%	0.01920
		Register R2	SR	permanent fault	0.0029	0%	SM1	20%	0.00232	SM1	100%	0.00000					
				transient fault									0.032005	0%	SM1	10%	0.02880
		Register R3	SR	permanent fault	0.0029	0%	SM1	20%	0.00232	SM1	100%	0.00000					
				transient fault									0.032005	0%	SM1	10%	0.02880
	ALU	ALU	SR	permanent fault	0.0348	0%	SM1	20%	0.02784	SM1	100%	0.00000					
				transient fault									0.00038	20%	SM1	10%	0.00027
		MUL	SR	permanent fault	0.0290	0%	SM1	20%	0.02320	SM1	100%	0.00000					
				transient fault									0.00037	70%	SM1	10%	0.00010
		DIV	SR	permanent fault	0.0232	0%	SM1	20%	0.01856	SM1	100%	0.00000					
				transient fault									0.00036	70%	SM1	10%	0.00010
	Control logic	Pipeline	SR	permanent fault	0.0174	0%	SM1	90%	0.00174	SM1	100%	0.00000					
				transient fault									0.00103	20%	SM1	90%	0.00008
		Sequencer	SR	permanent fault	0.0406	0%	SM1	90%	0.00406	SM1	100%	0.00000					
				transient fault									0.00307	50%	SM1	90%	0.00015
		Stack control	SR	permanent fault	0.0029	0%	SM1	70%	0.00087	SM1	100%	0.00000					
				transient fault									0.000325	50%	SM1	40%	0.00010
	Load Store Unit	Address generation	SR	permanent fault	0.0174	0%	SM1	60%	0.00696	SM1	100%	0.00000					
				transient fault									0.00103	10%	SM1	60%	0.00037
		Load Unit	SR	permanent fault	0.0145	0%	SM1	50%	0.00725	SM1	100%	0.00000					
				transient fault									0.000345	10%	SM1	50%	0.00016
		Store Unit	SR	permanent fault	0.0145	0%	SM1	50%	0.00725	SM1	100%	0.00000					
				transient fault									0.000345	10%	SM1	50%	0.00016
	Debug	Debug Inner Logic	SR	permanent fault	0.0058	20%	none	0%	0.00464	none							
				transient fault									0.00017	20%	none	0%	0.00014
		Debug Interface	NSR	permanent fault	0.0783												
				transient fault									0.001635				
				Σ					0.11049			0.00000					0.09764

Total failure rate 0.29000
Total Safety Related 0.21170
Total Not Safety Related 0.07830

Single Point Faults Metric 47.8%
Latent Faults Metric 100.0%

Total failure rate 0.13708
Total Safety Related 0.13545
Total Not Safety Related 0.00164

Single Point Faults Metric 27.91%

Source: ISO 26262 -5 / 10

$$\text{Single Point Fault Metric} = 1 - \frac{\sum\limits_{\text{Safety Related HW elements}} (\lambda_{SPF} + \lambda_{RF})}{\sum\limits_{\text{Safety Related HW elements}} \lambda}$$

$$\text{Latent Fault Metric} = 1 - \frac{\sum\limits_{\text{Safety Related HW elements}} (\lambda_{MPF,Latent})}{\sum\limits_{\text{Safety Related HW elements}} (\lambda - \lambda_{SPF} - \lambda_{RF})}$$

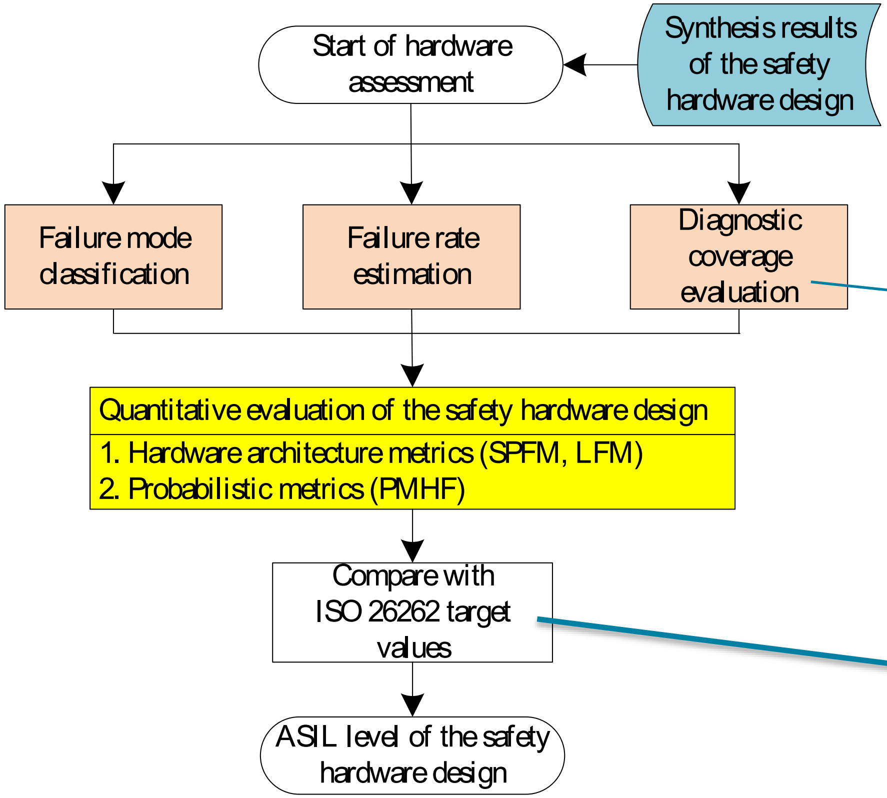
$$\text{PMHF} = \sum \lambda_{\text{SPF}} + \sum \lambda_{\text{RF}} + \sum \lambda_{\text{MPF latent}}$$

	ASIL B	ASIL C	ASIL D
SPFM	> =90%	>= 97%	>= 99%
LFM	>= 60%	>= 80%	>= 90%
PMHF	< 100 FITs	<100 FITs	< 10FITs

PMHF: Probabilistic Metric for Hardware Failure

FMEDA for ISO-26262

❖ FMEDA Practice



Start of hardware assessment

Synthesis results of the safety hardware design

Failure mode classification

Failure rate estimation

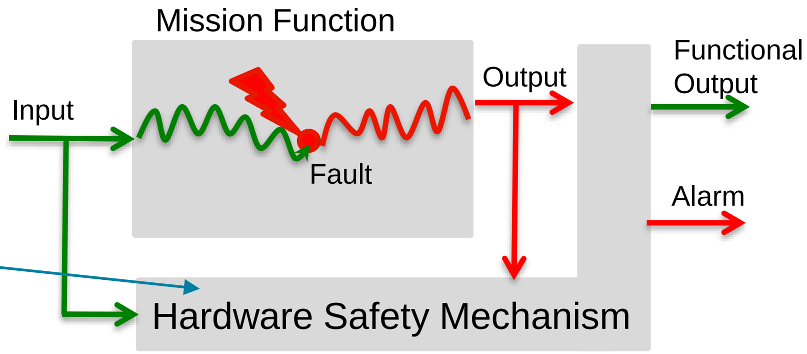
Diagnostic coverage evaluation

Quantitative evaluation of the safety hardware design

1. Hardware architecture metrics (SPFM, LFM)
2. Probabilistic metrics (PMHF)

Compare with ISO 26262 target values

ASIL level of the safety hardware design



- ✓ Run time check correctness of mission function.
- ✓ D.C. indicates the ability of S.M.. and is different with the test coverage of DFT.

	ASIL B	ASIL C	ASIL D
SPFM	> =90%	>= 97%	>= 99%
LFM	>= 60%	>= 80%	>= 90%
PMHF	< 100 FITs (optional)	<100 FITs	< 10FITs

FIDA – FMEDA-based Fault Injection and Data Analysis

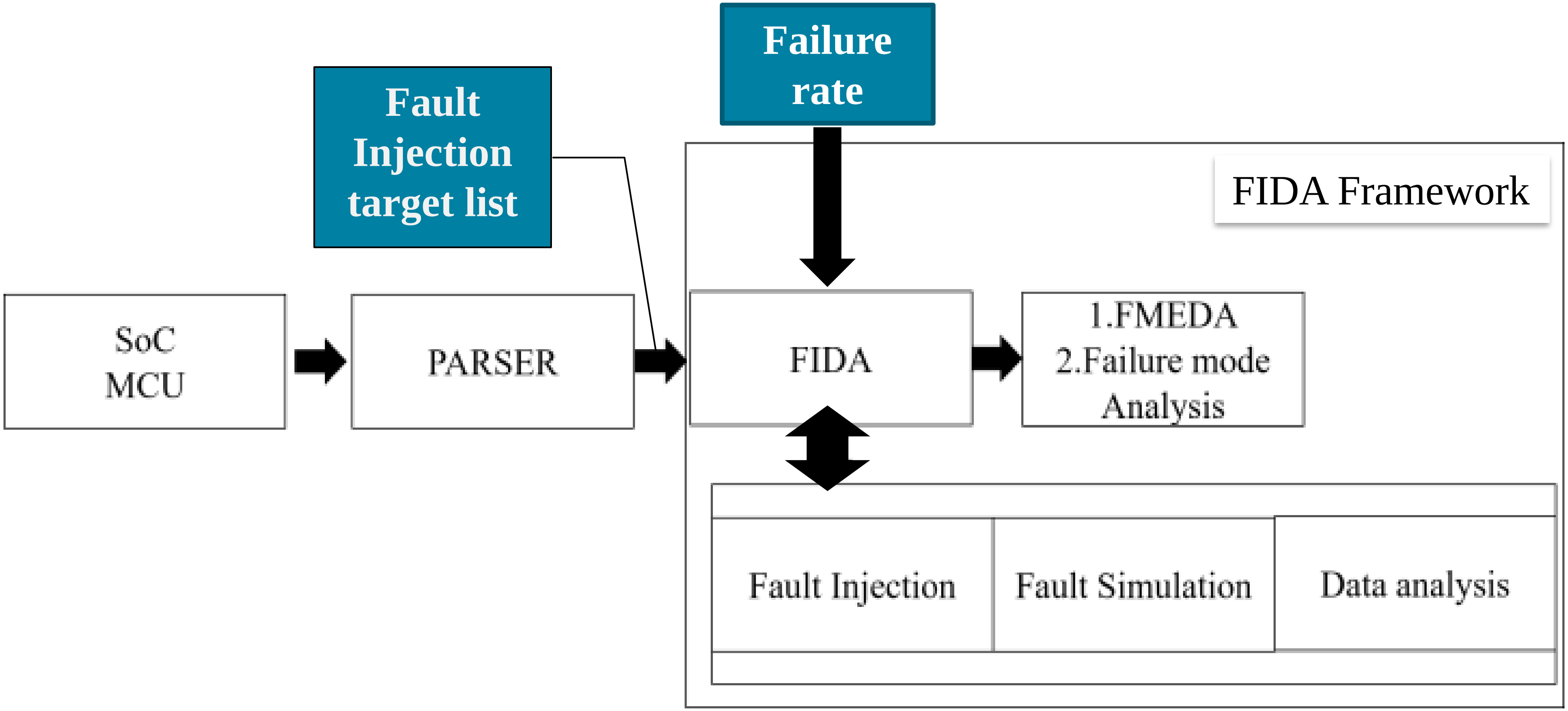
- Fault Injection is ISO-26262
  - Fault injection is highly recommended for HW safety verification, the main merits are summarized as follows:
    - Supporting the evaluation of the HW architectural metrics
      - Evaluating the diagnostic coverage of a safety mechanism
    - Evaluating the diagnostic time interval and the fault reaction time
    - Confirmation of fault impact
    - Evaluating the completeness and correctness of a safety mechanism
      - Demonstrating the completeness and correctness of the safety mechanism to detect faults and control their effect
      - Demonstrating completeness and correctness of the functionality of the safety mechanism with respect to requirements.

FIDA – FMEDA-based Fault Injection and Data Analysis

- Avoidance of Systematic Fault
  - Due to the complexity of modern microcontrollers (hundreds or thousands of parts and sub-parts), to guarantee completeness of the analysis, it is helpful to support the division process with **automatic tools**. (ISO 26262- part 10)
  - the usage of measures for the reproducibility and automation of the design implementation process (**script based, automated work and design implementation flow**); (ISO 26262- part 10)

FIDA – FMEDA-based Fault Injection and Data Analysis

- FIDA Framework



FIDA – FMEDA-based Fault Injection and Data Analysis

- Fault Injection
  - Generate a fault injection campaign constituted by:
    a) number of injected faults and the distribution of different injection targets
    b) fault types like stuck-at-fault or bit-flips
    c) fault duration and fault instance time
    d) permanent or transient faults.
  - Users need to specify items (a) and (d) as the inputs for FIDA. Then items (b) and (c) can be generated by FIDA automatically
    - FIDA determines the fault type and duration for each fault according to the permanent or transient faults specified.

FIDA – FMEDA-based Fault Injection and Data Analysis

- Fault Simulation
  - According to the specified number of injected faults, the same number of test benches are generated by FIDA
  - Only action needed is to execute a FIDA command which will automatically simulates all testbenches
  - After all fault simulations are finished, waveform files in VCD format for all simulated faults are dumped and stored in pre-specified directory
  - A fault-free simulation is also executed to generate the golden waveform.

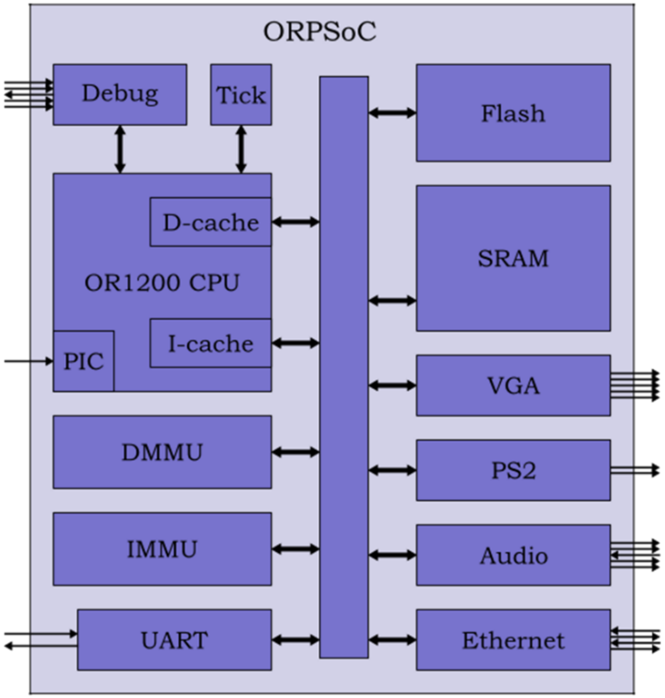
FIDA – FMEDA-based Fault Injection and Data Analysis

Data Analysis

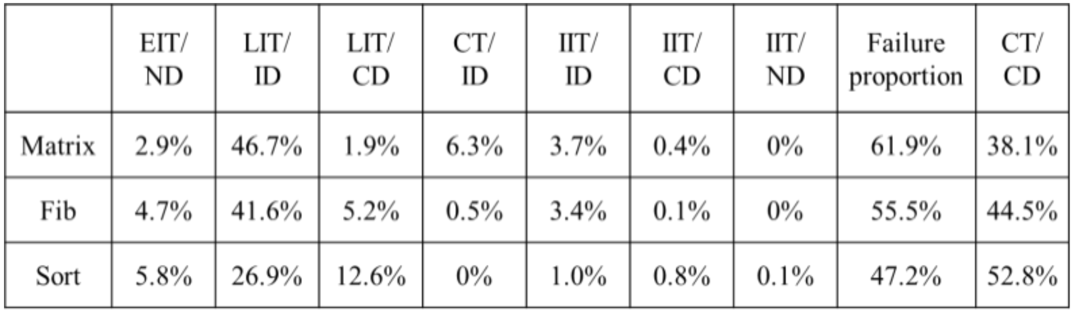
- For each injected fault, FIDA compares following results for the fault simulation and the fault-free simulation
  a) the fault simulation waveform with the golden waveform
  b) SRAM contents
- FIDA repeats the comparison for each injected fault and accumulate the amount of "SoC failures"
- Once all fault simulation results are compared, the total number of SoC failures is acquired
- Then FIDA can calculate the diagnostic coverage (DC) or called failure mode coverage (FMC) in FMEDA report through dividing the number of SoC failures by total number of injected faults
- Finally, the HW architecture metrics can be calculated and filled in FMEDA report.

FIDA – FMEDA-based Fault Injection and Data Analysis

- Data Analysis
  - FIDA generates two reports
  - Case study: An OpenCores SoC
    - SoC failure mode classification
    - Used for up-layer FMEA/FMEDA

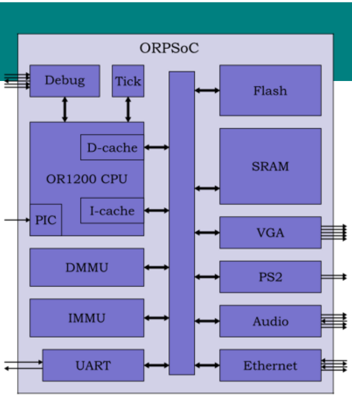


SoC Failure mode	Description
EIT/ID	Simulation ends incorrectly (earlier than expectation) with incorrect results
EIT/CD	Simulation ends incorrectly (earlier than expectation) with correct results
EIT/ND	Simulation ends incorrectly (earlier than expectation) with no results
LIT/ID	Simulation ends incorrectly (later than expectation) with incorrect results
LIT/CD	Simulation ends incorrectly (later than expectation) with correct results
LIT/ND	Simulation ends incorrectly (later than expectation) with no results
IIT/ID	Simulation breaks down with incorrect results
IIT/CD	Simulation breaks down with correct results
IIT/ND	Simulation breaks down with no results
CT/ID	Simulation ends normally with incorrect results
CT/CD	Simulation ends normally with correct results
CT/ND	Simulation ends normally with no results

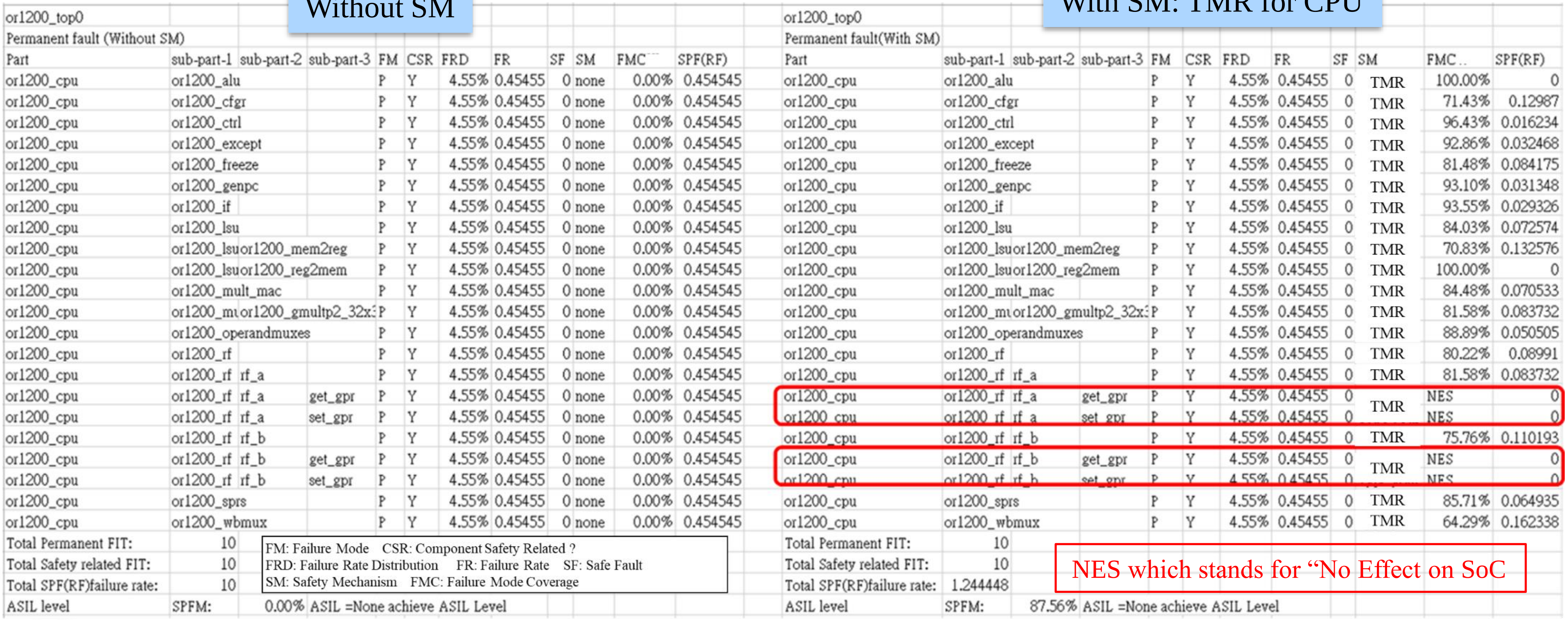


	EIT/ ND	LIT/ ID	LIT/ CD	CT/ ID	IIT/ ID	IIT/ CD	IIT/ ND	Failure proportion	CT/ CD
Matrix	2.9%	46.7%	1.9%	6.3%	3.7%	0.4%	0%	61.9%	38.1%
Fib	4.7%	41.6%	5.2%	0.5%	3.4%	0.1%	0%	55.5%	44.5%
Sort	5.8%	26.9%	12.6%	0%	1.0%	0.8%	0.1%	47.2%	52.8%

FIDA – FMEDA-based Fault Injection and Data Analysis



- Data Analysis
  - FIDA generates two reports
  - Case study: An OpenCores SoC
    - FMEDA report (Only CPU part shows below)



Without SM

or1200_top0											
Permanent fault (Without SM)											
Part	sub-part-1	sub-part-2	sub-part-3	FM	CSR	FRD	FR	SF	SM	FMC	SPF(RF)
or1200_cpu	or1200_alu			P	Y	4.55%	0.45455	0	none	0.00%	0.454545
or1200_cpu	or1200_cfgr			P	Y	4.55%	0.45455	0	none	0.00%	0.454545
or1200_cpu	or1200_ctrl			P	Y	4.55%	0.45455	0	none	0.00%	0.454545
or1200_cpu	or1200_except			P	Y	4.55%	0.45455	0	none	0.00%	0.454545
or1200_cpu	or1200_freeze			P	Y	4.55%	0.45455	0	none	0.00%	0.454545
or1200_cpu	or1200_genpc			P	Y	4.55%	0.45455	0	none	0.00%	0.454545
or1200_cpu	or1200_if			P	Y	4.55%	0.45455	0	none	0.00%	0.454545
or1200_cpu	or1200_lsu			P	Y	4.55%	0.45455	0	none	0.00%	0.454545
or1200_cpu	or1200_lsu	or1200_mem2reg		P	Y	4.55%	0.45455	0	none	0.00%	0.454545
or1200_cpu	or1200_lsu	or1200_reg2mem		P	Y	4.55%	0.45455	0	none	0.00%	0.454545
or1200_cpu	or1200_mult_mac			P	Y	4.55%	0.45455	0	none	0.00%	0.454545
or1200_cpu	or1200_mı	or1200_gmultp2_32x:		P	Y	4.55%	0.45455	0	none	0.00%	0.454545
or1200_cpu	or1200_operandmuxes			P	Y	4.55%	0.45455	0	none	0.00%	0.454545
or1200_cpu	or1200_rf			P	Y	4.55%	0.45455	0	none	0.00%	0.454545
or1200_cpu	or1200_rf	rf_a		P	Y	4.55%	0.45455	0	none	0.00%	0.454545
or1200_cpu	or1200_rf	rf_a	get_gpr	P	Y	4.55%	0.45455	0	none	0.00%	0.454545
or1200_cpu	or1200_rf	rf_a	set_gpr	P	Y	4.55%	0.45455	0	none	0.00%	0.454545
or1200_cpu	or1200_rf	rf_b		P	Y	4.55%	0.45455	0	none	0.00%	0.454545
or1200_cpu	or1200_rf	rf_b	get_gpr	P	Y	4.55%	0.45455	0	none	0.00%	0.454545
or1200_cpu	or1200_rf	rf_b	set_gpr	P	Y	4.55%	0.45455	0	none	0.00%	0.454545
or1200_cpu	or1200_sprs			P	Y	4.55%	0.45455	0	none	0.00%	0.454545
or1200_cpu	or1200_wbmux			P	Y	4.55%	0.45455	0	none	0.00%	0.454545
Total Permanent FIT:	10										
Total Safety related FIT:	10										
Total SPF(RF)failure rate:	10										
ASIL level	SPFM:	0.00%	ASIL =None achieve ASIL Level								

FM: Failure Mode CSR: Component Safety Related ?
FRD: Failure Rate Distribution FR: Failure Rate SF: Safe Fault
SM: Safety Mechanism FMC: Failure Mode Coverage

With SM: TMR for CPU

or1200_top0											
Permanent fault(With SM)											
Part	sub-part-1	sub-part-2	sub-part-3	FM	CSR	FRD	FR	SF	SM	FMC	SPF(RF)
or1200_cpu	or1200_alu			P	Y	4.55%	0.45455	0	TMR	100.00%	0
or1200_cpu	or1200_cfgr			P	Y	4.55%	0.45455	0	TMR	71.43%	0.12987
or1200_cpu	or1200_ctrl			P	Y	4.55%	0.45455	0	TMR	96.43%	0.016234
or1200_cpu	or1200_except			P	Y	4.55%	0.45455	0	TMR	92.86%	0.032468
or1200_cpu	or1200_freeze			P	Y	4.55%	0.45455	0	TMR	81.48%	0.084175
or1200_cpu	or1200_genpc			P	Y	4.55%	0.45455	0	TMR	93.10%	0.031348
or1200_cpu	or1200_if			P	Y	4.55%	0.45455	0	TMR	93.55%	0.029326
or1200_cpu	or1200_lsu			P	Y	4.55%	0.45455	0	TMR	84.03%	0.072574
or1200_cpu	or1200_lsu	or1200_mem2reg		P	Y	4.55%	0.45455	0	TMR	70.83%	0.132576
or1200_cpu	or1200_lsu	or1200_reg2mem		P	Y	4.55%	0.45455	0	TMR	100.00%	0
or1200_cpu	or1200_mult_mac			P	Y	4.55%	0.45455	0	TMR	84.48%	0.070533
or1200_cpu	or1200_mı	or1200_gmultp2_32x:		P	Y	4.55%	0.45455	0	TMR	81.58%	0.083732
or1200_cpu	or1200_operandmuxes			P	Y	4.55%	0.45455	0	TMR	88.89%	0.050505
or1200_cpu	or1200_rf			P	Y	4.55%	0.45455	0	TMR	80.22%	0.08991
or1200_cpu	or1200_rf	rf_a		P	Y	4.55%	0.45455	0	TMR	81.58%	0.083732
or1200_cpu	or1200_rf	rf_a	get_gpr	P	Y	4.55%	0.45455	0	TMR	NES	0
or1200_cpu	or1200_rf	rf_a	set_gpr	P	Y	4.55%	0.45455	0		NES	0
or1200_cpu	or1200_rf	rf_b		P	Y	4.55%	0.45455	0	TMR	75.76%	0.110193
or1200_cpu	or1200_rf	rf_b	get_gpr	P	Y	4.55%	0.45455	0	TMR	NES	0
or1200_cpu	or1200_rf	rf_b	set_gpr	P	Y	4.55%	0.45455	0		NES	0
or1200_cpu	or1200_sprs			P	Y	4.55%	0.45455	0	TMR	85.71%	0.064935
or1200_cpu	or1200_wbmux			P	Y	4.55%	0.45455	0	TMR	64.29%	0.162338
Total Permanent FIT:	10										
Total Safety related FIT:	10										
Total SPF(RF)failure rate:	1.244448										
ASIL level	SPFM:	87.56%	ASIL =None achieve ASIL Level								

NES which stands for “No Effect on SoC

Conclusion

- In this study, a FMEDA-based fault Injection and data analysis framework in compliance with ISO-26262 is proposed
- Through the proposed three-phase framework with developed tool – FIDA, fault simulations and data analysis for the simulation results are executed automatically
- FIDA can also automatically perform failure mode classification and FMEDA report generation so that the designer can rapidly recognize the weakness of HW and establish safety mechanism to improve the safety level.
  - Therefore the effort to achieve the demanded HW's safety level is effectively reduced.

Thank you very much for your attention!