2024
DESIGN AND VERIFICATION™
DVCON
CONFERENCE AND EXHIBITION
UNITED STATES
SAN JOSE, CA, USA
MARCH 4-7, 2024

Tackling Missing Bins: Refining Functional Coverage in SystemVerilog for Deterministic Coverage Closure

Jikjoo Lee, Tony Gladvin George, Kihyun Park,
Dongkun An, Wooseong Cheong, ByungChul Yood
Memory Division, Samsung Electronics

SAMSUNG

accellera
SYSTEMS INITIATIVE

Table of Contents

Motivation

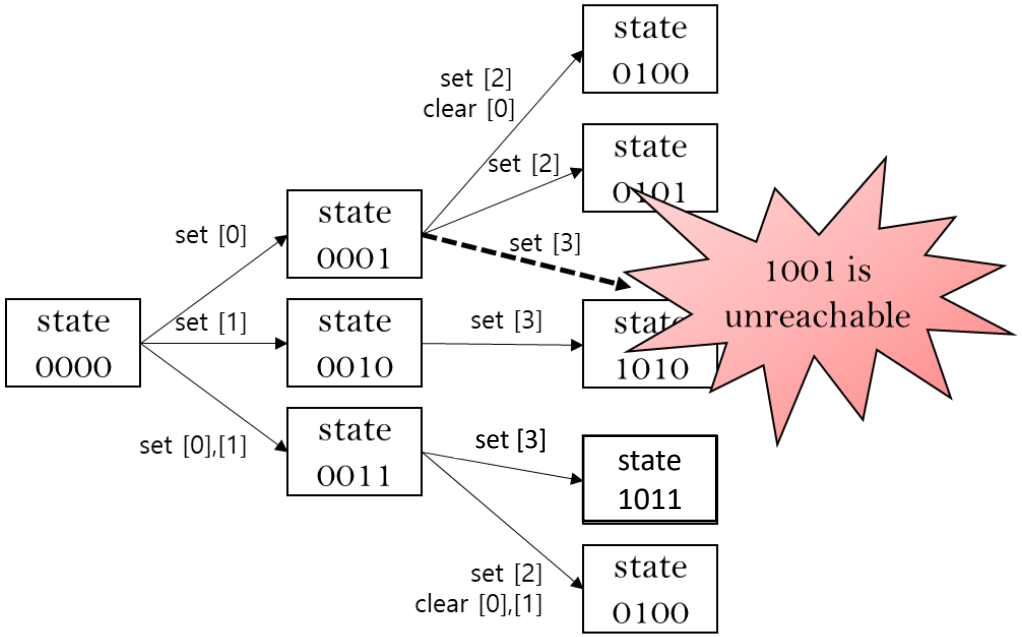
- Verification of state machines with a large number of states
- Configurable state machines that keep changing
- Test regression 100% pass, but still bugs exist
- Is verification truly complete?

Exhaustive Coverage Closure

- Define all possible cases
- Test and check all cases
- Functional coverage in SystemVerilog

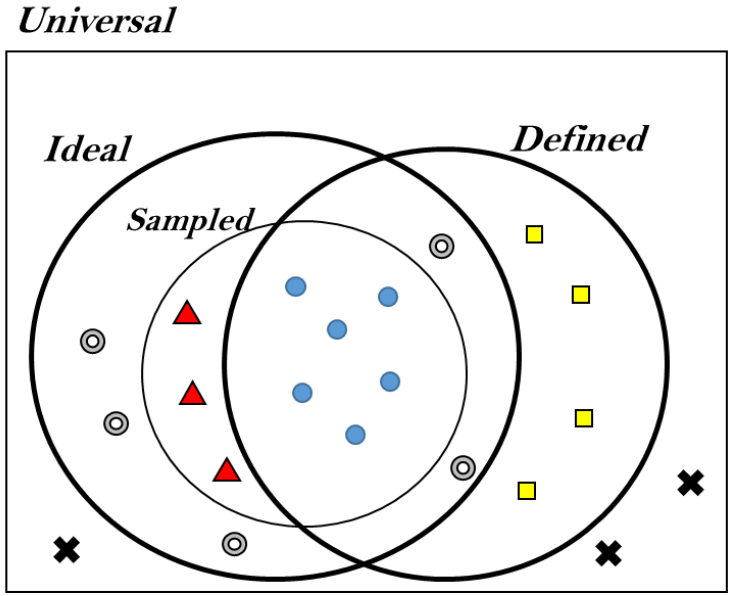
The Challenge of Defining Coverage

- Human errors in defining coverage
  - The next state depends on the previous state
  - Non-deterministic inputs
- 4-bit state transformed by specific rules
  - 9 / 16 cases can be possible
  - Inclusion of unreachable states by mistake
  - Potential omission of reachable states
- Exponentially increasing cases
  - more human errors



Negative Impacts of Missing Bins

- Missing bins: sampled in tests, but not defined in coverage
  - Falsely indicate comprehensive test
  - Potential Bugs in RTL



	coverage bin type	meaning
●	covered bins	sampled and defined
✖	excluded bins	excluded by ideal and defined both
◎	uncovered bins	ideal but not sampled
■	missed exclude bins	inadvertently included → need to waive
▲	missing bins	sampled but not defined → need to include

Traditional methodology

- Illegal_bins
  - Does not provide information about when the excluded bins were hit
  - Cannot control report

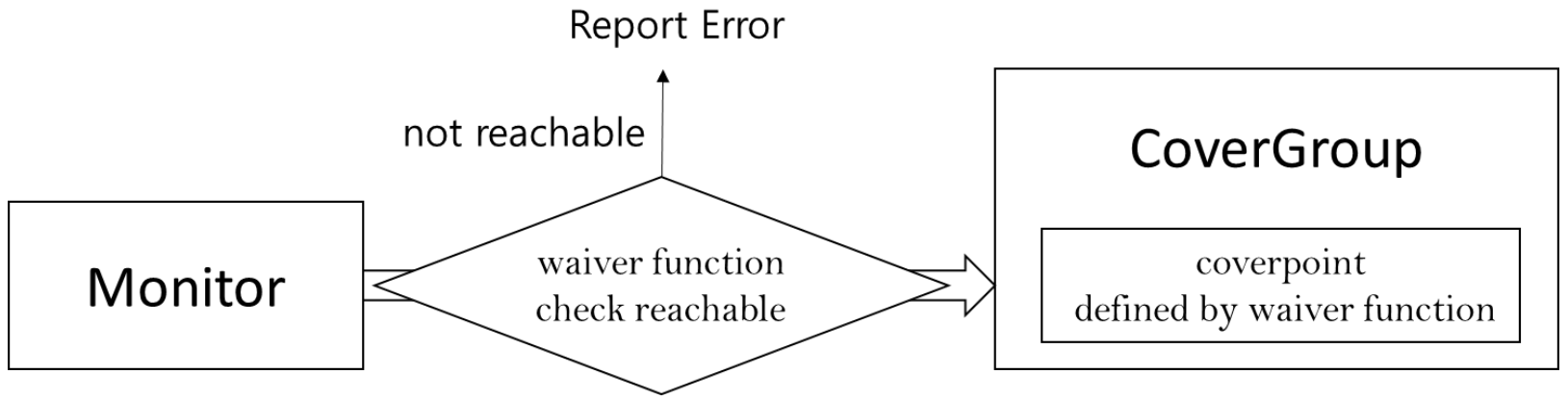
- Functional Coverage Management System, DVCon 2015
  - Generates SVA model and coverage model
  - Requires a separate tool, SpecGen
  - Performance drop 30%

Proposed Solution

- Writing functional coverage skill that detects missing bins
- No other tools required
- Provides detailed log information when missing bins occur
- Performance degradation within 3%

Three-step Approach to Remove Missing Bins

1. Defining coverage bins using the waiver function
2. Defining cross-coverage using CrossQueueType
3. Reporting missing bins before sampling



Step 1. Defining coverage bins using the waiver function

- Waiver Function
  - determines if the data is reachable
  - reusability across multiple scopes
  - if state is 0111, output 2 as illegal_idx
- Only valid data in the queue
- Flexible and reactive in cross-coverage

```
bit [10-1:0] state_cover_bins[$];
function int WaiveState(int state);
  int illegal_idx = 0;
  casez (state)
    'b?1???0??? : illegal_idx = 1
    'b??????11?? : illegal_idx = 2    ;
    'b??????1?1? : illegal_idx = 3    ;
     ...
  endcasez
   return illegal_idx ;
endfunction
...
for(int state=0; state < (1<<10) ; state++ ) begin
  if(WaiveState(state) == 0 ) continue;
  this.state_cover_bins.push_back(state);
end
```

Waiver Function

push only reachable bins

Step 2. Defining cross-coverage using CrossQueueType

- CrossQueueType
  - SystemVerilog keyword for cross-coverage bins

```
cross_state :cross s9,s8,s7,s6,s5,s4,s3,s2,s1,s0 {
   function CrossQueueType createIgnoreBins();
       for(int state=0; state < 1<<10; state++) begin
           if(state inside {this.state_cover_bins}) continue;
          else createIgnoreBins.push_back('{state[9], state[8], state[7], state[6], state[5],
state[4], state[3], state[2],state[1],state[0]});
      end
   endfunction

  ignore_bins ignore_cross = createIgnoreBins();
}
```

The queue defined in step 1

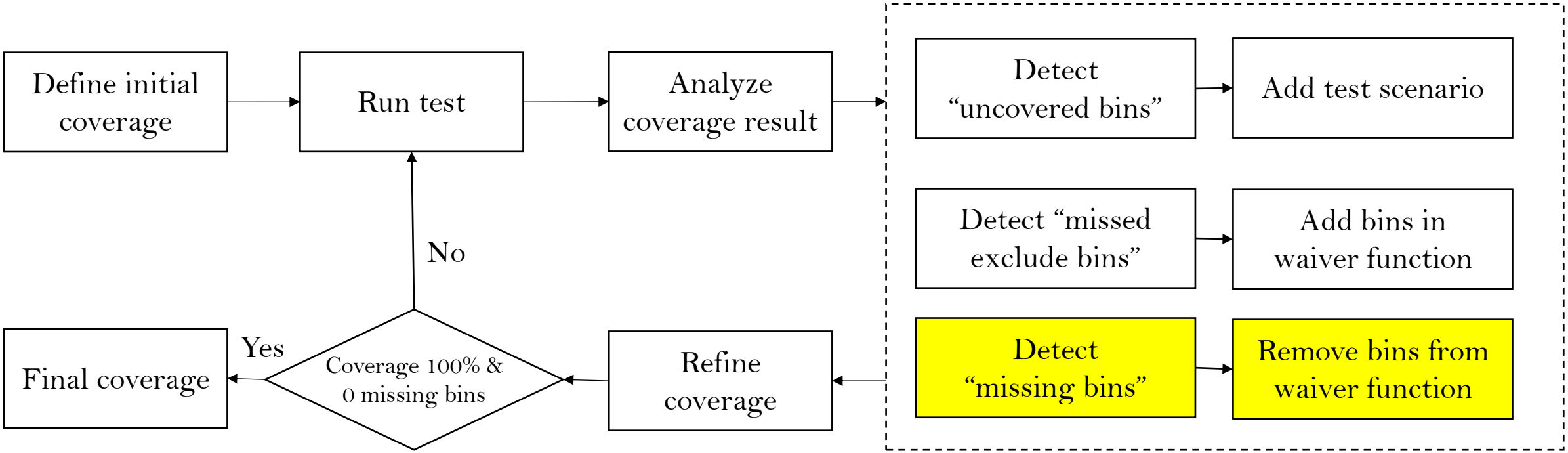
Step 3. Reporting Missing Bins Pre-Sampling

- The reporting method is configurable

```
illegal_idx = WaiveState(resp_state);
if(this.coverage_illegal_on && illegal_idx > 0) begin
  `ERROR($sformatf("Illegal RespState:%b Idx:%1d", resp_state, illegal_idx))
end
StateCovGrp.sample(resp_state);
```

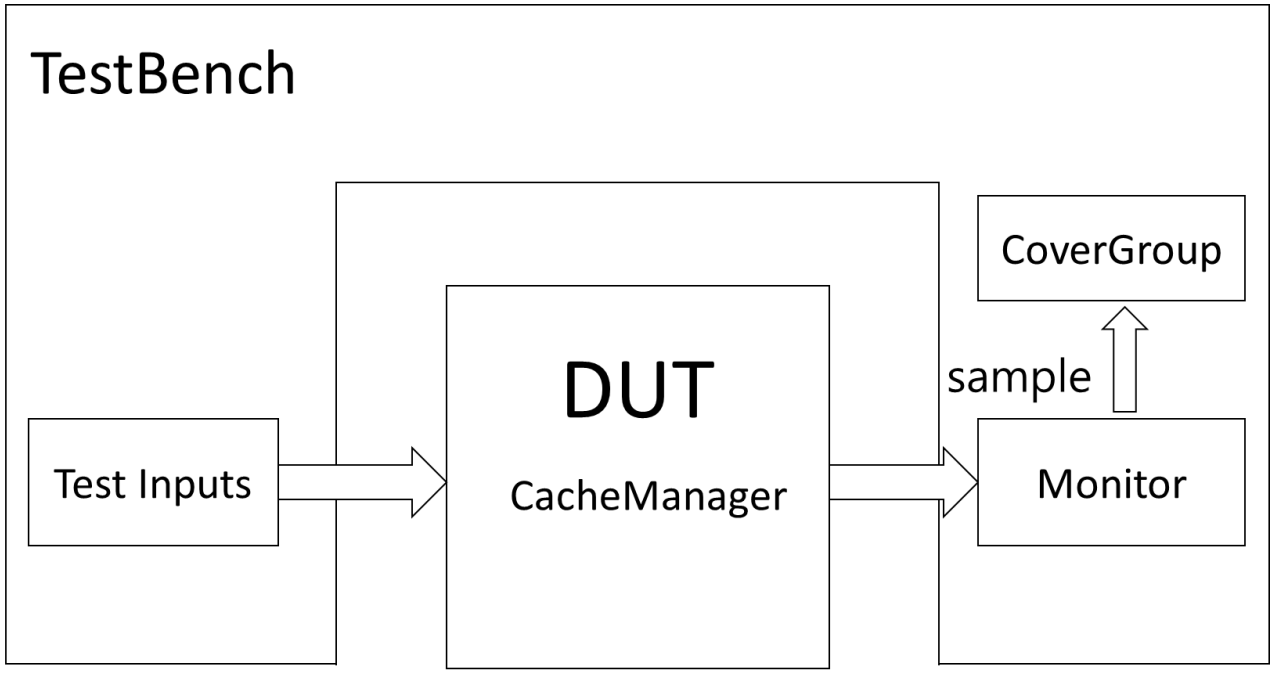
ERROR only when coverage_illegal_on is 1

Process of Refining Functional Coverage



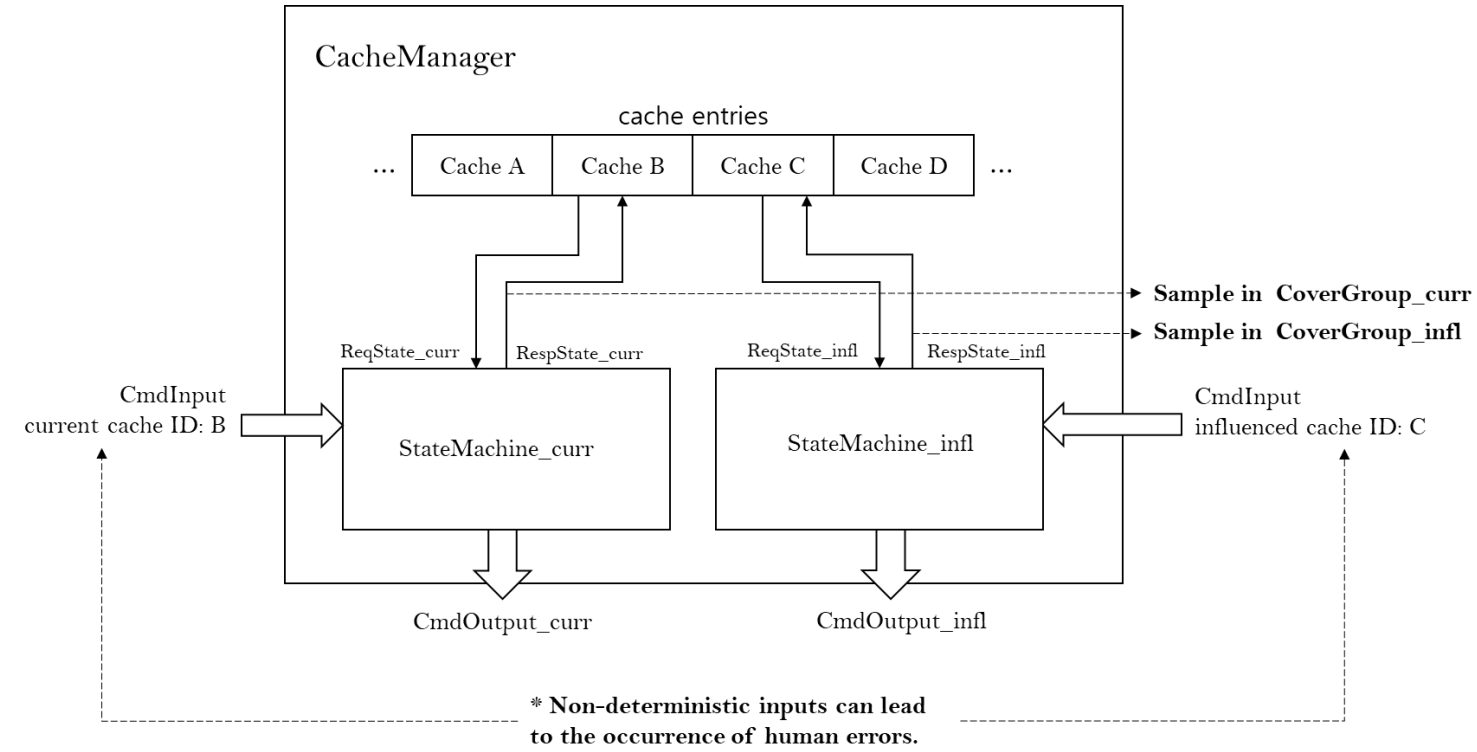
Experimental Setup

- Design Under Test
  - CacheManager
- Constrained random inputs
- Sampled data to CoverGroup



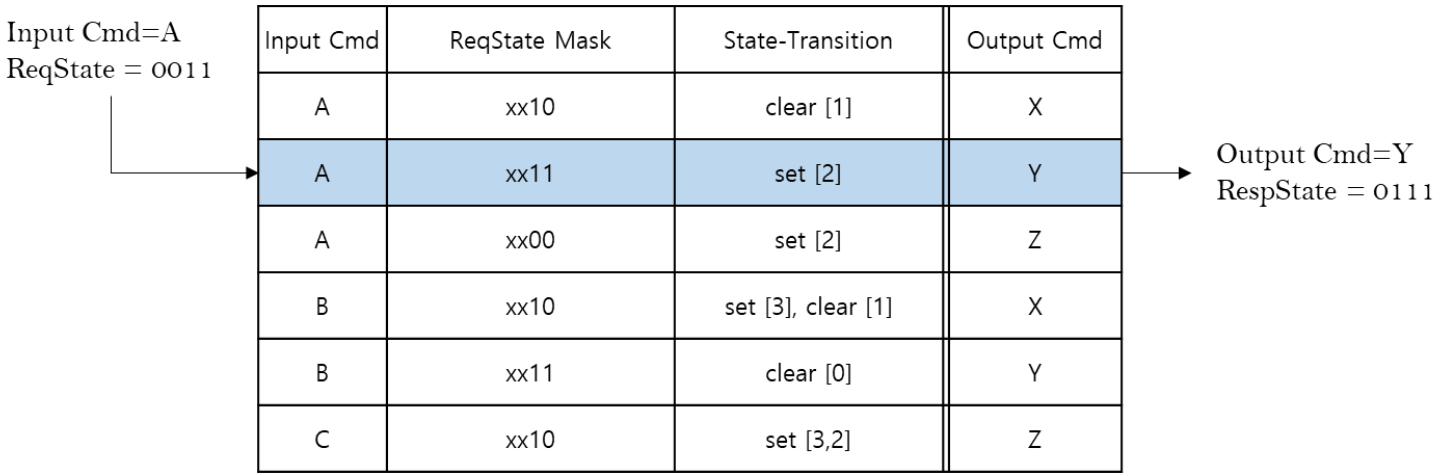
Command Flow of the CacheManager

- 10-bit state caches
- Current/Influenced StateMachines
- Random Command Inputs
- Samples Response States



How State Transition Works

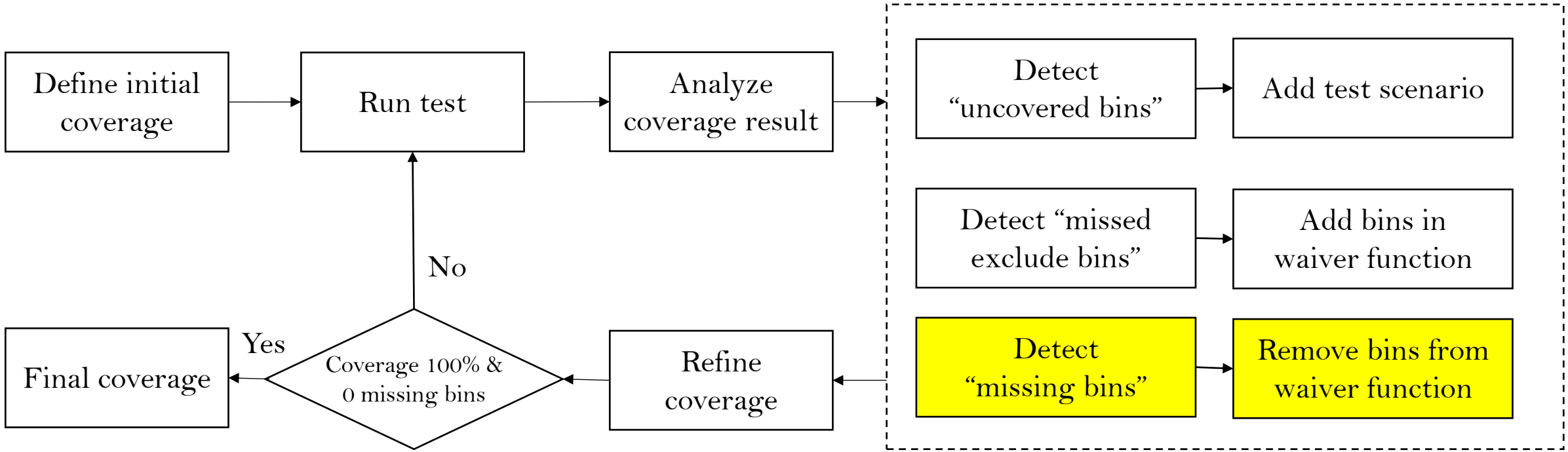
- Find a row that matches Input Command + Request State Mask
- Return Output Command + Response State



Input Cmd	ReqState Mask	State-Transition	Output Cmd
A	xx10	clear [1]	X
A	xx11	set [2]	Y
A	xx00	set [2]	Z
B	xx10	set [3], clear [1]	X
B	xx11	clear [0]	Y
C	xx10	set [3,2]	Z

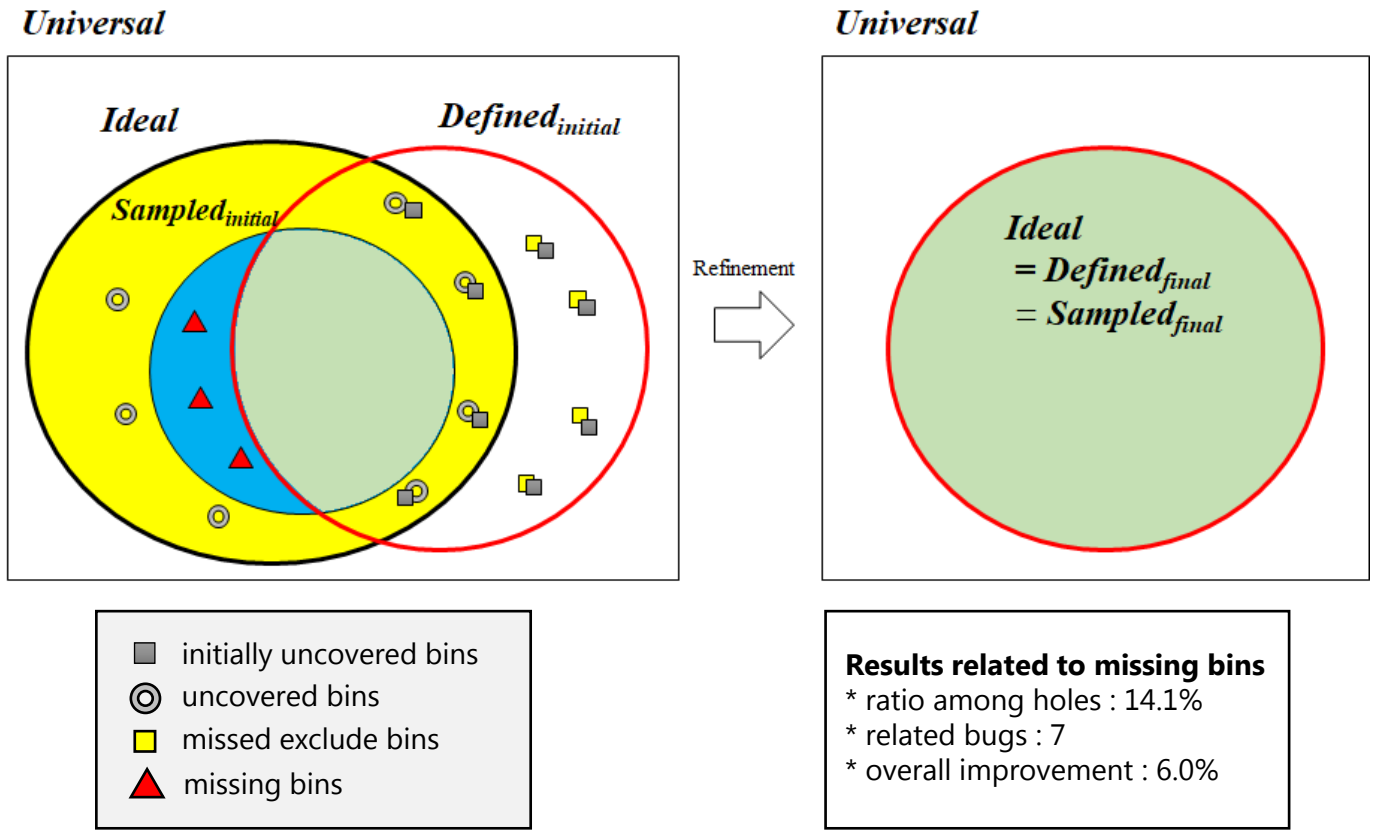
Experiment

- Process of Refining Functional Coverage



Experimental Result Summary

- Missing bins ratio: 14.1%
- Related bugs: 7
- Overall Improvement: 6%



Analysis of the Size of Coverage Set

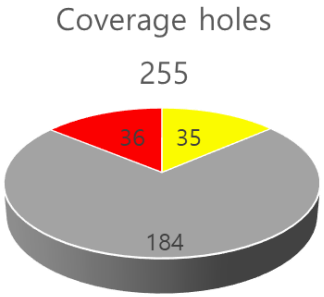
- Initially Defined Set: accuracy of 63.3%

the size of set	U	D_{init}	S_{init}	$D_{final}=I=S_{final}$	$D_{init} \cap D_{final}$
RespState_curr	1024	304	202	228	200
RespState_infl	1024	296	119	165	180
Total	2048	600	321	393	380

* D_{init} accuracy : 63.3%

Analysis of the coverage bin results

- Total coverage holes: 255
- Total missing bins: 36
- Missing bins: 14.1%



The number of coverage bins	Coverage holes	Initially uncovered	Uncovered	Missed exclude	Missing
RespState_curr	116	102	12	90	14
RespState_infl	139	117	23	94	22
Total	**255**	219	35	184	**36**

Conclusion

- How to effectively detect missing bins
  - Leveraging a waiver function
  - Using CrossQueueType
  - Reporting before sampling
- Methodology for thorough verification
  - improving accuracy and reliability
- Future Work
  - Adapt to transition bins
  - Reactive coverage closure through script automation

References

- [1] "what is difference between ignore bins and illegal bins.", Verificaton Academy, last modified July 29, 2021, accessed Sep 1, 2023, https://verificationacademy.com/forums/systemverilog/what-difference-between-ignore-bins-and-illegal-bins.
- [2] S.Ikram, J.Perveilier, I.Akkawi, J.Ellis, D.Asher, " Table-based Functional Coverage Management for SOC Protocols" in DVCon 2015, https://dvcon-proceedings.org/wp-content/uploads/table-based-functional-coverage-management-for-soc-protocols.pdf
- [3] Cadence Incisive vManager User Guide, https://support.cadence.com/apex/ProductManuals?pageName=ProductManuals
- [4] "How to Ignore Cross Coverage Bins Using Expressions in SystemVerilog", AMIQ Consulting, last modified September 17, 2014, accessed Sep 1, 2023, https://www.amiq.com/consulting/2014/09/17/how-to-ignore-cross-coverage-bins-using-expressions-insystemverilog/
- [5] IEEE Standard for SystemVerilog,19.6.1.3, https://ieeexplore.ieee.org/document/8299595/metrics#metrics

Questions