AURIX 2G LBIST Module Detail

Thomas
IFCN ATV SMD GC SAE MC
2018/5/2

History List / Changes

- Module Name: LBIST
- Module Owner: Martin Musiol / Vladimir Litovtchenko
- Slide Target: LBIST Detail
- Slide Version: 1.0
- Slide Status: Released
- Last Modification: 05/05/2017
- Revision History:
  - 1.0 First version for TC3xx

LBIST Scope

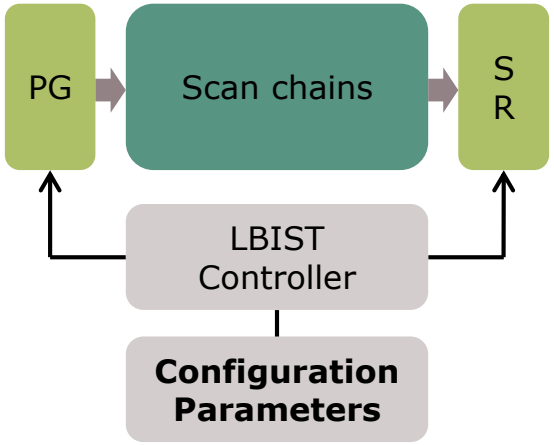
- LBIST is a safety mechanism that aims are preventing dual-point faults from being latent and therefore contributes to the achievement of the Latent Fault Metric targets
  - It is not the only safety mechanisms to achieve the LFM targets for information:
    - Lockstep Comparator Periodic HW-Self Test
      - Applicable to any LOCKSTEP instance: CPU, SPU
    - CPU and SPU SBST
    - Redundant SMU in Standby domain to monitor main SMU in core domain
    - Analog BIST for the voltage monitor and Band Gaps
    - Analog Test Modes for the ADCs
    - …

LBIST Characteristics (short summary)

- LBIST Operation
  - During LBIST operation the entire MCU enters test mode
    - The MCU is not functionally available
    - The MCU state before LBIST is lost
    - LBIST terminates with a LBIST reset
- LBIST control
  - LBIST can be started by boot Firmware (based on boot option)
  - LBIST can be started by application software
- LBIST Parameters
  - Frequency, Split-Shift, Number of Patterns, Initial Seed, Final Signature
- Result = LBIST Diagnostic Coverage
  - Is dependent from the LBIST Parameters and, for sure, device complexity
  - IFX will simulate/validate per device several sets of {LBIST Parameters, Final Signature, Final DC} to be used by the application

TC39x Logic BIST

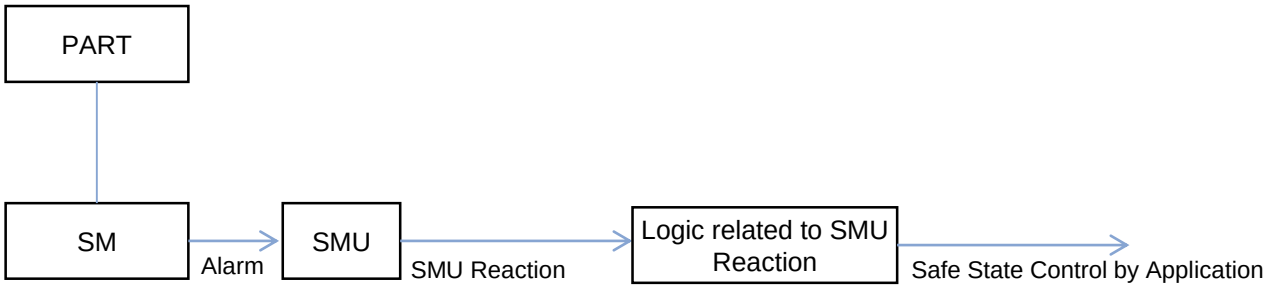
- Logic BIST (LBIST)
  - Logic built-in self test (LBIST) is being used in SoCs for increasing safety and to provide a self-testing capability
  - Pattern generator (PG) is used for generating the test stimuli for the scan input
  - Signature Registers (SR) are used for collecting the scan output
  - If the collected signature matches with the golden or expected signature, the LBIST status is “Pass”.



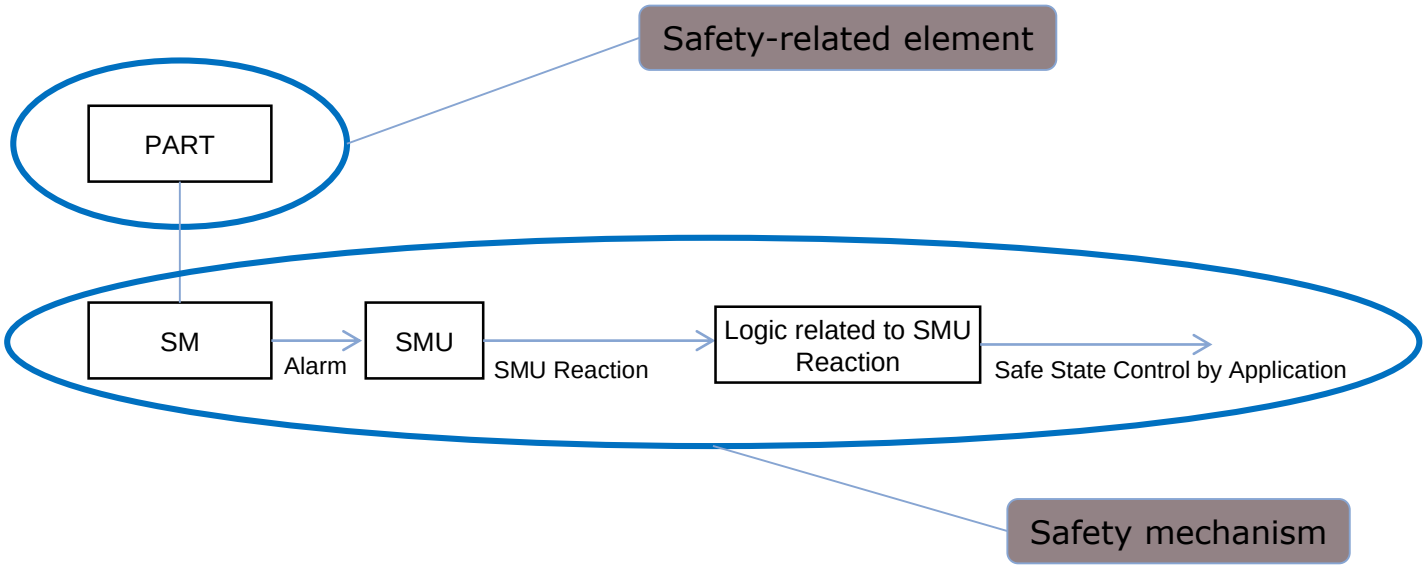
- TC39x implementation
  - TC39x A/B-Step have a configurable LBIST implemented
  - LBIST can be configured to be automatically triggered; LBIST trigger with a SCU-Register bit
  - Runtime is a function of frequency, number of scan chains (split) and number of patterns (shifts)
    - Start of extensive set of patterns requires up to 800ms runtime at a clock rate of 16.6MHz (clock limit)
  - TC39x A-Step: Number of scan shifts is configurable, e.g. 512 -> ca. 40ms runtime, coverage limited
  - TC39x B-step: similar to A-Step, mix of LBIST and other safety measures to achieve LFM targets

Contributors to the LFM

- Generic Overview assuming a given safety mechanism SM
  - SM detects failure modes of given PART (mission logic) and prevents them from being latent by notifying their presence to SMU via given alarm(s)
  - We assume SM is only based on fault detection.
  - The SMU propagates the alarm to the application by the defined interfaces: ErrorPin, NMI, Reset(s), Interrupts.

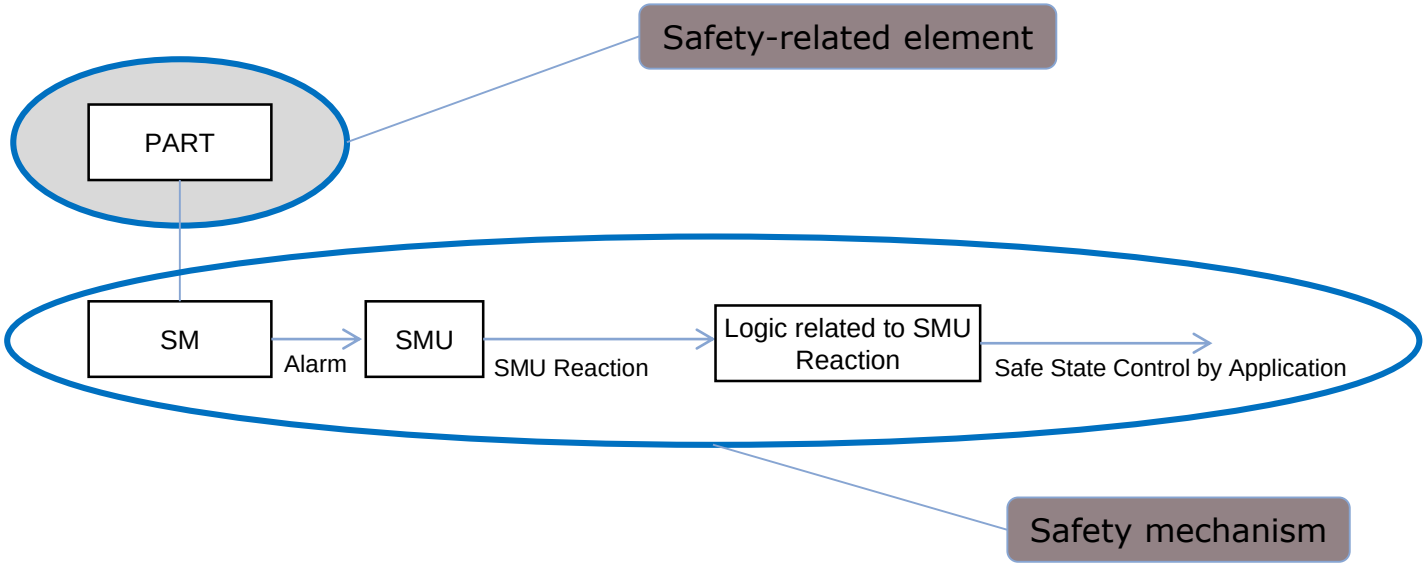


Contributors to the LFM (Cont'd)



Contributors to the LFM (Cont'd)

› **Independent** dual-point faults of PART

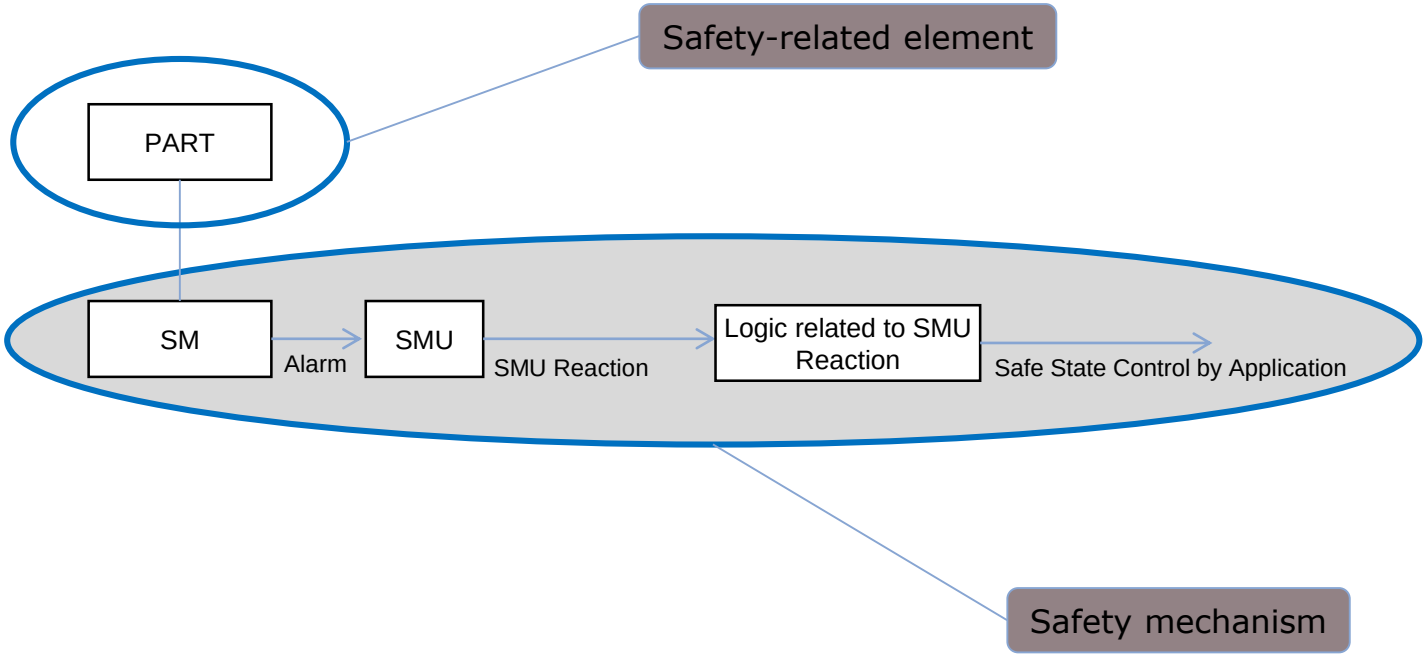


› The first set of dual point faults defined in ISO2626 comprises all the PART faults that are detected by SM and reported to SMU. We call them PART DPF,D (PART Dual Point Faults Detected)

- If SM has a coverage of 99% over PART failure modes, 99% of PART PVSG Failure Rate contribute to the PART DPF,D failure rate
- The remaining 1% are the residual faults of PART

Contributors to the LFM (Cont'd)

› **Independent** dual-point faults of Safety Mechanism

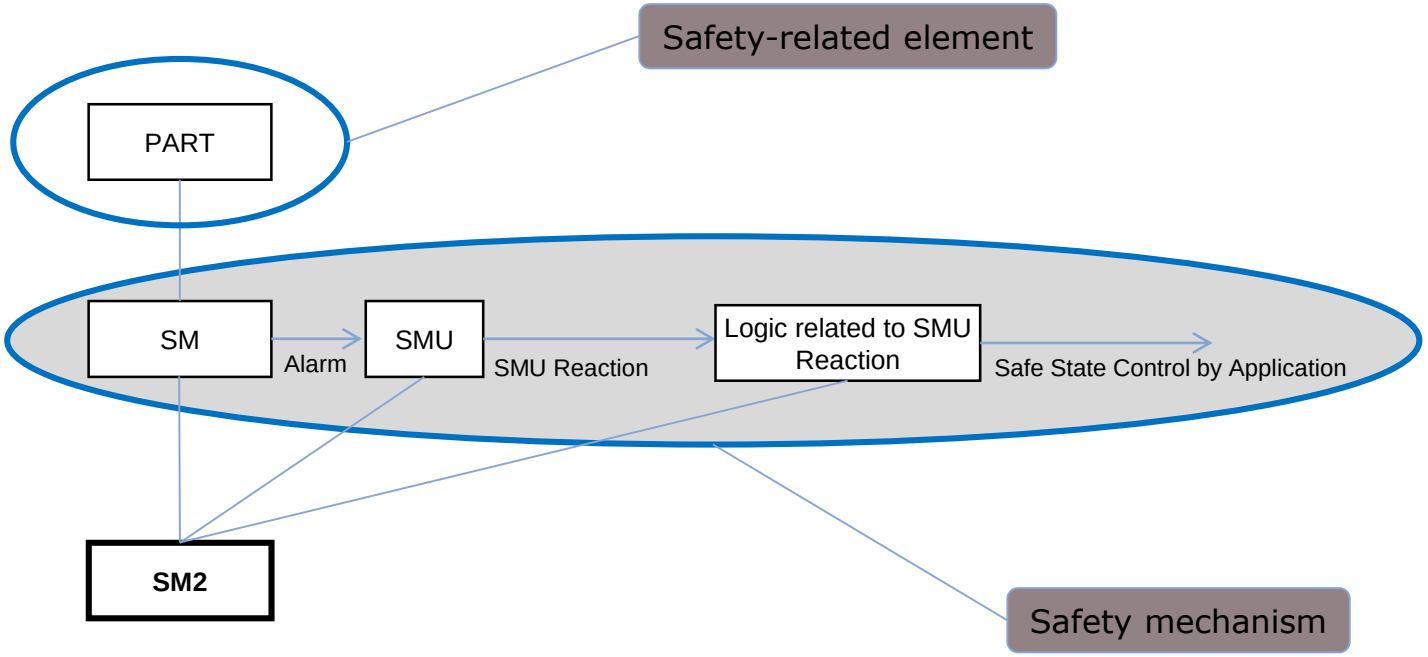


› The safety mechanism: SM, SMU, and Logic related to SMU reaction, represent the second independent set of Dual Point Faults to consider.

- We call them SM DPF.
- If there is no mean to test the safety mechanism, all the SM DPF are Latent → SM DPF,L

Contributors to the LFM (Cont'd)

› Coverage of SM dual-point faults



› Dual point faults of safety mechanisms can be tested by another safety mechanism (here called SM2) at startup or shutdown or run-time (if not intrusive) to reduce the proportion of SM DPF,L.

– Let's assume **SM2 = Integrated Logic BIST (LBIST)**

ISO26262 requirements for LFM

› ISO26262-5 8.4.6 sets the quantitative targets to be achieved for the safety goal(s)

Table 5 — Possible source for the derivation of the target "latent-fault metric" value

	ASIL B	ASIL C	ASIL D
Latent-fault metric	≥60 %	≥80 %	≥90 %

› At MCU level a possible strategy is to apply the same targets for given set of PART as defined by the MCU safety concept.

- E.g. Compute cluster made of CPU + tightly coupled memories
- E.g. SPU + Radar Memories

LFM Parameters

Latent Fault Metric Definition:

$$LFM = \frac{\text{DPF,D Dual Point Fault Detected} + \text{Safe}}{\text{DPF,D Dual Point Fault Detected} + \text{DPF,L Dual Point Fault Latent} + \text{Safe}}$$

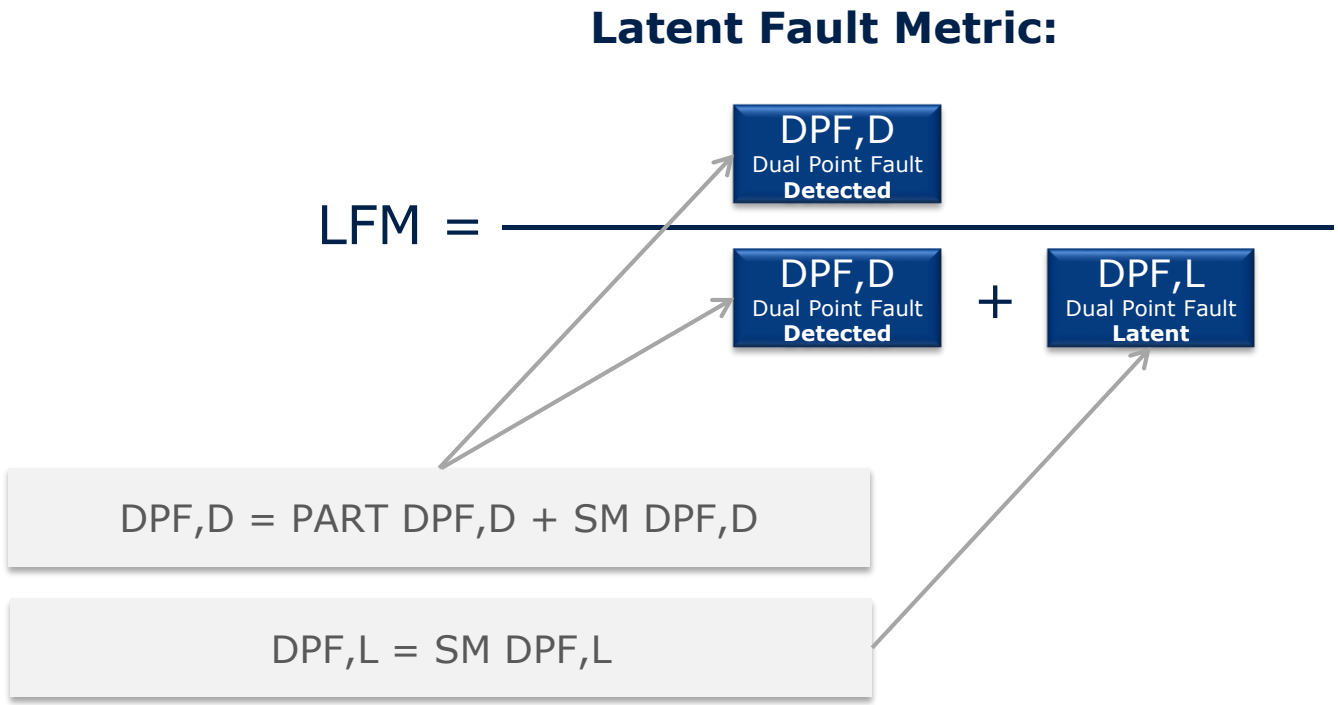
› For now we are going to assume that Safe Faults = 0

Latent Fault Metric:

$$LFM = \frac{\text{DPF,D Dual Point Fault Detected}}{\text{DPF,D Dual Point Fault Detected} + \text{DPF,L Dual Point Fault Latent}}$$

LFM Parameters Mapping

- Contribution of PART and SM DPF



LFM Parameters Mapping (Cont'd)

- What are the parameters that influence the achievement of the LFM targets
  - The Diagnostic Coverage of SM2 = LBIST over the Dual Point Faults of SM
    - LBIST reduces the amount of SM DPF,L
  - The relative area of PART and SM
    - If PART dominates the DPF,D mainly come from PART and DPF,L from SM have less importance.
    - The relative area also depends on the DC of SM over the PART failures modes. With a low DC the proportion of PART DPF,D is lower

Latent Fault Metric:

$$LFM = \frac{DPF,D \text{ Dual Point Fault } \textbf{Detected}}{DPF,D \text{ Dual Point Fault } \textbf{Detected} + DPF,L \text{ Dual Point Fault } \textbf{Latent}}$$

Derivation of LBIST Requirements

- The Model PART, SM, SMU is captured in the FMEDA tool
  - The parameters are:
    - Area ratios of $SM_{area}/PART_{area}$
    - SM DC over PART Failure Modes. A fixed DC of 99% has been used
    - LBIST Diagnostic Coverage values

Derivation of LBIST Requirements (Cont'd)

› LFM Results = F(SM/PART area ratio, LBIST DC)

		Relative area SM vs PART							
		5%	10%	15%	25%	50%	75%	100%	115%
LBIST DC	0	95,19	90,83	86,84	79,84	66,44	56,90	49,75	46,26
	10	95,67	91,74	88,16	81,85	69,80	61,21	54,77	51,64
	20	96,15	92,66	89,47	83,87	73,15	65,52	59,80	57,01
	30	96,63	93,58	90,79	85,89	76,51	69,83	64,82	62,38
	40	97,12	94,50	92,11	87,90	79,87	74,14	69,85	67,76
	50	97,60	95,41	93,42	89,92	83,22	78,45	74,87	73,13
	60	98,08	96,33	94,74	91,94	86,58	82,76	79,90	78,50
	70	98,56	97,25	96,05	93,95	89,93	87,07	84,92	83,88
	80	99,04	98,17	97,37	95,97	93,29	91,38	89,95	89,25
	85	99,28	98,62	98,03	96,98	94,97	93,53	92,46	91,94
	90	99,52	99,08	98,68	97,98	96,64	95,69	94,97	94,63

LBIST Stuck-At Coverage

AURIX2G Step B LBIST Concept

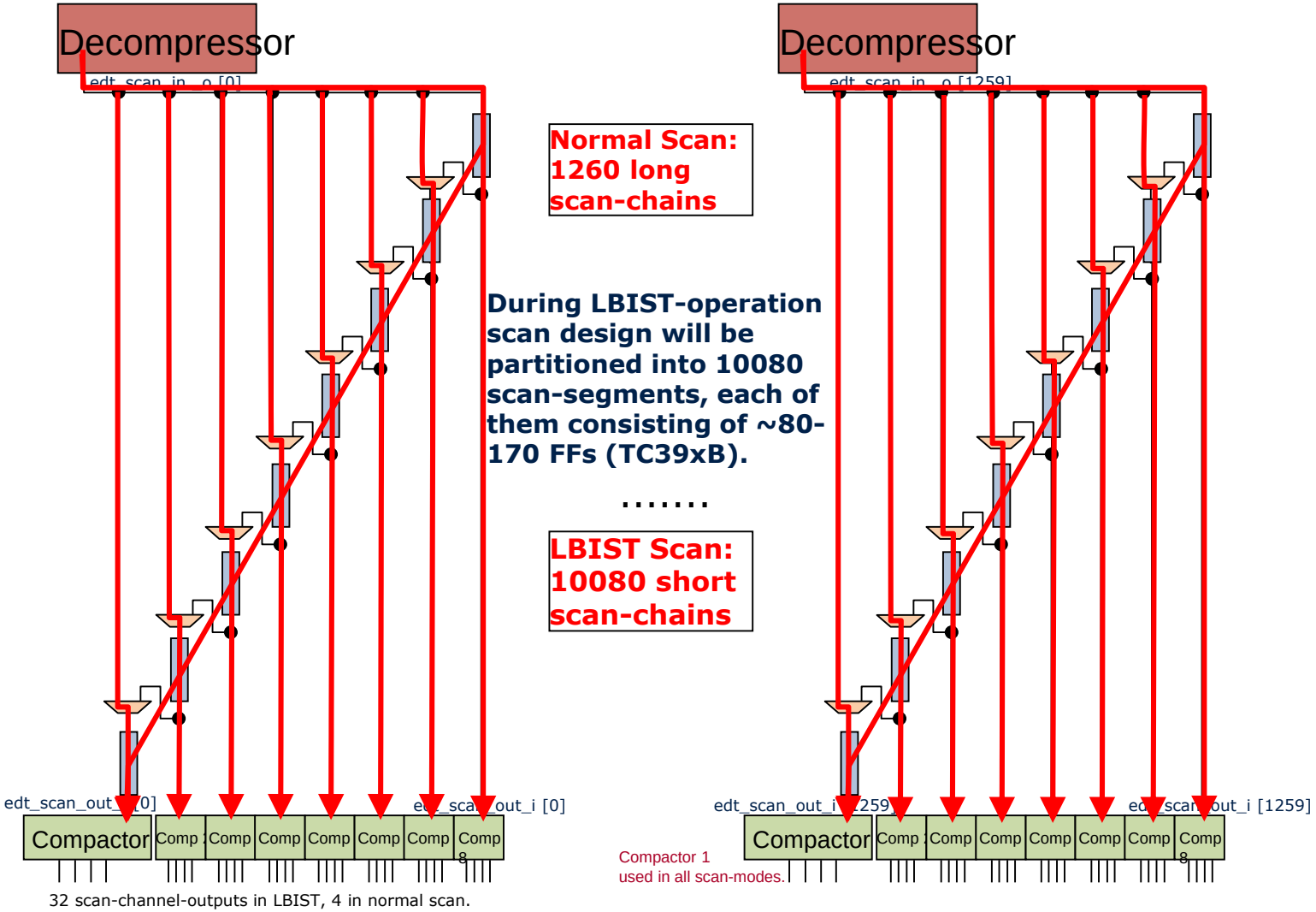
- Lessons learned from AURIX
  1. LBIST runtime dominated by time for shifting of scan chain
     → Execution time too high if several thousand scan-loads are applied w/o exceeding power limits.
  2. Efficiency of random LBIST scan-patterns significantly lower compared to deterministic ATPG-patterns.
     → Test-coverage saturating below 90% coverage target.
- Possible counter measures:
  1. Reduce number of shift cycles during LBIST application
     → Increase number of pseudo-random patterns which can be applied within given timeframe.
  2. Improve circuit testability of safety-relevant IP to facilitate stuck-at fault detection by pseudo-random test patterns.

AURIX2G Step B LBIST Concept

- Objectives for AURIX2G Step B Concept
  1. Scan-chain-length Reduction:
     - Individual scan chain configuration defined for production test and LBIST (Multimode Architecture).
     - Divide each scan-chain into 8 shorter segments during LBIST-execution → Increase scan-chain count to 10.080 if acceptable from implementation side (routing complexity).
  2. Coverage Improvement
     - To support Functional Safety requirements (LFM)
     - To support quality requirements (garage tests)

AURIX2G Step B LBIST Concept

› Multimode Scan-Architecture



AURIX 2G Step B LBIST Concept
Updates

1. Scan-chain-length Reduction Target for Aurix 2G:

› Individual scan chain configuration defined for production test and LBIST (Multimode Architecture).

› Divide each scan-chain into 8 shorter segments during LBIST-execution → Increase scan-chain count to 10.080 if acceptable from implementation side (routing complexity).

➔ LBIST-Performance calculation example for TC39xB:

- Reduce run-time through shorter scan-chains (~170 FFs).
- 15k scan-loads can be executed within ~612-648ms (frequency: 16.6Mhz, split-shift 4, dynamic power: ~300mA).

› For smaller AURIX 2G products LBIST execution time will shrink with reduction of flip-flops.

➔ Goal: Reach reasonable LBIST-coverage during system-startup through reduced shift-time.

Aurix 2G B-Step LBIST Concept
Example: TC39x A-Step

› Coverage vs. Run-time calculation examples for 16.6 Mhz & split-shift 4:

Scan-chain length 270FFs for TC39xA

Scan-chain length 170FFs for TC39xB

		LBIST Exec Time depending on FFs in longest scan-chain (in ms)								
# Scan Loads	LBIST Stuck-At Coverage	450	400	350	300	250	200	170	150	130
64	61.38% (>92% LFM)	7.0	6.2	5.4	4.7	3.9	3.1	2.7	2.4	2.0
128	65.94%	14.0	12.4	10.9	9.3	7.8	6.2	5.3	4.7	4.1
192	68.30%	20.9	18.6	16.3	14.0	11.7	9.4	8.0	7.1	6.1
256	70.02% (>93% LFM)	27.9	24.8	21.7	18.7	15.6	12.5	10.6	9.4	8.2
320	71.19%	34.9	31.0	27.2	23.3	19.5	15.6	13.3	11.8	10.2
384	72.17%	41.9	37.2	32.6	28.0	23.4	18.7	16.0	14.1	12.3
448	72.94%	48.8	43.5	38.1	32.7	27.3	21.9	18.6	16.5	14.3
512	73.63%	55.8	49.7	43.5	37.3	31.2	25.0	21.3	18.8	16.3
576	74.17%	62.8	55.9	48.9	42.0	35.0	28.1	23.9	21.2	18.4
640	74.67% (>94% LFM)	69.8	62.1	54.4	46.7	38.9	31.2	26.6	23.5	20.4
1024	76.78%	111.7	99.3	87.0	74.6	62.3	50.0	42.6	37.6	32.7
2048	~79%	223.3	198.6	174.0	149.3	124.6	99.9	85.1	75.3	65.4
4032	~81% (>96% LFM)	439.6	391.1	342.5	293.9	245.3	196.7	167.6	148.2	128.7
10048	84.18%	1095.6	974.5	853.5	732.4	611.4	490.3	417.7	369.2	320.8
20000	85.63% (>97% LFM)	2180.7	1939.8	1698.8	1457.8	1216.9	975.9	831.3	734.9	638.6

Figure refers to top-level stuck-at coverage.
For safety-critical IP TC is higher (e.g. 90% for 20k scan-loads)

Aurix 2G B-Step LBIST Concept
Updates

2. Coverage Improvement

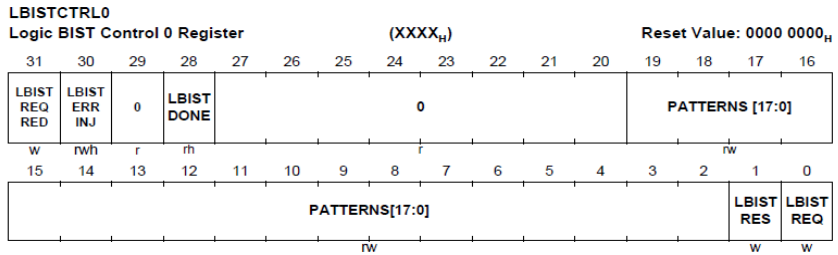
› Structure of safety relevant IPs like CPUs, GTM, PMU will be improved to show better random pattern ATPG coverage.

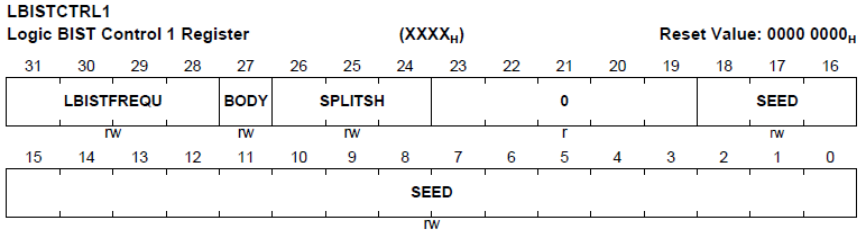
› Can be done during synthesis through automatic test point insertion tools.

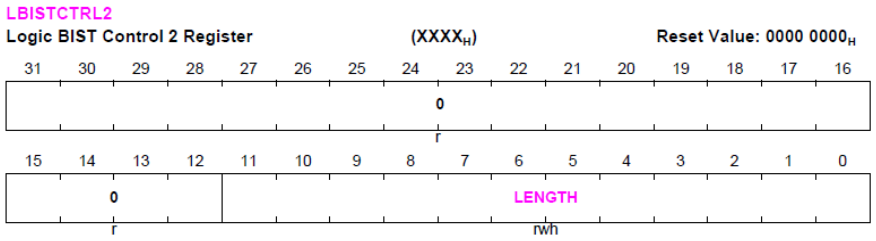
› Example: For GTM random test coverage can be improved on IP-level from 74.2% to 90.7% (with 20k scan-loads) by adding 10.000 test points (1.7% area increase).

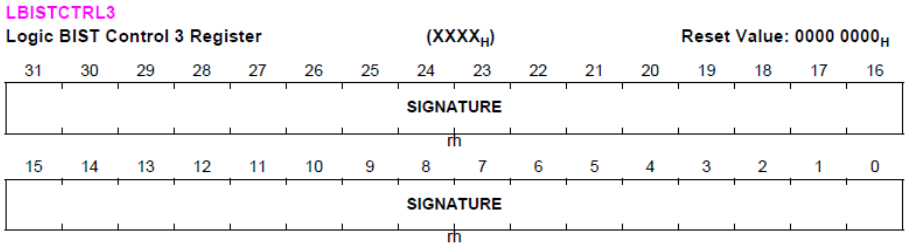
AURIX 2G B-Step LBIST Concept
Functional Changes

- Signature increased to 32bit (4 inputs from 8 scan-compactors).
- Scan-chain shift-count now configured via LBISTCTRL2-reg (scan-chain-length will be stored in Flash and automatically transferred to LBISTCTRL2-register during boot-sequence).
- Number of scan-loads can be determined in range of 1-262k (in AURIX 1G they have been limited to 65k).
- Execution frequency and split-shift settings identical to A1G.
- LBIST-done flag can now be reset and LBIST restarted in a functional way to allow multiple LBIST runs w/o external PORST.







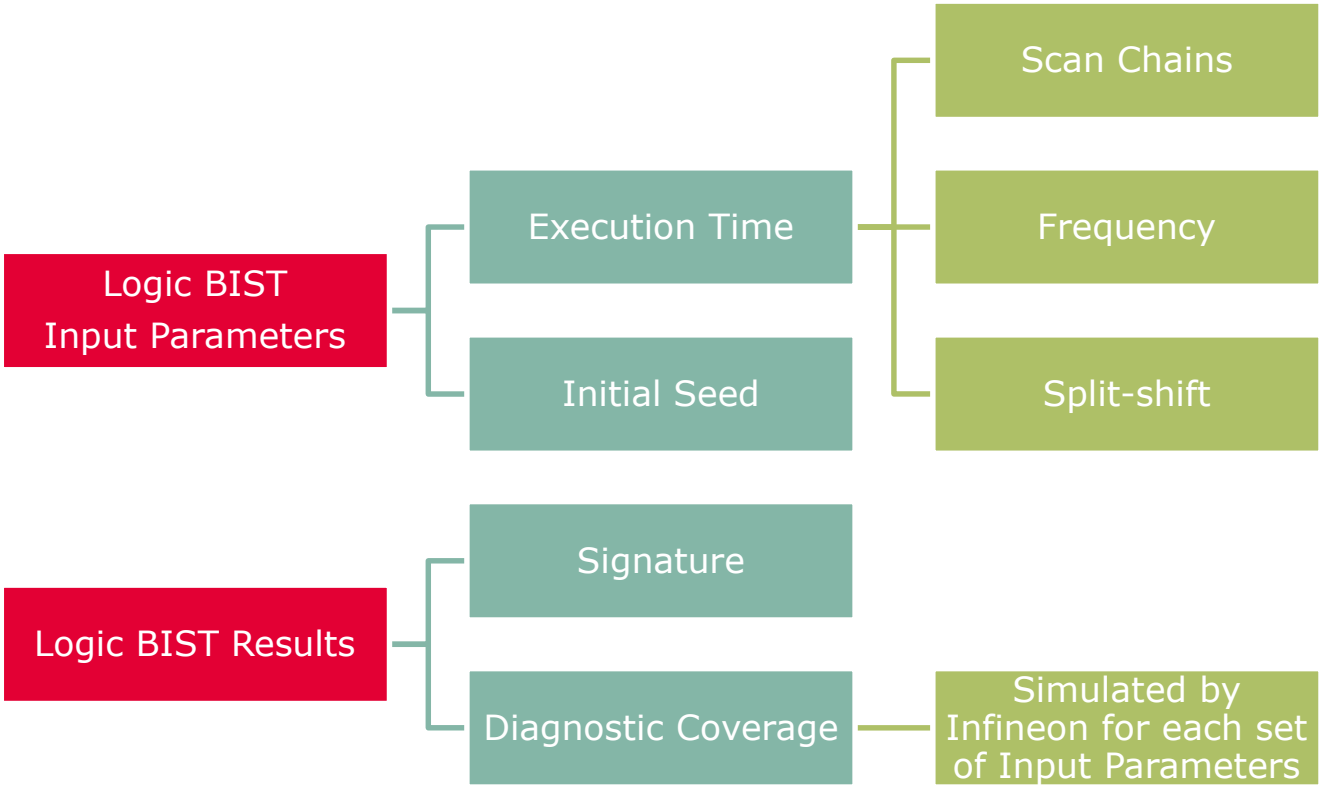


AURIX 2G Step B LBIST Concept
Control during Startup-Sequence

- **Pre-configuration** of LBISTCTRL-registers will be done automatically during system boot (configuration parameters stored in Flash).
- LBIST-execution can be **triggered automatically during boot** (if BMI.LBISTENA=’1’) or from software at a later time.
- LBIST affects whole SoC design and will automatically reset the device and restart boot process.
- **Evaluation of LBIST-results** (i.e. signature) is done **by software** (comparison with golden signature from Flash).
- Software can now always reset LBIST-done flag and trigger further LBIST-sequence w/o power-on-reset.

LBIST Changes

[√] Controls Dual Point Faults Latent
[√] Structural test of mission logic



LBIST Changes

- Usage as a Safety Mechanism
- Register interface via SCU
- Support latent fault detection coverage for Safety Mechanisms: >90%

Part of your life. Part of tomorrow.

Infineon