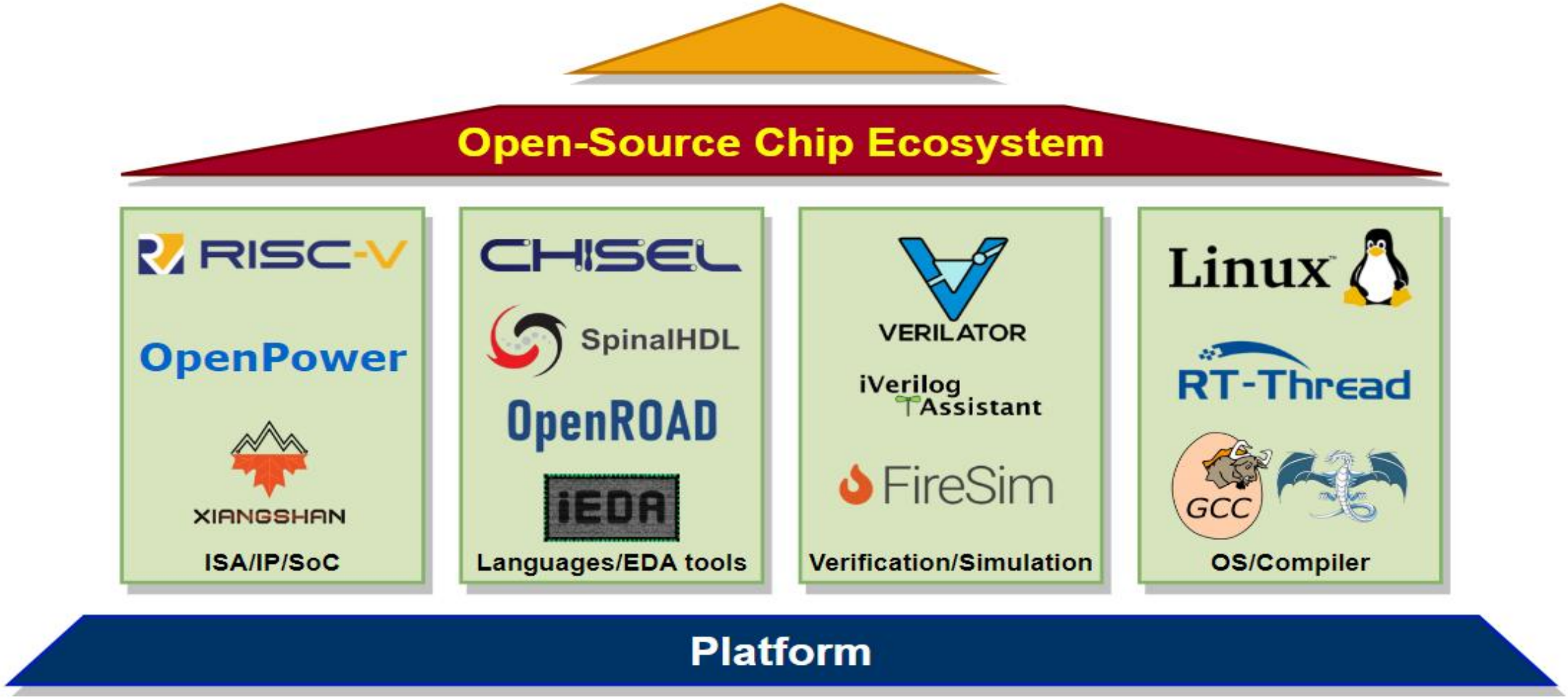
Structured DFT Development Approach for Chisel-Based High Performance RISC-V Processors

Bin Zhang, Ye Cai*, Zhiheng He, Sen Liang, Wei He

Shenzhen University
Beijing Institute of open source chip
Institute of Computing Chinese Academy of Sciences
PengCheng Laboratory

The Era of Agile and Open-Source Hardware Design



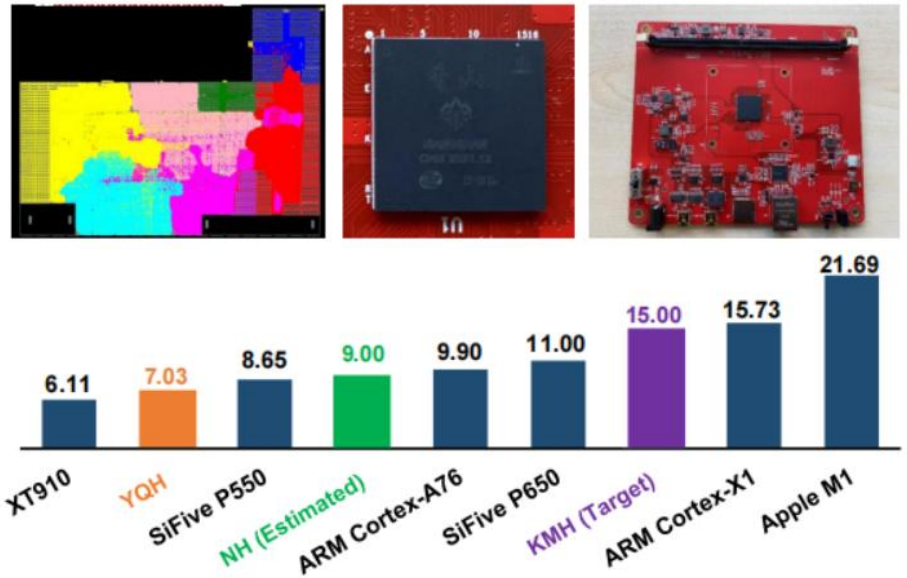
- Dr Yungang Bao’ s keynote at RISC-V summit Europe 2023

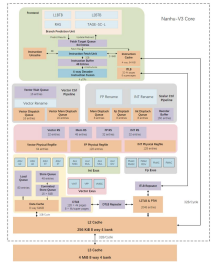
Xiangshan： Open-Source High Performance Processors

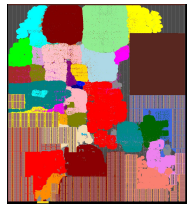
- **1st generation: YQH(Chisel-Based)**
  - RV64GC, single-core, superscalar OoO
  - 28nm tape-out, 1.3GHz, July 2021
  - SPEC CPU2006 7.01@1GHz, DDR4-1600

- **2nd generation: NH(Chisel-Based)**
  - RV64GCBK, dual-core, superscalar OoO
  - Scheduled 2GHz@14nm tape-out, Q4 2023
  - Estimated** SPEC CPU2006 19.45@2GHz

- **3rd generation: KMH(Chisel-Based)**
  - RV64GCBKHV, quad-core, superscalar OoO
  - Close collaboration with industrial partners



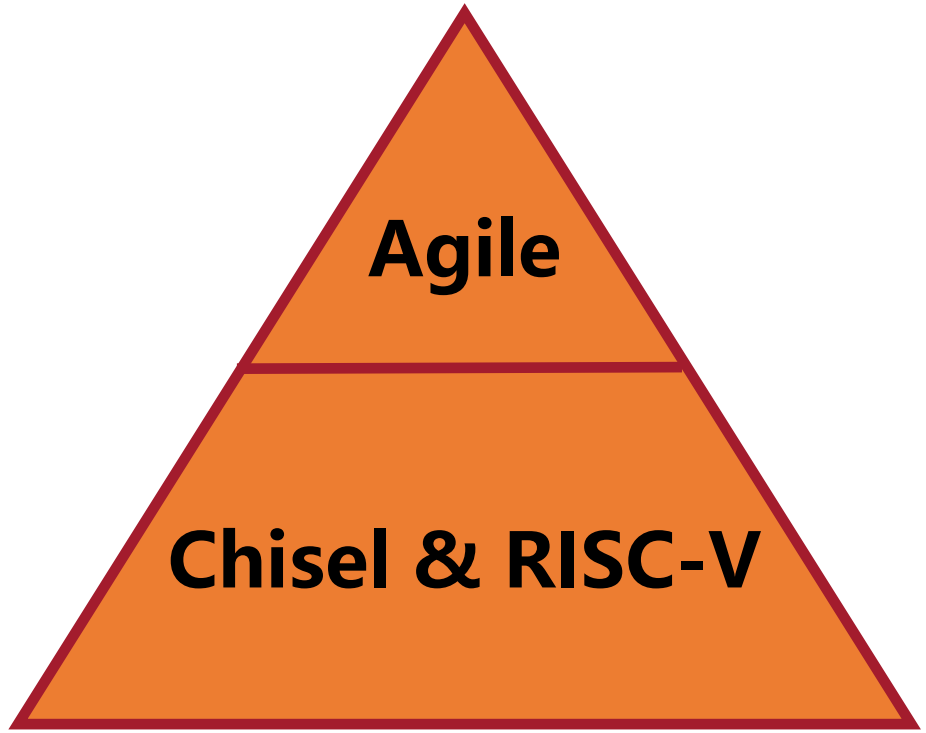




Chisel,Agile Development and DFT

Chisel’ s key feaures performs the advantages of agile development :

- Simplifies the complexity of signal connections
- Supports metaprogramming
- Use the object-oriented nature of high-level languages
- Supports functional programming



But, We meet some challenges:

- Structured DFT focus on RTL or netlist in the industry
  But not in Chisel
- Design and technology scaling may lead to high costs of outdated DFT

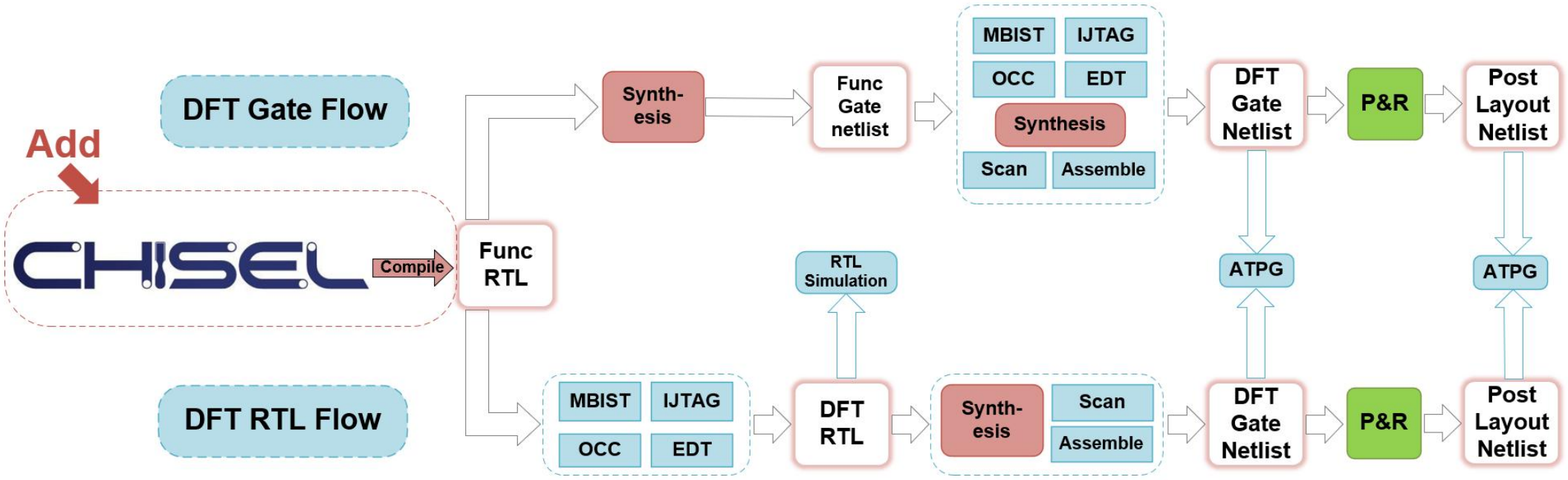
DFT Industry Practices For High Performance Processors

- ARM proposed Shared bus MBIST
- Tessent proposed the shift left DFT
- March Algorithm of MBIST is updating

Can we keep these advanced DFT techniques in Chisel-based design?

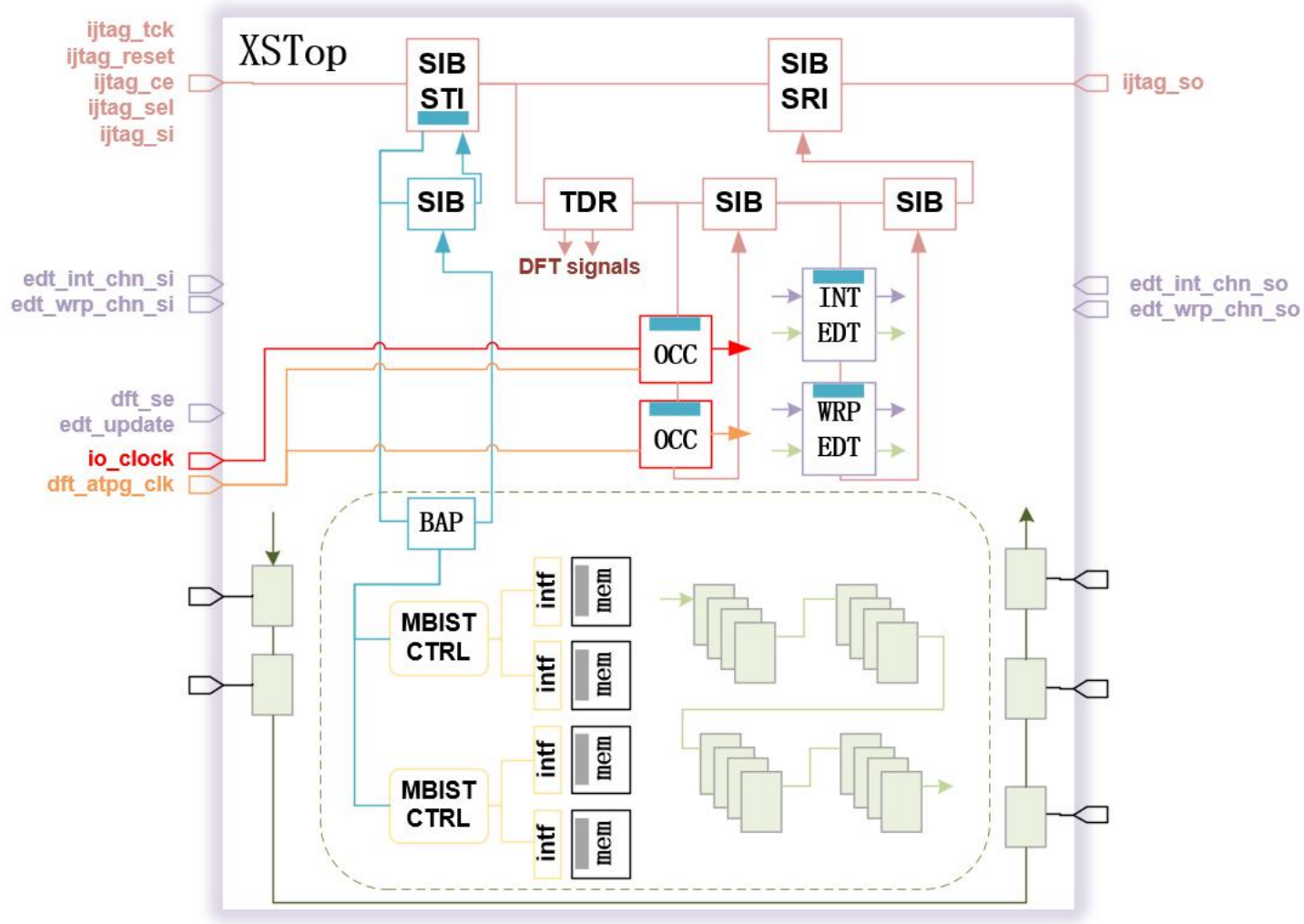
Proposed Chisel-Based DFT Design Flow in YQH

- **Reuse traditional DFT toolchain ecosystem**
- **DFT RTL flow and DFT Gate flow supported**



DFT Design Overall Program for YQH

- IJTAG Network
- OCC
- Full Scan Chain With Wrapper Chain
- Internal and Wrapper EDT
- Retargetable Pattern Supported
- 17 MBIST Controller

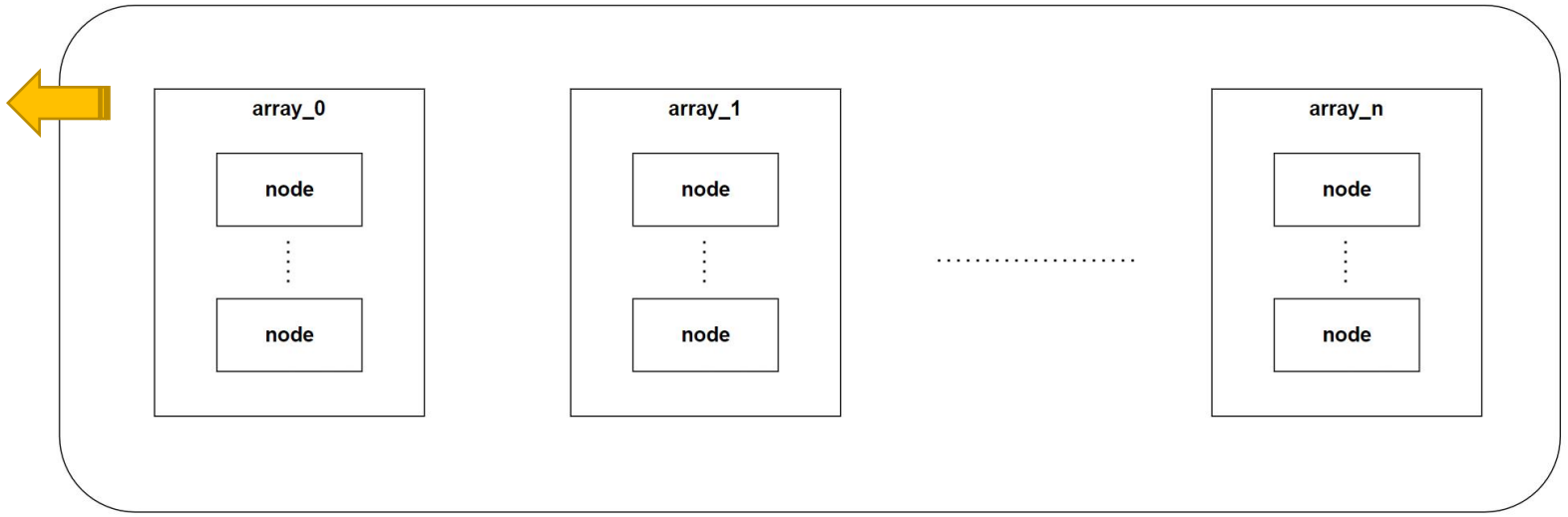


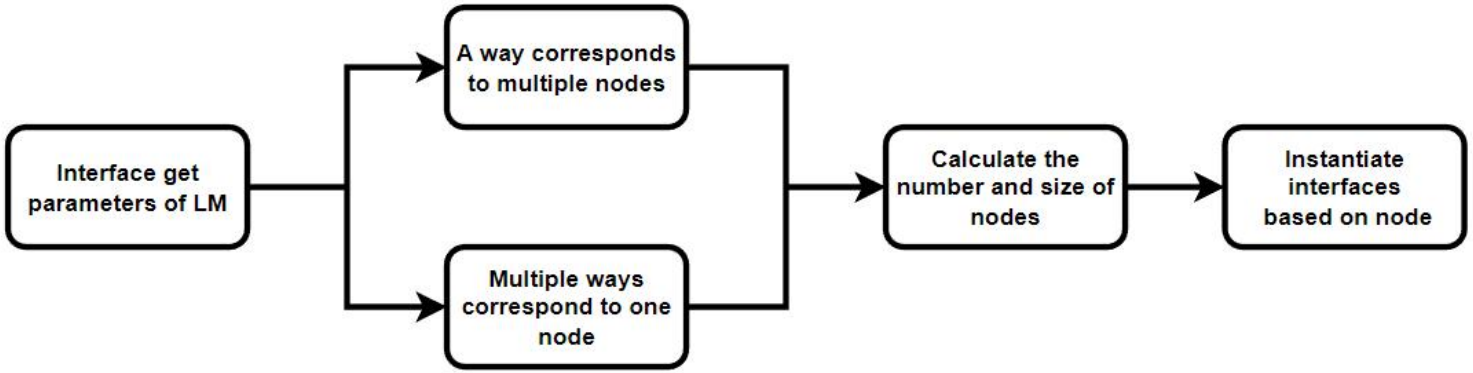
XS-Shared Bus MBIST in NH —— Adaptive Size Adjustment

We can determine the specification of the array(LM) with the parameter

Mbist_array in XS-shared bus is automatically scheduled

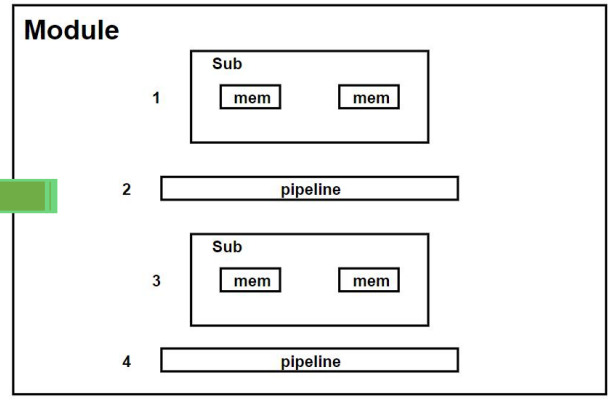
XS-SharedBus circuit can be self-adapted

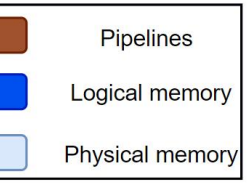




XS-Shared Bus MBIST in NH —— Flexible Pipeline

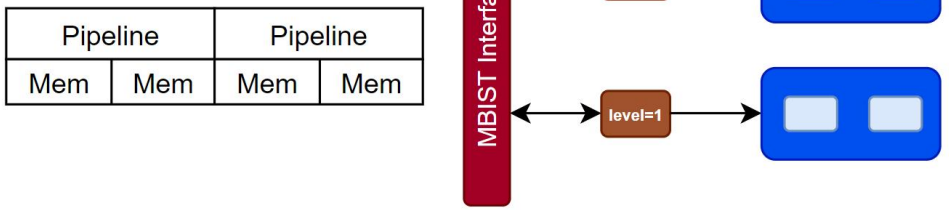
We can adjust the order in which modules are instantiated



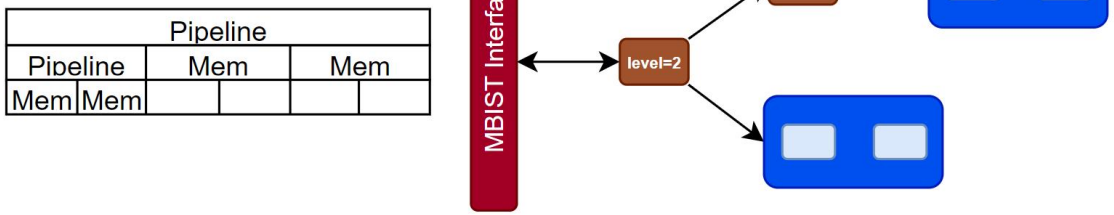


Realization of different topologies for pipeline

①



②

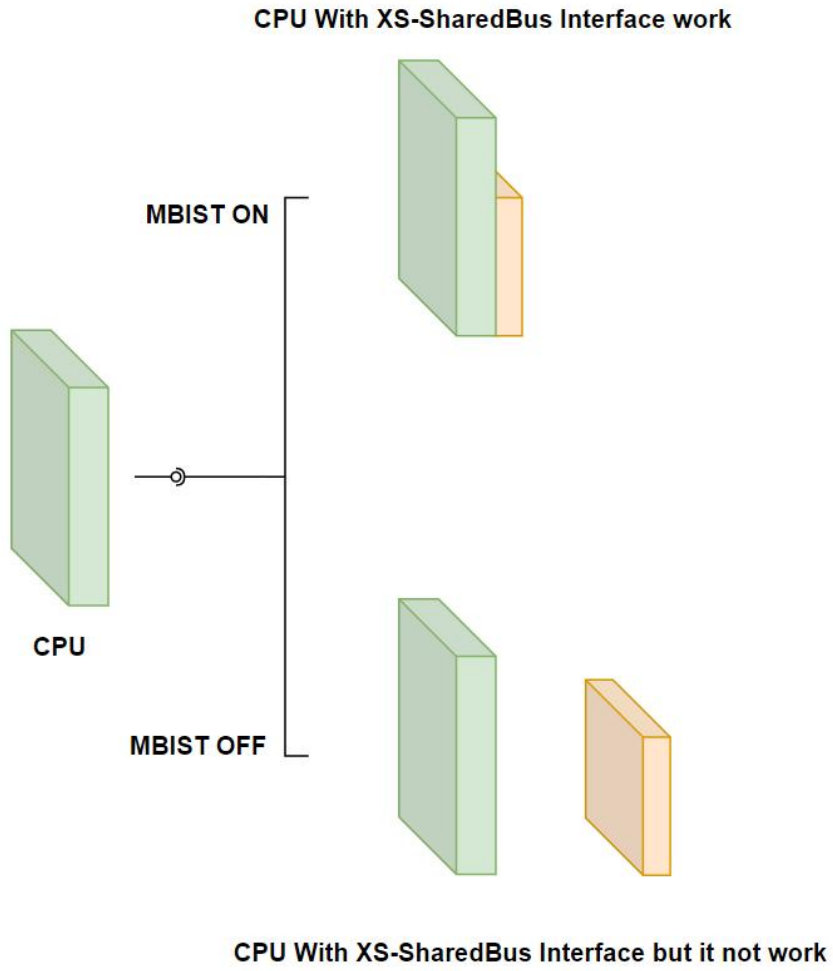


XS-Shared Bus MBIST in NH —— Plug and Play

With Chisel we can instantiate it directly wherever we want

No additional wiring operations are required

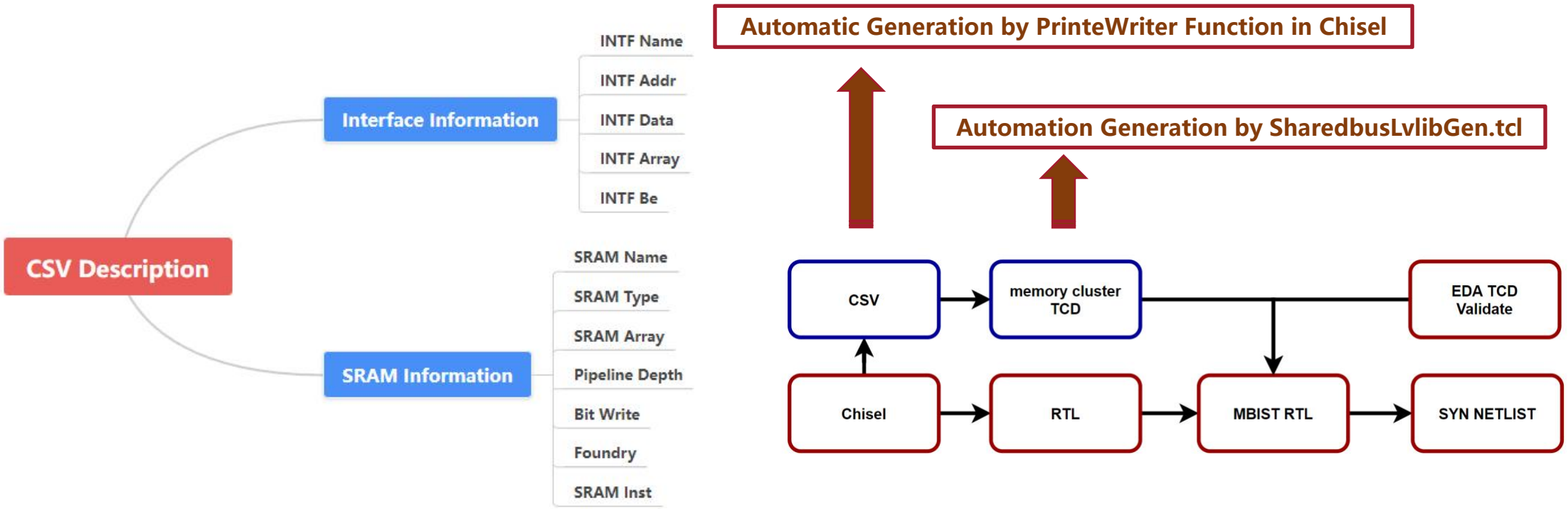
Parameter control Whether to enable



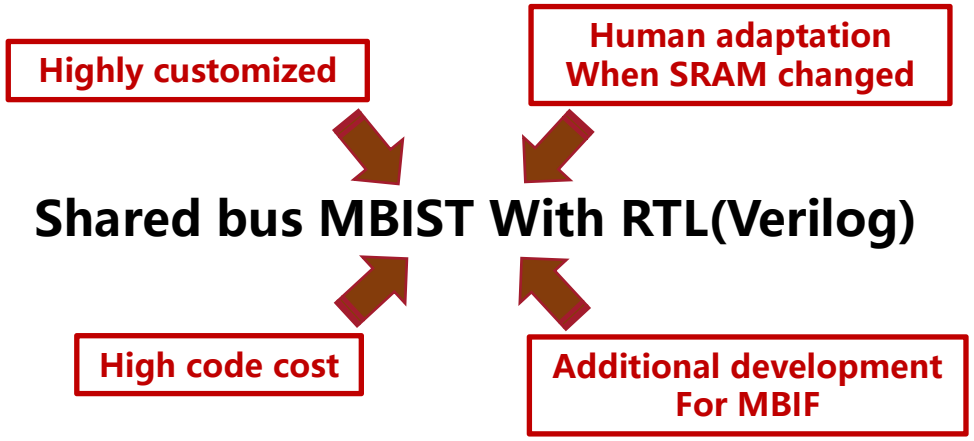
XS-SharedBus MBIST Flow Automation

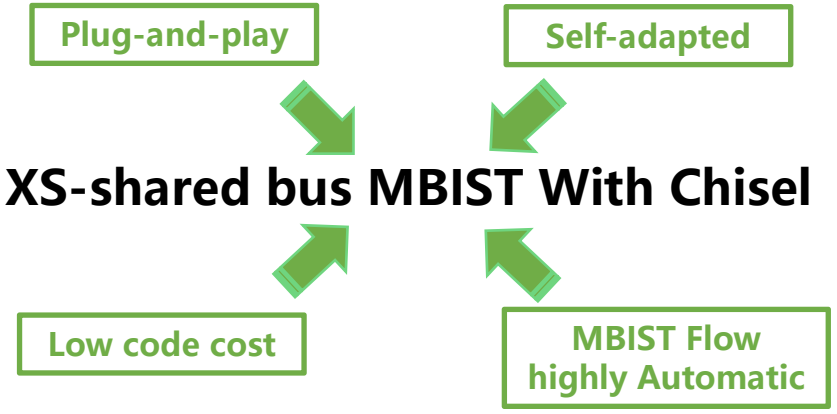
EDA tools need to know

how the mapping relationship between sharedbus and sram is formed!



Proposed Flexible Chisel-Based XS-SharedBus Interface in NH





Some Research on SRAM Dynamic Faults

The Faults Types Researched in Our Work	
Fault Type	**FFM**
Simple Static Fault	SAF,SF,TF,WDF,RDF,DRDF,IRF CFst,CFds,CFtr,CFwd,CFrd,CFdrd,CFir
Connection Fault	LF1(single<->single) LF2(single<->couple) LF3(single<->single)
Dynamic Fault	dRDF,dDRDF,dIRF,**dWDF**,dCFrd,dCFdrd,dCFir, **dCFwd**, **dCFds**

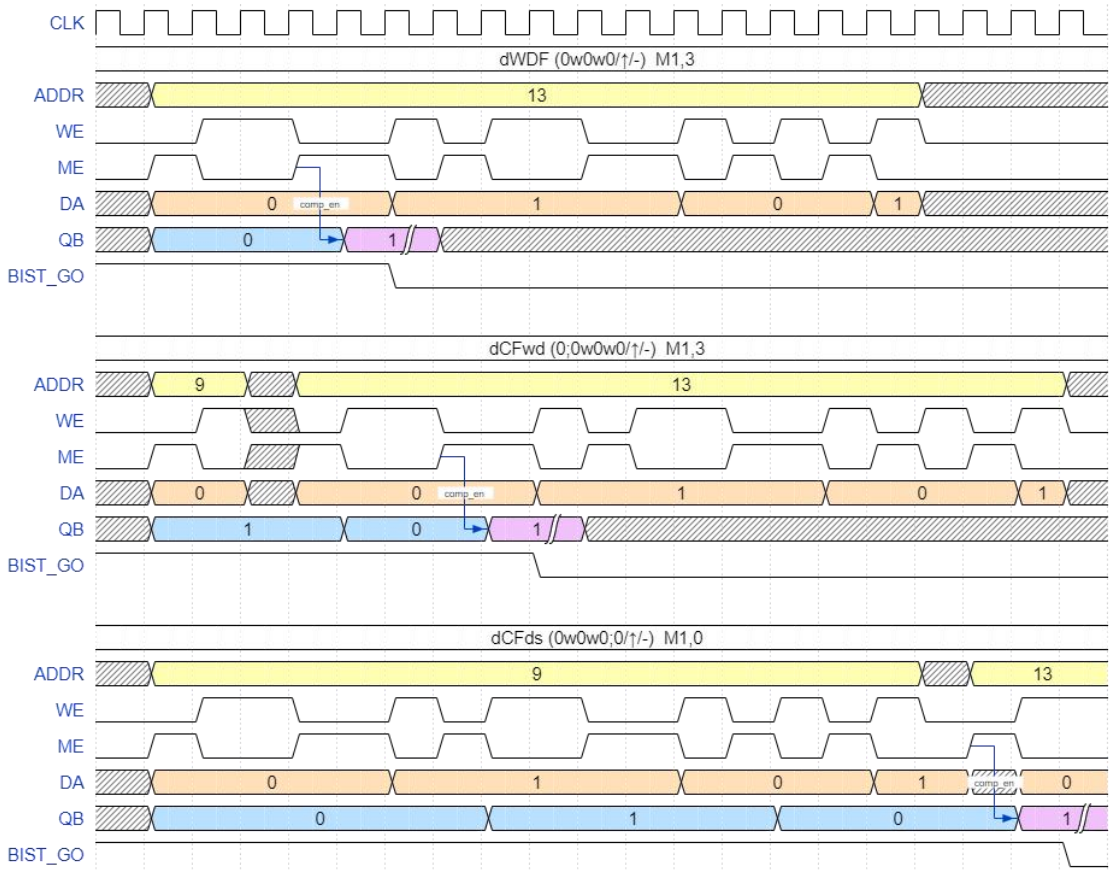
Fault Types that Existing Algorithms Cannot Cover	
FFM	**FP**
dWDF	0w0w0/↑/-, 1w1w1/↓/-
dCFwd	0;0w0w0/↑/-, 1;0w0w0/↑/-, 0;1w1w1/↓/-, 1;1w1w1/↓/-
dCFds	0w0w0;0/↑/-, 1w1w1;0/↑/-, 0w0w0;1/↓/-, 1w1w1;1/↓/-

Proposed March SLD Algorithm

{ ⇳(w0);
M0;
⇑(r0,w0,w0,r0,r0,w1,r1,w1,w1,r1,r1,w0,r0,w0,r0,w1);
M1;
⇑(r1,w1,w1,r1,r1,w0,r0,w0,w0,r0,r0,w1,r1,w1,r1,w0);
M2;
⇓(r0,w0,w0,r0,r0,w1,r1,w1,w1,r1,r1,w0,r0,w0,r0,w1);
M3;
⇓(r1,w1,w1,r1,r1,w0,r0,w0,w0,r0,r0,w1,r1,w1,r1,w0); }
M4;

dWDF and dCFwd faults in red font
dCFds fault in blue font

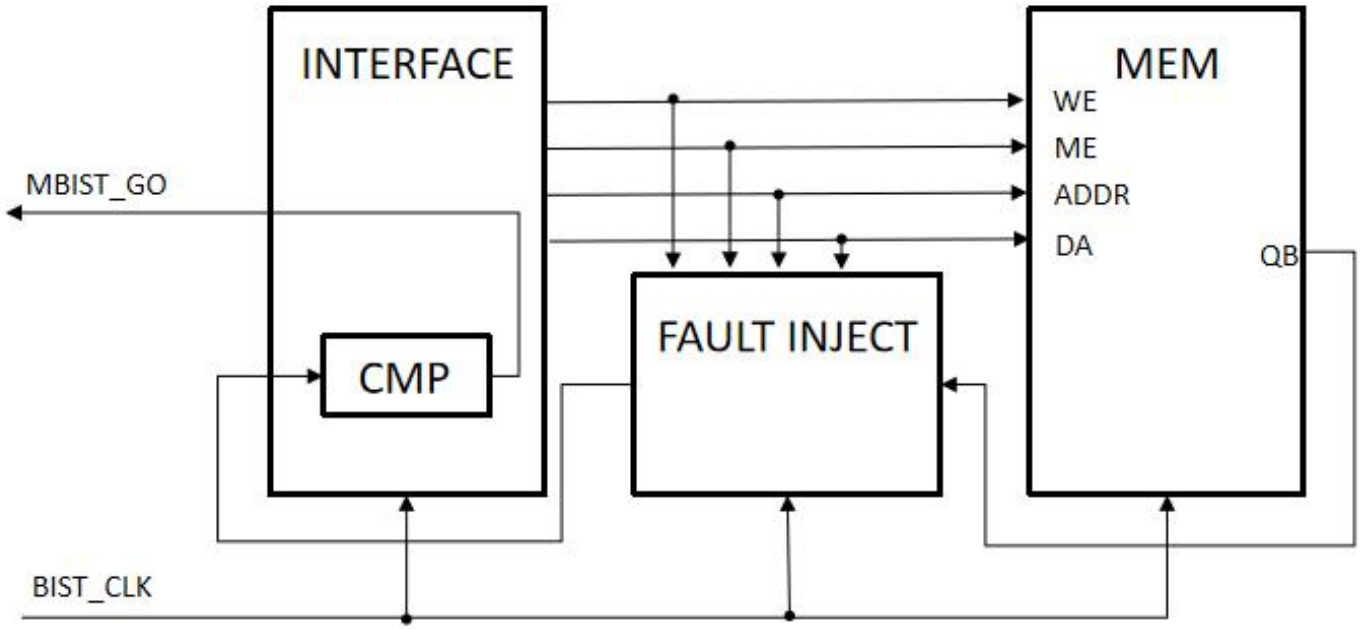
FFM	FP	Phase
dWDF	0w0w0/↑/-	M1,3 M2,9 M3,3 M4,9
	1w1w1/↓/-	M1,9 M2,3 M3,9 M4,3
dCFwd	0;0w0w0/↑/-	M1,3 M2,9 M3,3 M4,9
	1;0w0w0/↑/-	M1,3 M2,9 M3,3 M4,9
	0;1w1w1/↓/-	M1,9 M2,3 M3,9 M4,3
	1;1w1w1/↓/-	M1,9 M2,3 M3,9 M4,3
dCFds	0w0w0;0/↑/-	M1,0 M3,0
	0w0w0;1/↓/-	M2,0 M4,0
	1w1w1;0/↑/-	M1,0 M3,0
	1w1w1;1/↓/-	M2,0 M4,0



Fault Simulation Module

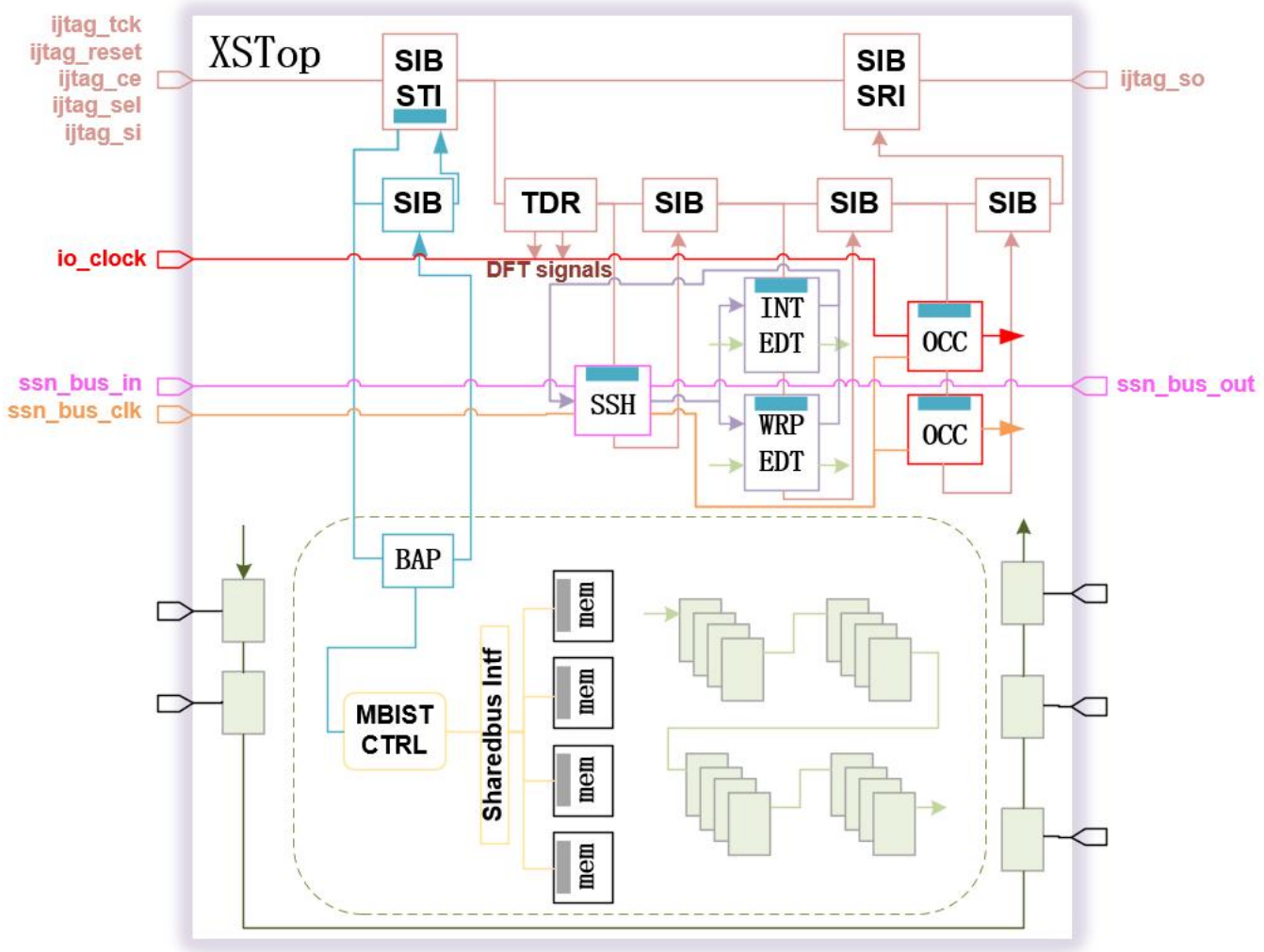
Portable Algorithm Verification Module

- **generic and portable fault simulation**
- **get signals synchronously**
- **processe value when fault trigger**

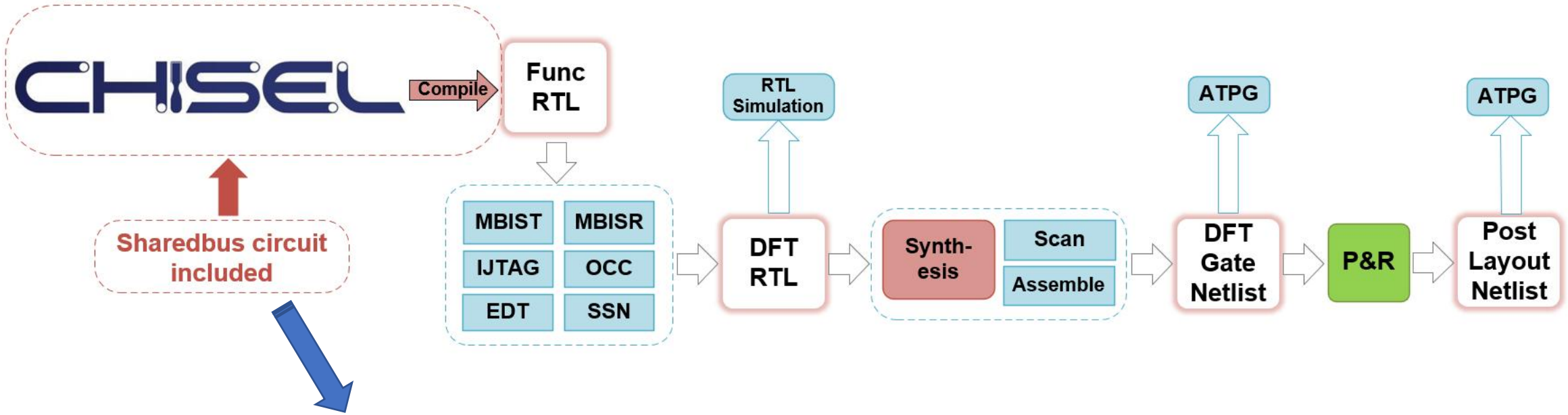


Hierarchical DFT Intergrated Optimization in NH

- **The Integration of Streaming Scan Network(SSN)**
- **Add MBIST support for SRAM with XS-shared bus scheme**
- **MBISR for Memory Repair**



Chisel-Based DFT Design Flow Optimization in NH



More and more DFT IPs will be developed in Chisel further
We proposed DFT shift-left more and more!

DFT Practice in the YQH and NH

YanQiHu, 1.3GHz@28nm	Dual Core NanHu, 2GHz@14nm
17 MBIST Controller	**8 MBIST Controller**
MBIST algorithm complex 44N	MBIST algorithm complex 68N **Extra March SLD Hard-Coded** L3 Cache MBISR
MBIST test time 4.37ms	**MBIST test time 17.86ms**
3.86M Instances, 508k ScanReg	**7.9M Instances, 1.59M ScanReg total**
Scan pin Total 25 Global Signal:3 Int/Wrp edt channel 10:10/1:1	Scan pin Total 33 Ssn_bus_clk:1 Ssn_bus_in/out 16:16
Scan shift@48MHz	**Scan shift@100MHz** **ssn_bus_clk@200MHz**
Stuck-at cov 99.92% Transition cov 98.20%	Stuck-at cov 99.89% Transition cov 97.86%
SA Pattern count 22660 Tr Pattern count 41313	SA Pattern count 24119 Tr Pattern count 45559
SA+TR test time 434.41ms	**SA+TR test time 281.4ms**
MBIST+SA+TR test time 438.78ms	**MBIST+SA+TR Test time 299.26ms**

53% ↓

Memory serial testing leads to increased testing time in Sharedbus

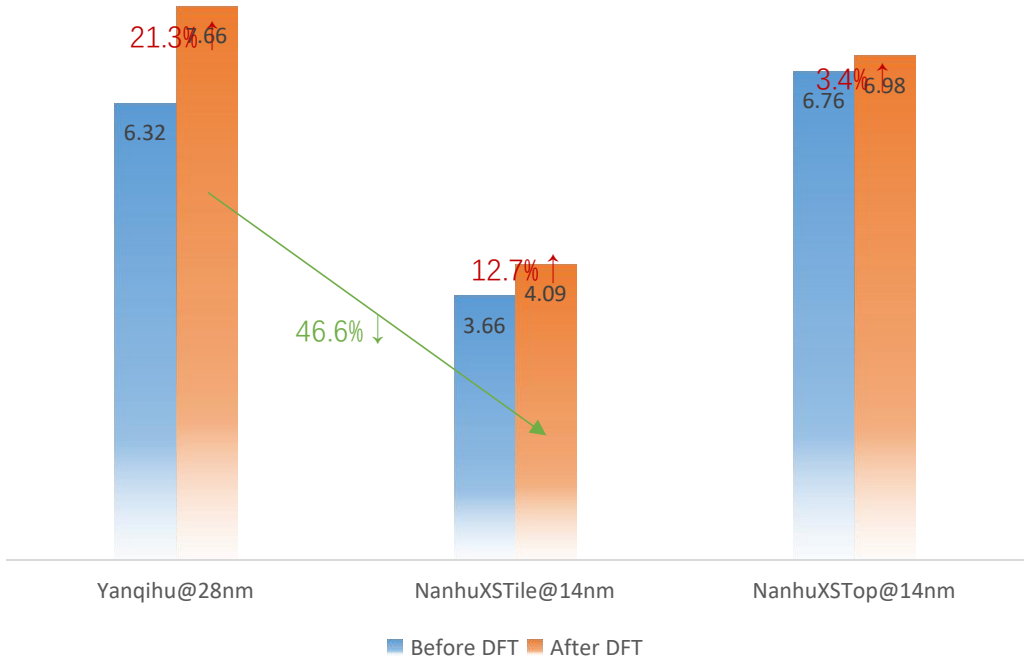
Instances 105% ↑
RegCount 213% ↑

NH optimizes ATPG test time by increasing shift freq. and upgrading SCAN integration architecture to SSN

108% speed up

NH test time is only 68.2% of YQH. Although instances of NH is more than 2x of YQH

Comparison of the DFT area growth of YQH and NH



- **DFT area growth optimized from 21.3% to 12.7%**
- **The overall area is reduced by 46.6% cause of 14nm process node**
- **XSTop DFT area growth ratio is only 3.4% in huge cache size**
- **Sharedbus is more suitable for dealing with these scenarios to reduce mbist controllers and friendly place and route.**

Remarks:

- YQH and NH XSTile have similar structure and Instances size
- SharedBus logic is not included in NH area data before DFT
- IJTAG, MBIST, OCC, EDT, SCAN replacement, SCAN stitching is included after DFT
- In addition, SSN and MBISR is include in NH area data after DFT

March SLD Practice in the NH

Algrithm	Coverage	Complexity	Area	Power
March SL	79.14%	41N	14723814	2.35E+03
March SLE	97%	49N	14724015	2.35E+03
March SLD	**100%**	**65N**	**14724430**	**2.35E+03**

- **The coverage is only counted for the types of faults mentioned in this article**

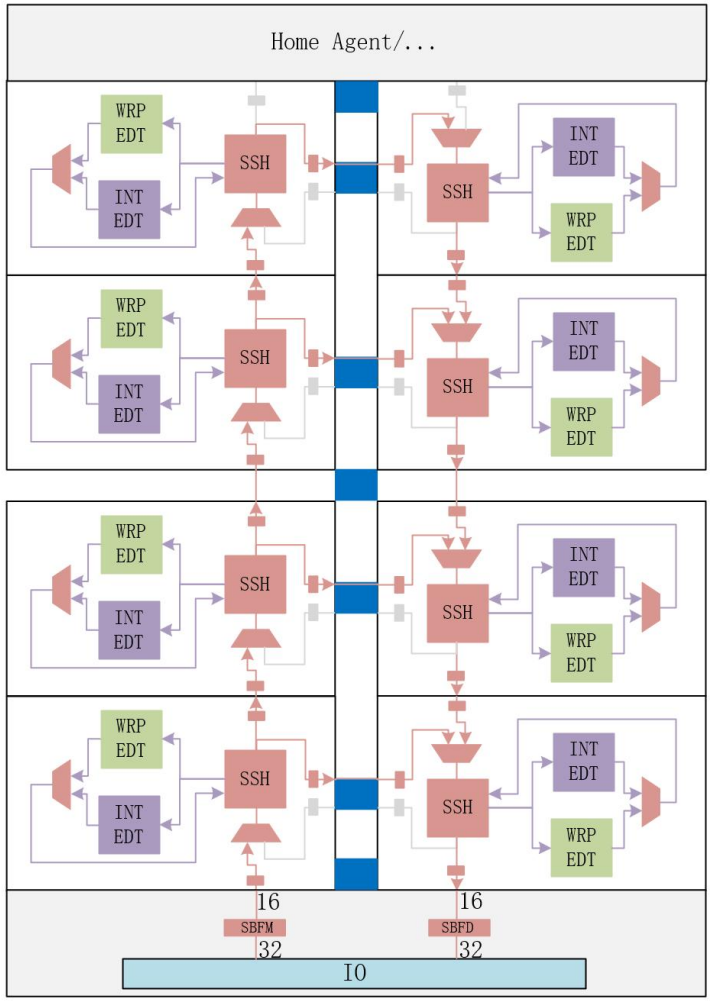
SCAN Integration for many-core XiangShan processors

SSN realizes less signal interaction between design modules,
serial connection between design modules is more suitable for tile-based design

	TEST Group1	TEST Group1
SSN ATPG TEST	Retargetable Pattern: XSTop(I)+XSTile(E) Retargeting Group: CLUSTER0-CLUSTER7	Retargetable Pattern: XSTile(I) Retargeting Group: CLUSTER0-CLUSTER7

	SA_GP1	TR_GP1	SA_GP1	TR_GP2	TOTAL
SSN	22.1ms	62.0ms	77.2ms	122.1ms	283.4ms

On-chip compare makes constant test costs for identical cores design



SSN Integration

Conclusion

- Proposed Chisel-based DFT design flow reused the traditional DFT toolchain ecosystem
- Proposed flexible Chisel-Based XS-SharedBus interface is plug-and-play，self-adapted and effectively improving design PPA
- Proposed MarchSLD algorithm improves the coverage of SRAM dynamic fault detetion to 100%
- Facing the design scaling ,proposed optimized Chisel-based DFT design flow can support agile development requirements and achieve industry-competitive performance

THANKS !

Contact us via zhangbin@bosc.ac.cn

Bin Zhang, Ye Cai*, Zhiheng He, Sen Liang, Wei He

Shenzhen University
Beijing Institute of open source chip
Institute of Computing Chinese Academy of Sciences
PengCheng Laboratory