Whitebox Approach for Verifying PCIe Link Training and Status State Machine

Pinal Patel, Gaurang Chitroda
eInfochips Ltd
Colm McSweeney
Synopsys Ltd

September 23, 2014
SNUG Austin

Agenda

- Introduction
- LTSSM Testbench Architecture
- Rx PIPE Agent & Error Stimulus Generation
- Coverage & Debug Features
- Results & Summary

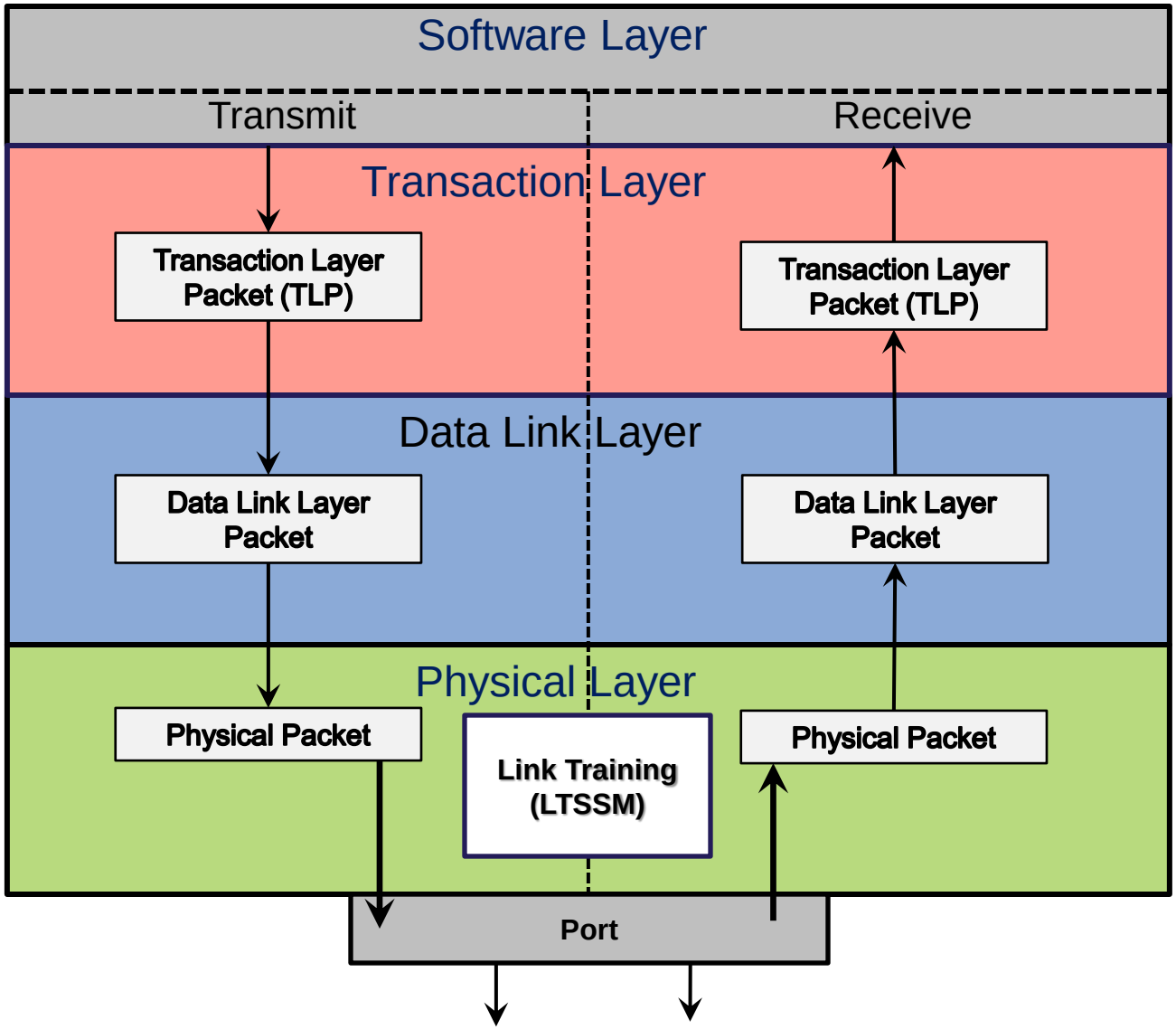
Introduction

- **Introduction**
- LTSSM Testbench Architecture
- Rx PIPE Agent & Error Stimulus Generation
- Coverage & Debug Features
- Results & Summary

Introduction

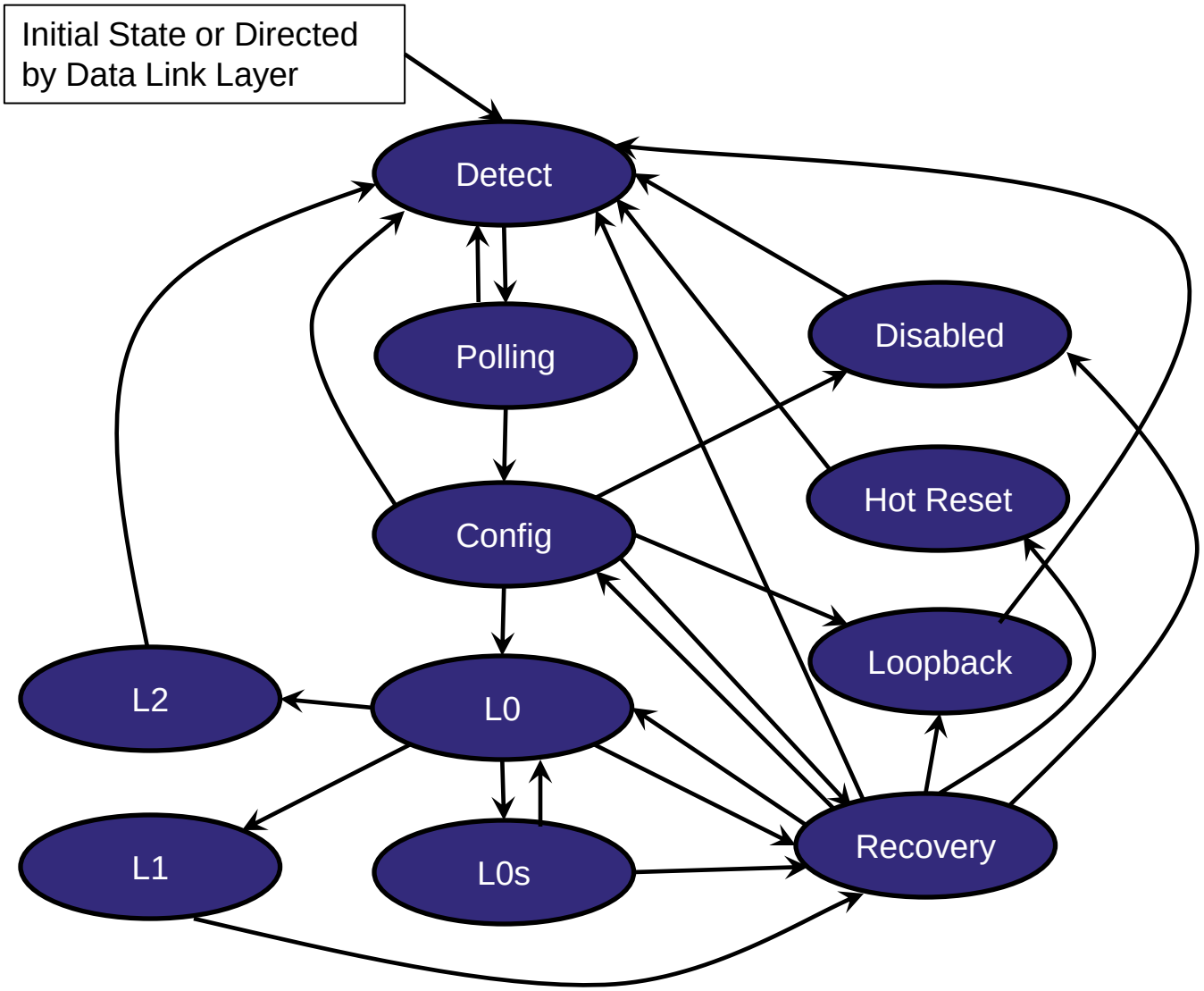
- Increasing speed demand resulted in complexity of Serial Protocols like PCIe , USB over years
  - Physical Layer has become more and more complex
    - Link Training & Status State Machine (LTSSM) becomes complex

Introduction (Cont.)



LTSSM Location

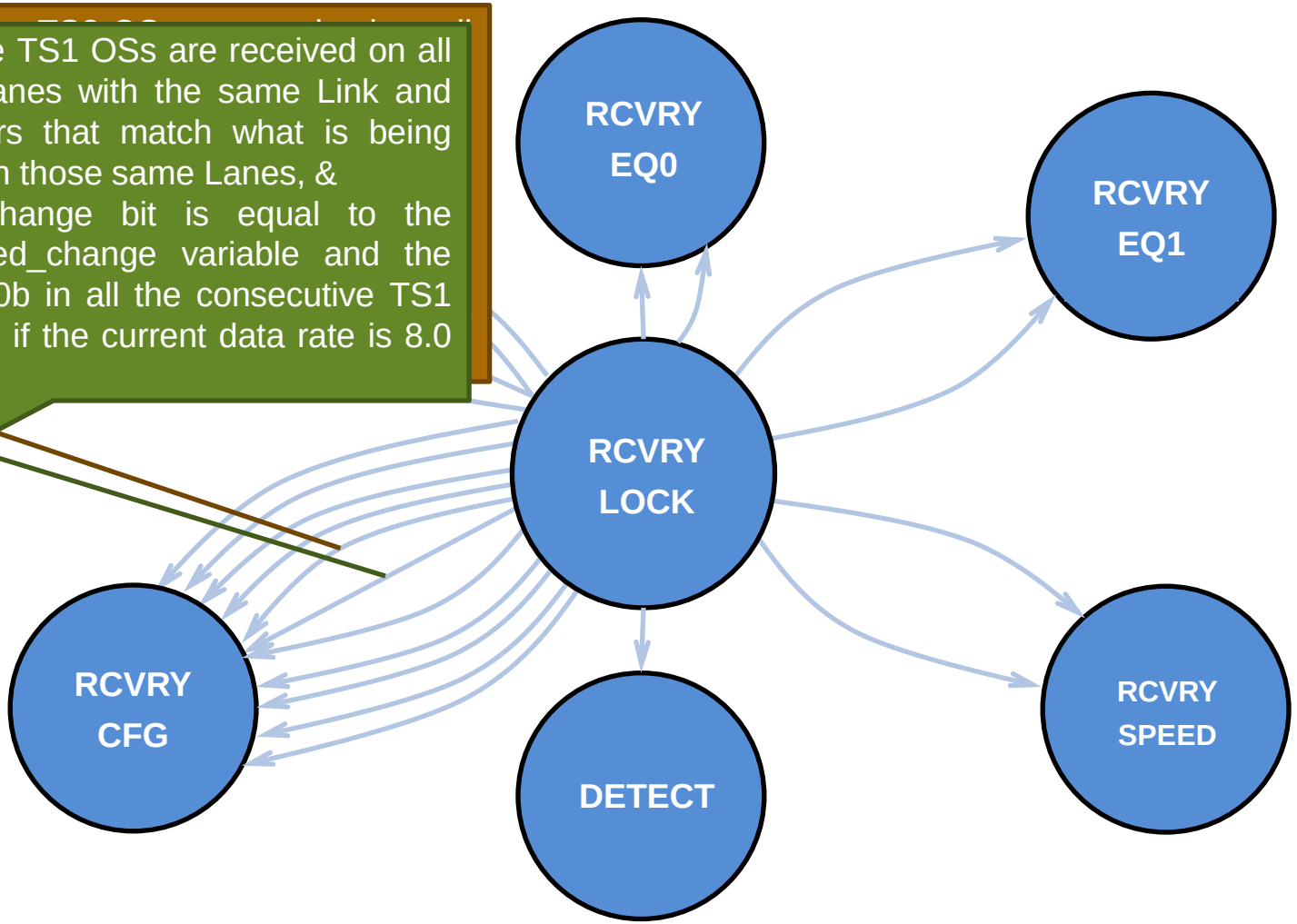
Introduction (Cont.)



Main State Diagram of LTSSM

Introduction (Cont.)

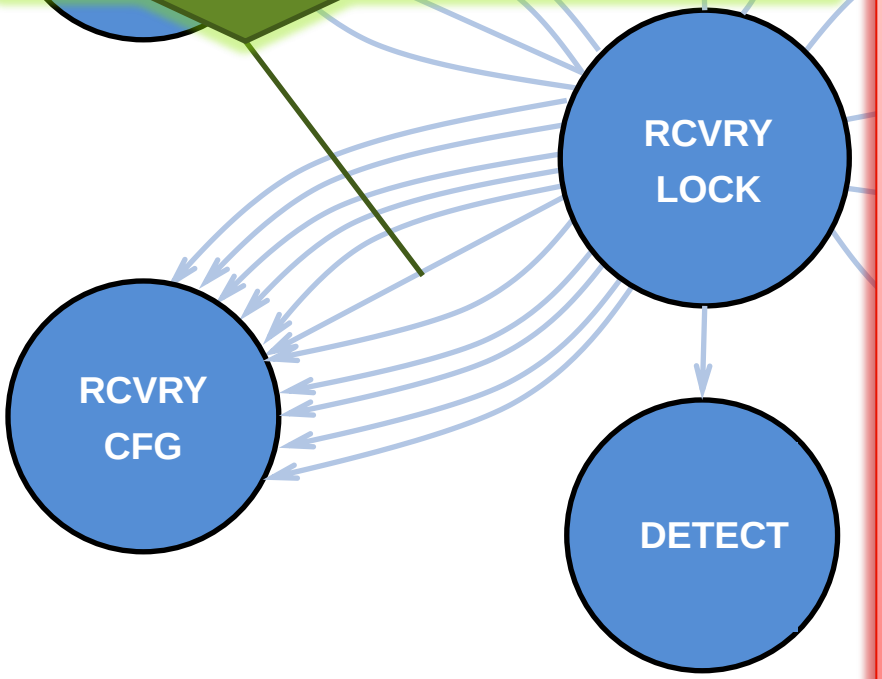
- 8 consecutive TS1 OSs are received on all configured Lanes with the same Link and Lane numbers that match what is being transmitted on those same Lanes, &
- the speed_change bit is equal to the directed_speed_change variable and the EC field is 00b in all the consecutive TS1 Ordered Sets if the current data rate is 8.0 GT/s.



State Diagram of Recovery Lock Sub State

Introduction (Cont.)

- 8 consecutive TS1 OSs are received on all configured Lanes with the same Link and Lane numbers that match what is being transmitted on those same Lanes, & the speed_change bit is equal to the directed_speed_change variable and the EC field is 00b in all the consecutive TS1 Ordered Sets if the current data rate is 8.0 GT/s.



Possible Test cases:

1) Directed Speed change = Speed change bit with 7 Good TS OS and 1 corrupted (Directed Speed change != Speed change) TS OS with All Lane condition.

2) 4 Good TS OS and 4 Bad TS OS for All Lane condition.

3) Repeat 1st and 2nd for Any Lane condition.

4) Corrupt Symbol 6 to break the persistency.

5) Lane Number not equal to the configure Lane number.

6) Link Number not equal to the configure Link number.

7) For Gen3 data rate EC is not equal to zero.

8) For Gen3 data Rate Corrupt Symbol 9.

9) Insert SKIP, EIOS between TS OS's and check the persistency is break or not.

10) Inject Missing Comma error in TS OSs..

11) Corrupt the Data Rate ID. (i.e. bit for Gen3 = 1 and Gen2 =0)

All above test cases are with ALL Lane / ANY Lane condition and also with Differnt Error Distribution (i.e. 7-1 , 4-4) .

Introduction (Cont.)

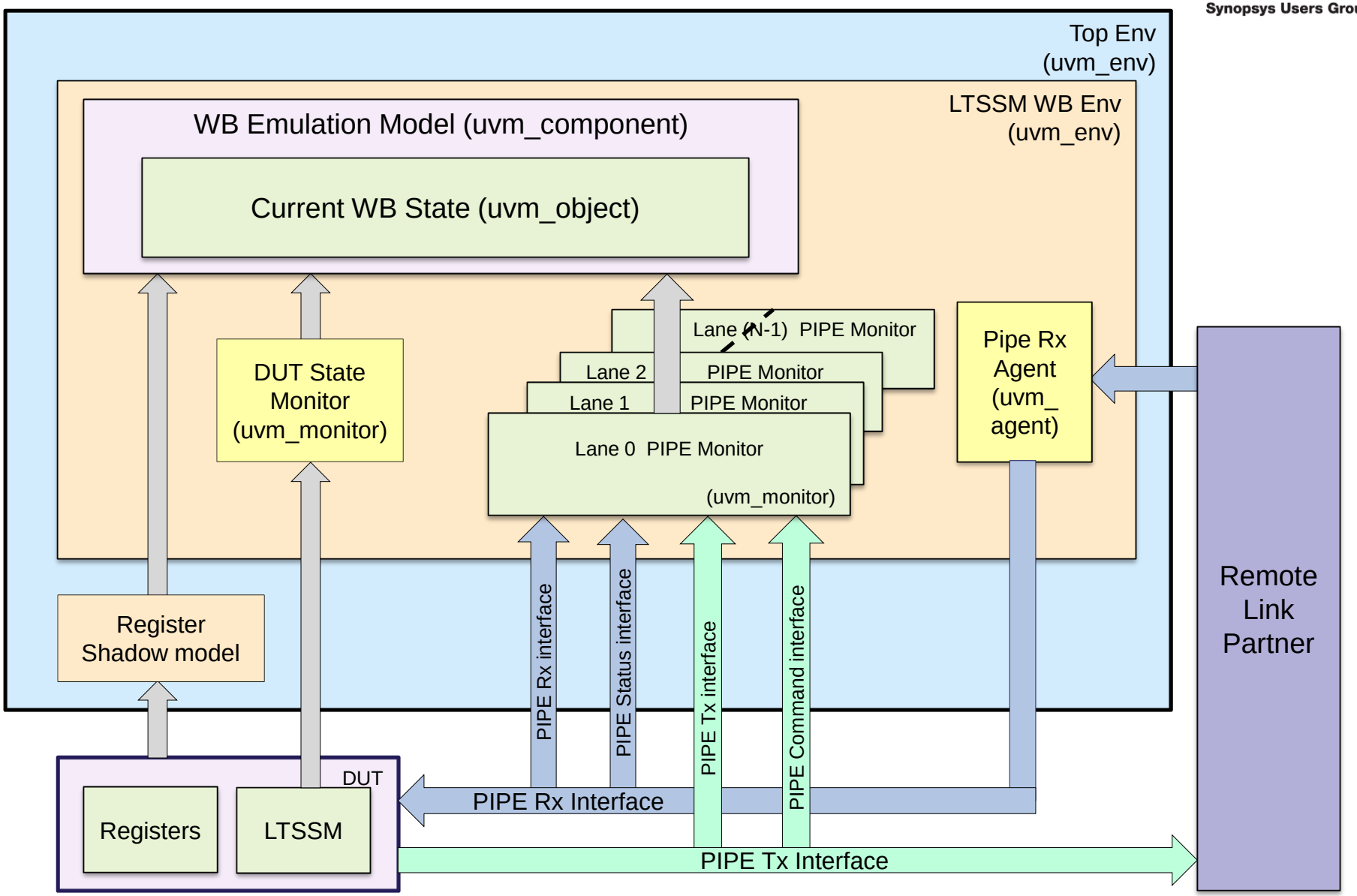
- Modeling of LTSSM becomes very critical and important
  - Need for LTSSM White Box(WB) reference model
  - Generation of Stimulus , Error Injection and Coverage

- This WB approach is not just limited to the LTSSM, but can be re-purposed to verify any other complex state machine

LTSSM Test bench Architecture

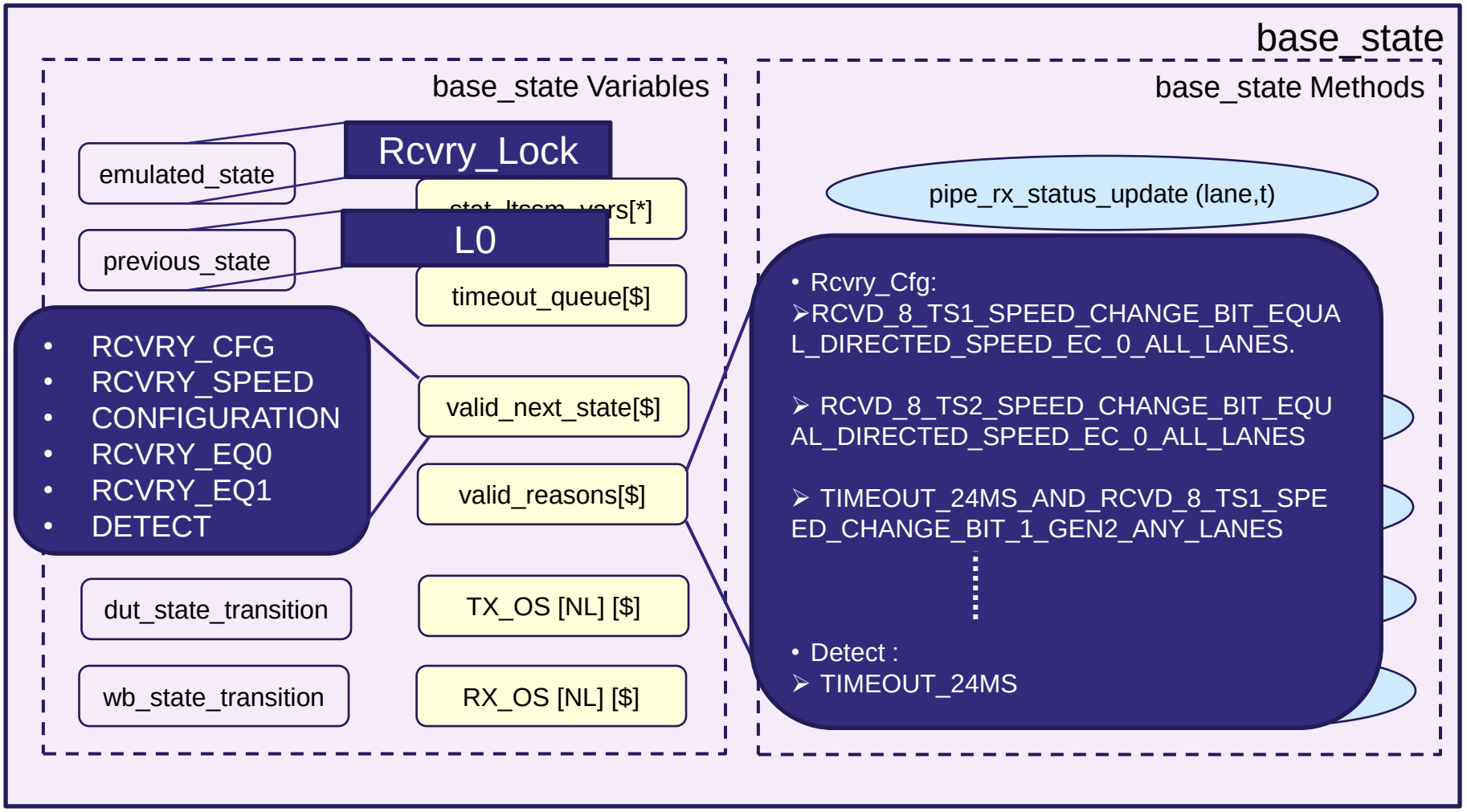
- Introduction
- **LTSSM Test bench Architecture**
- Rx PIPE Agent & Error Stimulus Generation
- Coverage & Debug Features
- Results & Summary

LTSSM Testbench Architecture



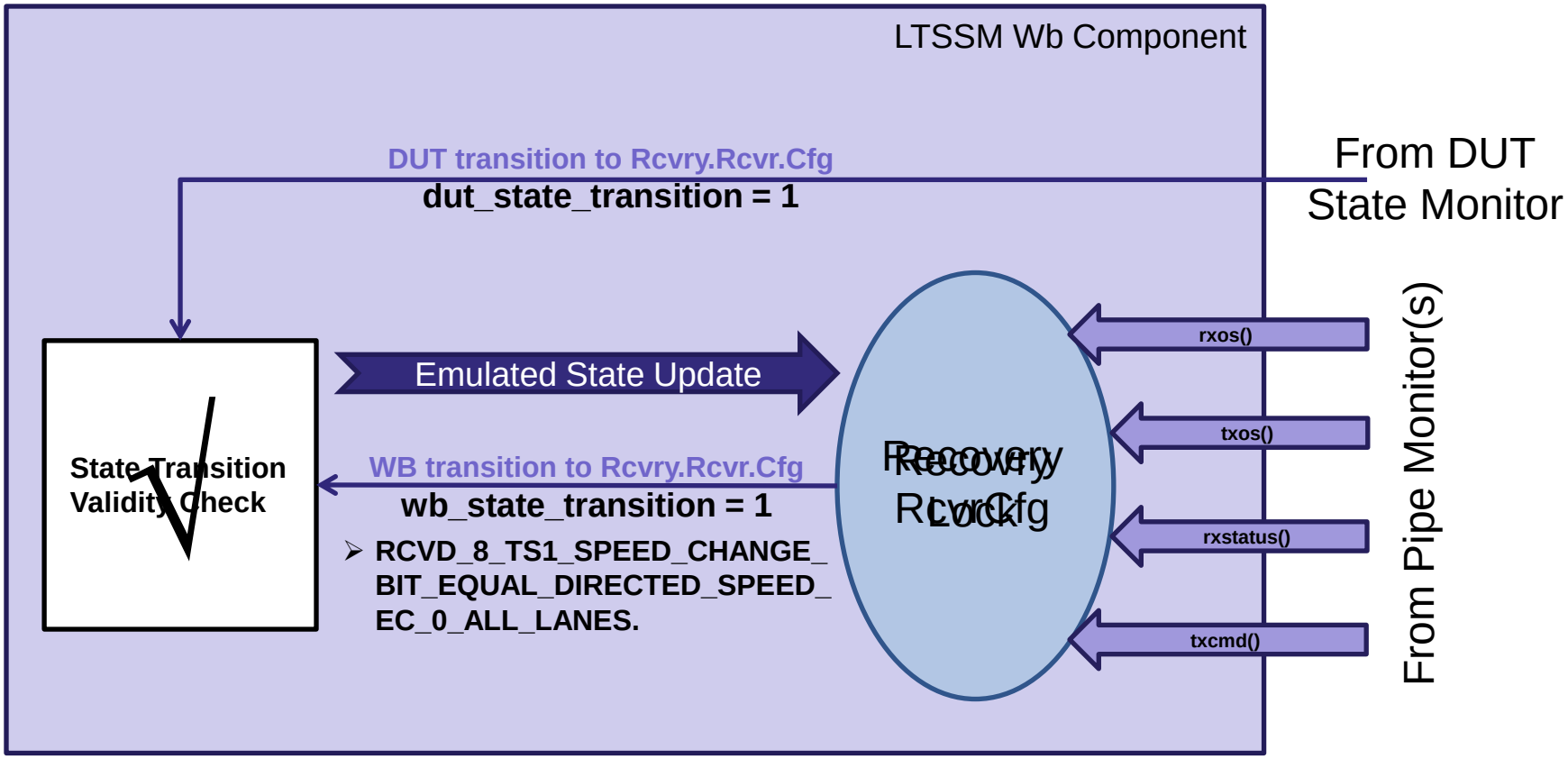
LTSSM Testbench Architecture (Cont.)

LTSSM WB Base State



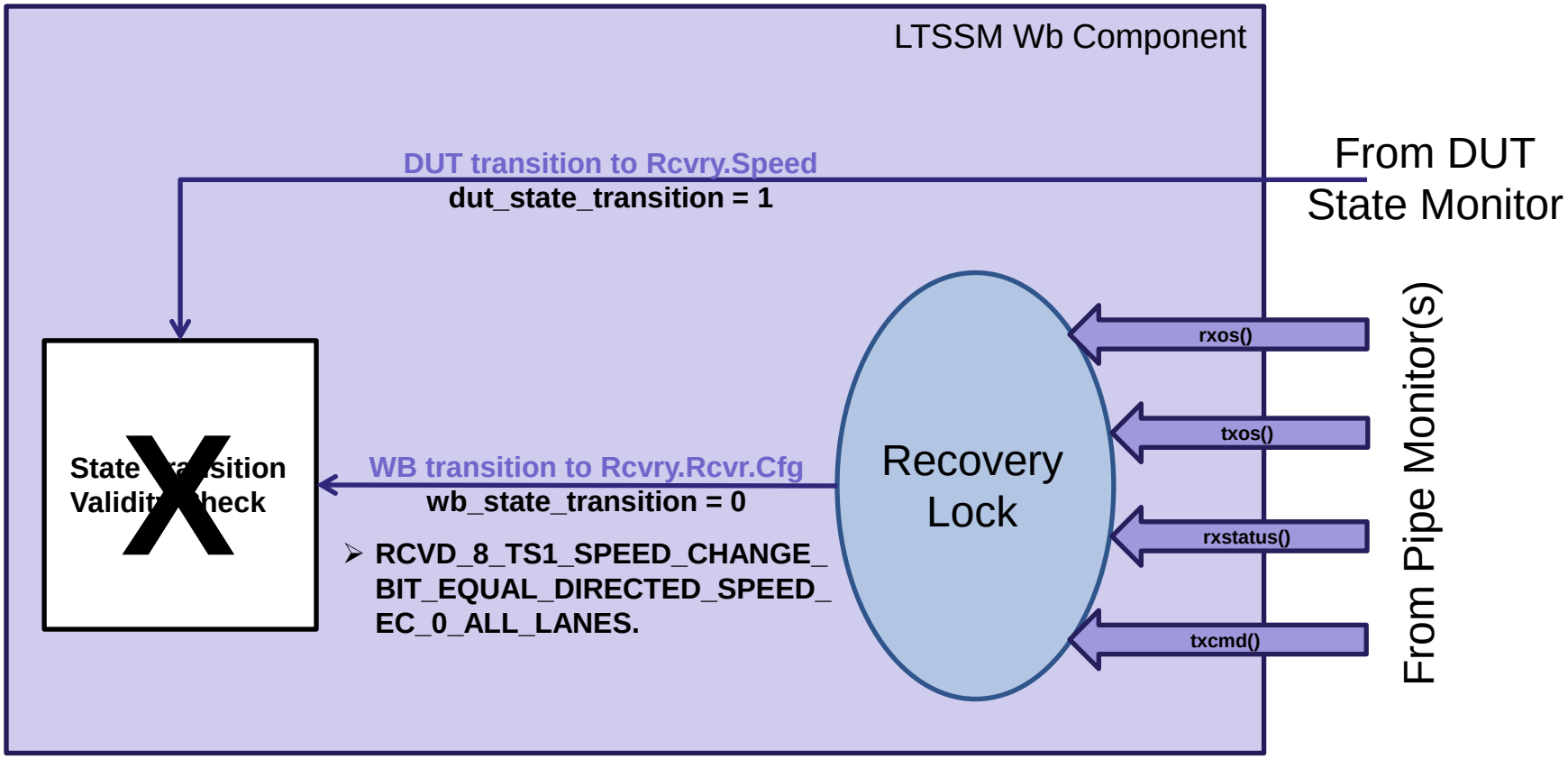
LTSSM Testbench Architecture (Cont.)

LTSSM WB Base State (Contd.)



LTSSM Testbench Architecture (Cont.)

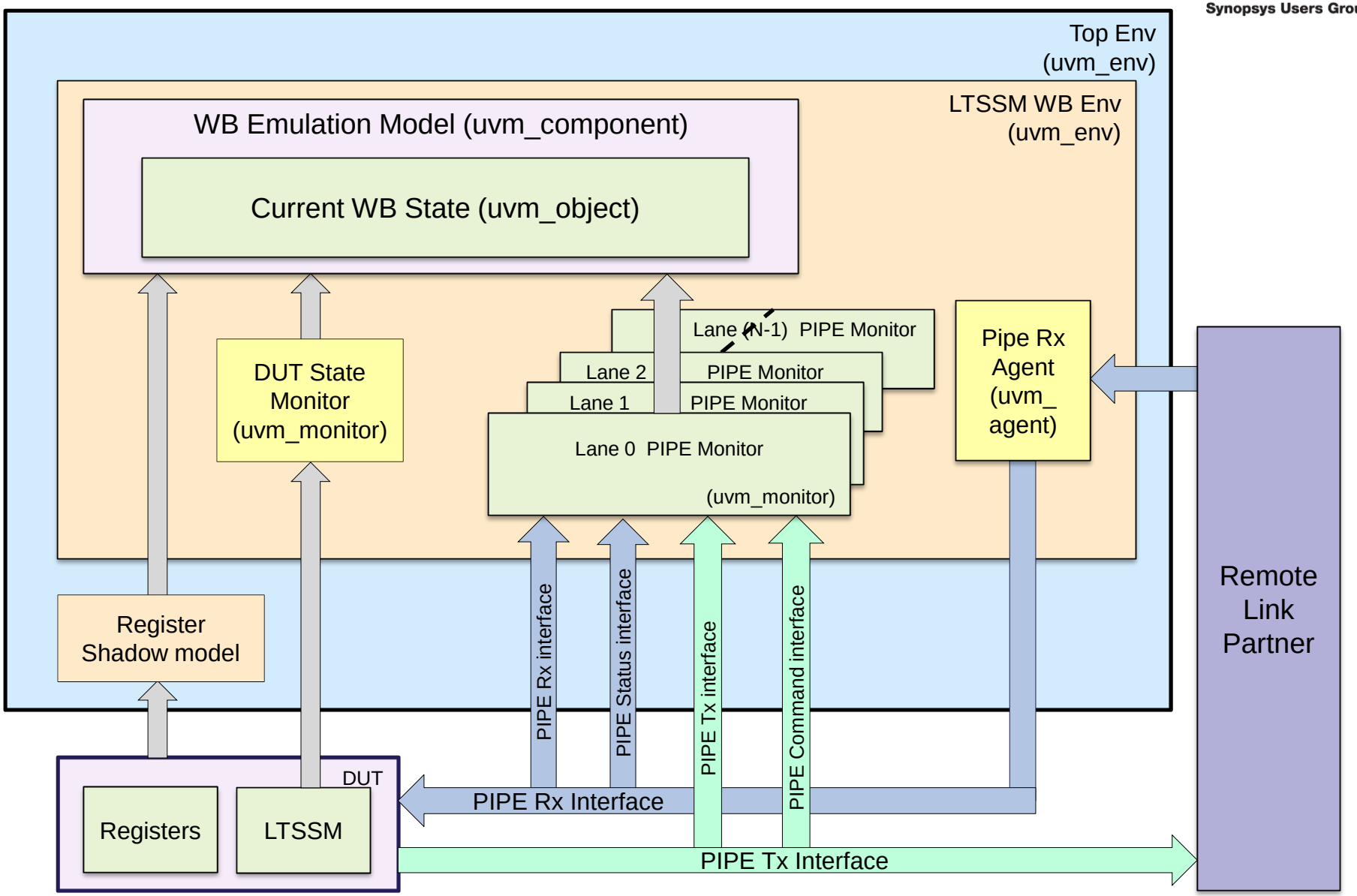
LTSSM WB Base State (Contd.)



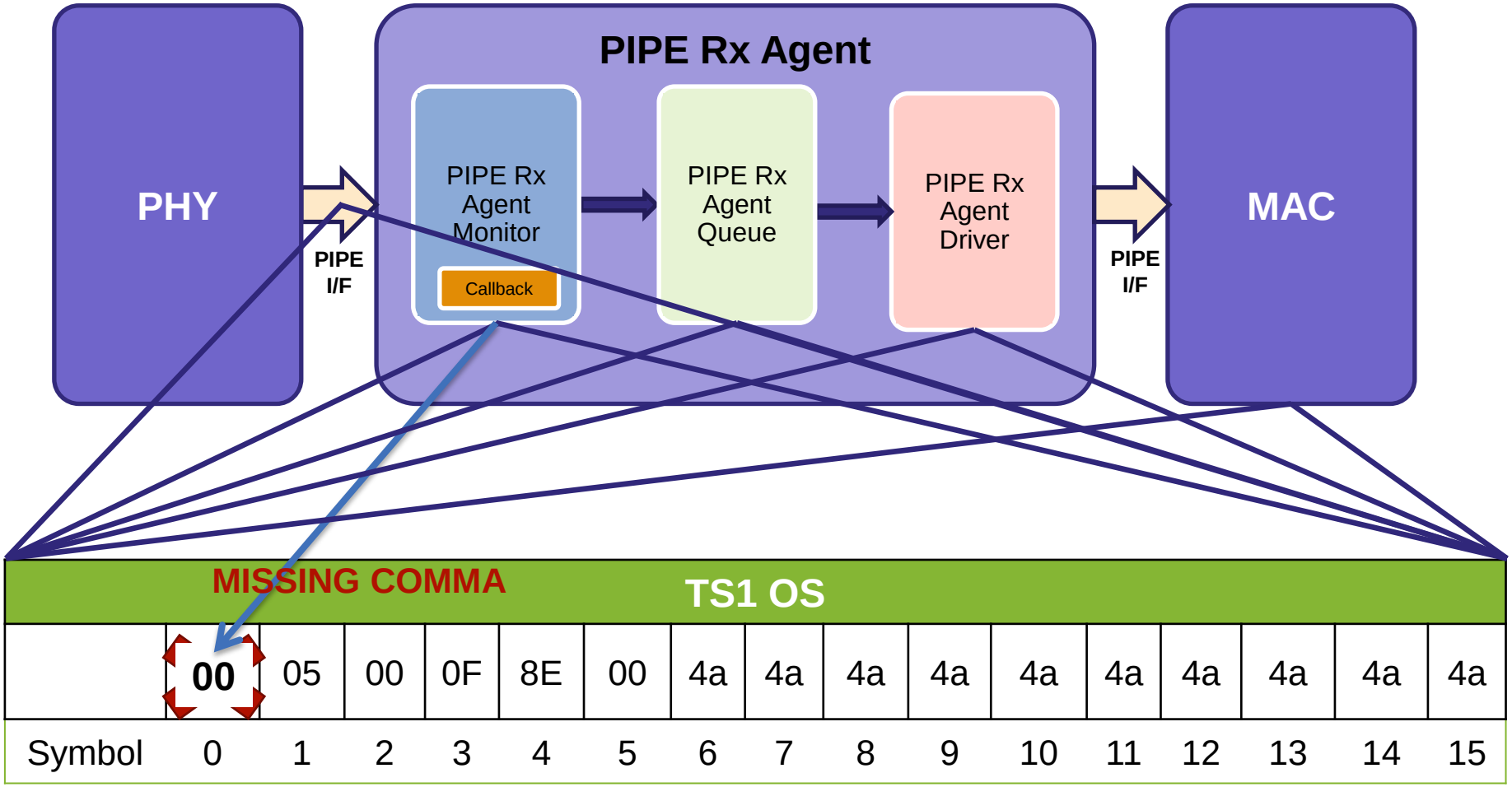
Rx PIPE Agent & Error Stimulus Generation

- Introduction
- LTSSM Test bench Architecture
- **Rx PIPE Agent & Error Stimulus Generation**
- Coverage & Debug Features
- Results & Summary

Rx PIPE Agent & Error Stimulus Generation

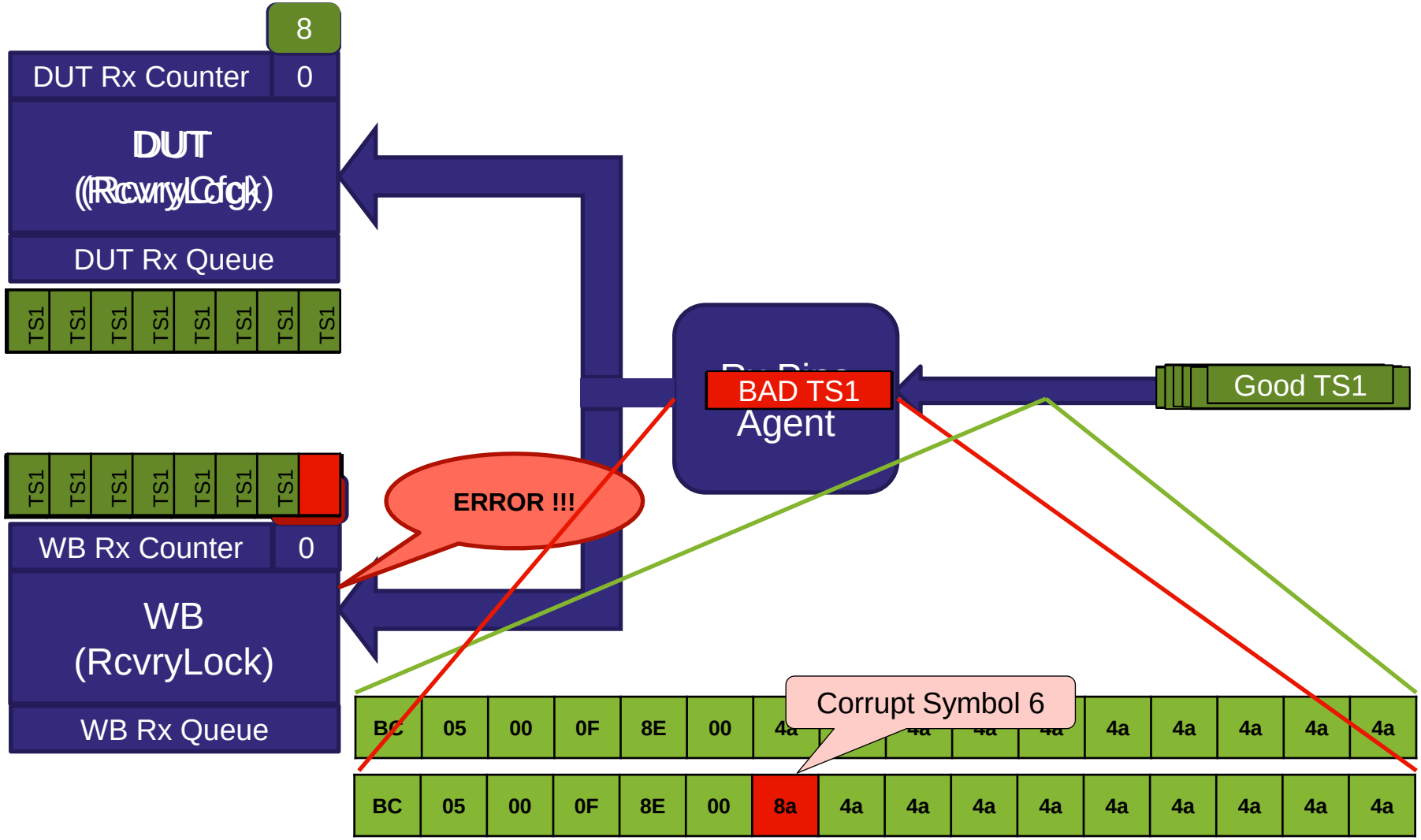


Rx PIPE Agent & Error Stimulus Generation



Rx PIPE Agent & Error Stimulus Generation (Cont.)

How Rx PIPE Agent helps generating corner case scenarios?



Coverage & Debug Features

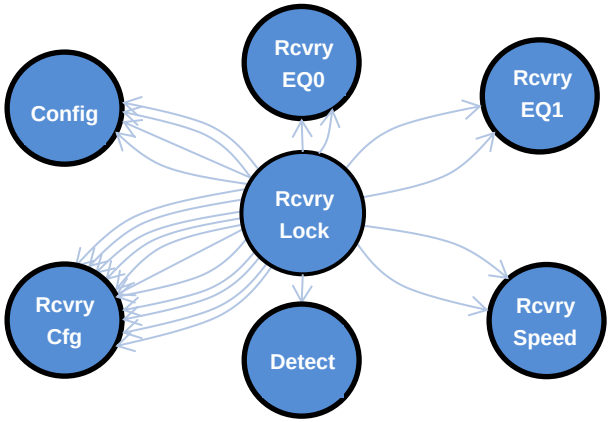
- Introduction
- LTSSM Test bench Architecture
- Rx PIPE Agent & Error Stimulus Generation
- **Coverage & Debug Features**
- Results & Summary

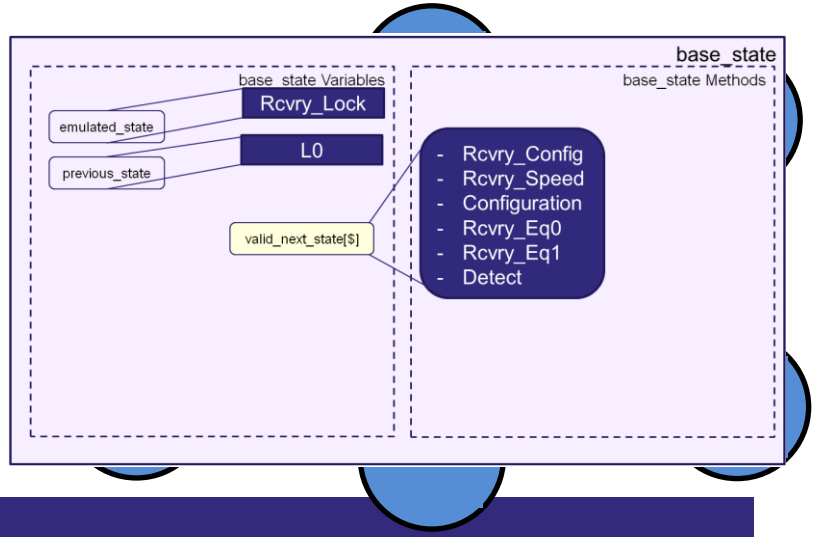
Coverage & Debug Features

- Every LTSSM sub-states, create a verification plan, which consists of the following coverage items
  - All possible state transitions for the sub-state
  - All possible state transition conditions/reasons for the sub-state
  - Transmit rules for the sub-state
  - Stimulus(Error Injection) coverage for the sub-state
  - Boundry coverage for the sub-state
- Coverage plan for state transitions and state transition conditions is extracted using a script from the emulated WB state.

Coverage & Debug Features (Cont.)

All Possible State Transitions

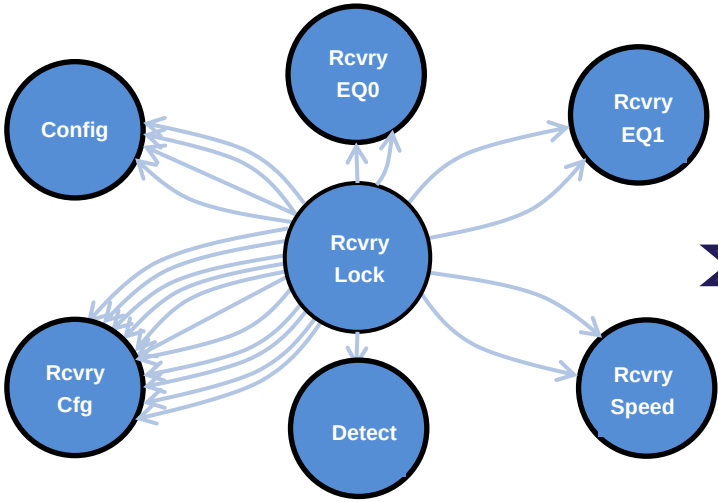


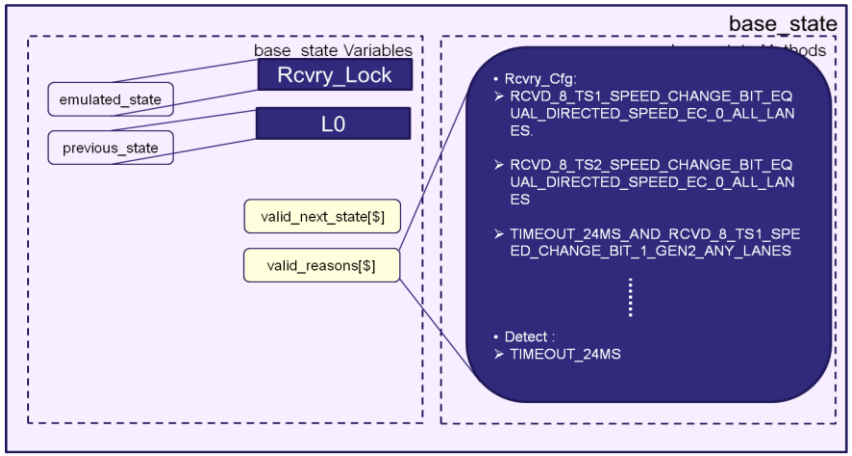



```
coveragroup cg_ltssm_state_transitions
            with function sample(dut_state tr);
  cp_ltssm_state_transitions: coverpoint tr.e_current_state
  {
    bins lock_to_rcfg   = (RCVRY_LOCK => RCVRY_RCVR_CFG);
    bins lock_to_speed  = (RCVRY_LOCK => RCVRY_SPEED);
    bins lock_to_eq0    = (RCVRY_LOCK => RCVRY_EQ0);
    bins lock_to_eq1    = (RCVRY_LOCK => RCVRY_EQ1);
    bins lock_to_cfg    = (RCVRY_LOCK => CFG_LINKWD_START);
    bins lock_to_detect = (RCVRY_LOCK => DETECT_QUIET);
  }
endgroup : cg_ltssm_state_transitions
```

Coverage & Debug Features (Cont.)

All Possible State Transitions Reasons






```
covergroup cg_ltssm_wb (ref base_state state)
           with function sample(base_state tr);

  cp_reason : coverpoint tr.reason {
    bins valid_reason[] = cp_reason with (
      validate_reasons(e_transition_code'(item), state) );
  }

  cp_expected_state : coverpoint tr.wb_expecte_state {
    bins valid_transition[] = cp_expected_state with (
      validate_next_state(e_ltssm_state'(item), state) );
  }

 cc_next_state_reason : cross cp_expected_state, cp_reason;

endgroup : cg_ltssm_wb
```

Coverage & Debug Features (Cont.)

Transmit Rules

```
//Check link and lane numbers transmitted in
// recovery lock matching with Received TS1/TS2.
event check_rlock_ts1_link_lane_num;
property p_check_rlock_ts1_link_lane_num();
  @(check_rlock_ts1_link_lane_num ) disable iff(!core_rst_n)
    1==1;
endproperty : p_check_rlock_ts1_link_lane_num

cov_p_check_rlock_ts1_link_lane_num :
  cover property( p_check_rlock_ts1_link_lane_num() );
```

```
if( sel_link_num==t.symbols[1] &&
    sel_lane_num[lane]==t.symbols[2])
  // Trigger event for cover property
  ->i_utbCovProp.check_rlock_ts1_link_lane_num;
else
  `uvm_error("RCVRY_LOCK", $psprintf("Failed check")
```

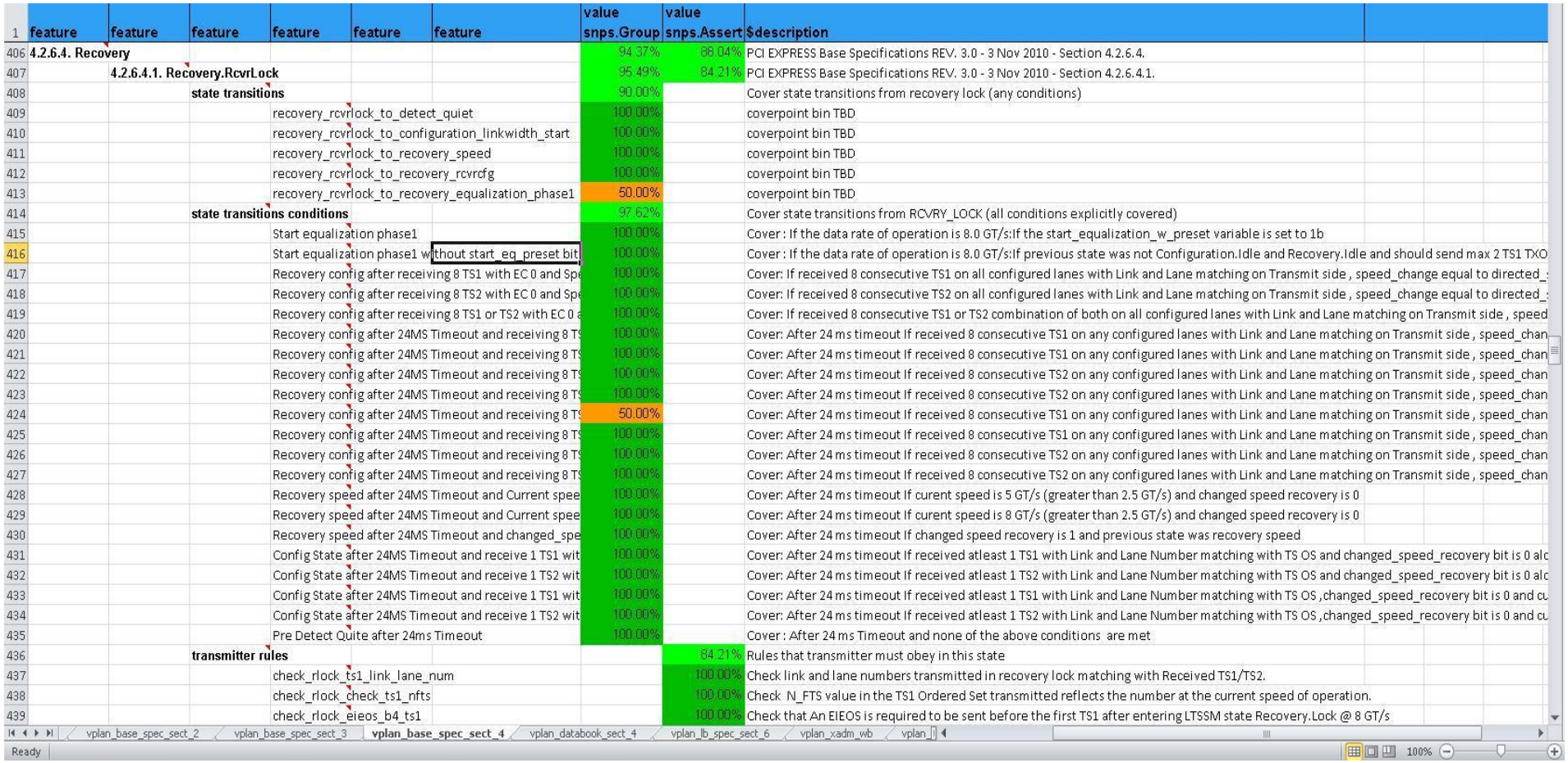
Coverage & Debug Features (Cont.)

Boundary Coverage

```
cp_good_ts1 : coverpoint (cov_ts_pad_lane_num[TS1]==0 &&
cov_ts_pad_link_num[TS1]==0)
{
  type_option.comment = "Covering a sequence where no
transition is expected.";
  // 7 good TS1s followed by one bad TS1
  bins seven_good = (1=>1=>1=>1=>1=>1=>1=>0 );
}
```

Coverage & Debug Features (Cont.)

Screenshot of Verification Plan XML output



1	feature	feature	feature	feature	feature	feature	value snps.Group	value snps.Assert	$description
406	4.2.6.4. Recovery						94.37%	88.04%	PCI EXPRESS Base Specifications REV. 3.0 - 3 Nov 2010 - Section 4.2.6.4.
407		4.2.6.4.1. Recovery.RcvrLock					95.49%	84.21%	PCI EXPRESS Base Specifications REV. 3.0 - 3 Nov 2010 - Section 4.2.6.4.1.
408			state transitions				90.00%		Cover state transitions from recovery lock (any conditions)
409				recovery_rcvrlock_to_detect_quiet			100.00%		coverpoint bin TBD
410				recovery_rcvrlock_to_configuration_linkwidth_start			100.00%		coverpoint bin TBD
411				recovery_rcvrlock_to_recovery_speed			100.00%		coverpoint bin TBD
412				recovery_rcvrlock_to_recovery_rcvrcfg			100.00%		coverpoint bin TBD
413				recovery_rcvrlock_to_recovery_equalization_phase1			50.00%		coverpoint bin TBD
414			state transitions conditions				97.62%		Cover state transitions from RCVRY_LOCK (all conditions explicitly covered)
415				Start equalization phase1			100.00%		Cover : If the data rate of operation is 8.0 GT/s:If the start_equalization_w_preset variable is set to 1b
416				Start equalization phase1 without start_eq_preset bit			100.00%		Cover : If the data rate of operation is 8.0 GT/s:If previous state was not Configuration.Idle and Recovery.Idle and should send max 2 TS1 TXO
417				Recovery config after receiving 8 TS1 with EC 0 and Spe			100.00%		Cover: If received 8 consecutive TS1 on all configured lanes with Link and Lane matching on Transmit side , speed_change equal to directed_
418				Recovery config after receiving 8 TS2 with EC 0 and Spe			100.00%		Cover: If received 8 consecutive TS2 on all configured lanes with Link and Lane matching on Transmit side , speed_change equal to directed_
419				Recovery config after receiving 8 TS1 or TS2 with EC 0 a			100.00%		Cover: If received 8 consecutive TS1 or TS2 combination of both on all configured lanes with Link and Lane matching on Transmit side , speed
420				Recovery config after 24MS Timeout and receiving 8 TS			100.00%		Cover: After 24 ms timeout If received 8 consecutive TS1 on any configured lanes with Link and Lane matching on Transmit side , speed_chan
421				Recovery config after 24MS Timeout and receiving 8 TS			100.00%		Cover: After 24 ms timeout If received 8 consecutive TS1 on any configured lanes with Link and Lane matching on Transmit side , speed_chan
422				Recovery config after 24MS Timeout and receiving 8 TS			100.00%		Cover: After 24 ms timeout If received 8 consecutive TS2 on any configured lanes with Link and Lane matching on Transmit side , speed_chan
423				Recovery config after 24MS Timeout and receiving 8 TS			100.00%		Cover: After 24 ms timeout If received 8 consecutive TS2 on any configured lanes with Link and Lane matching on Transmit side , speed_chan
424				Recovery config after 24MS Timeout and receiving 8 TS			50.00%		Cover: After 24 ms timeout If received 8 consecutive TS1 on any configured lanes with Link and Lane matching on Transmit side , speed_chan
425				Recovery config after 24MS Timeout and receiving 8 TS			100.00%		Cover: After 24 ms timeout If received 8 consecutive TS1 on any configured lanes with Link and Lane matching on Transmit side , speed_chan
426				Recovery config after 24MS Timeout and receiving 8 TS			100.00%		Cover: After 24 ms timeout If received 8 consecutive TS2 on any configured lanes with Link and Lane matching on Transmit side , speed_chan
427				Recovery config after 24MS Timeout and receiving 8 TS			100.00%		Cover: After 24 ms timeout If received 8 consecutive TS2 on any configured lanes with Link and Lane matching on Transmit side , speed_chan
428				Recovery speed after 24MS Timeout and Current spee			100.00%		Cover: After 24 ms timeout If curent speed is 5 GT/s (greater than 2.5 GT/s) and changed speed recovery is 0
429				Recovery speed after 24MS Timeout and Current spee			100.00%		Cover: After 24 ms timeout If curent speed is 8 GT/s (greater than 2.5 GT/s) and changed speed recovery is 0
430				Recovery speed after 24MS Timeout and changed_spe			100.00%		Cover: After 24 ms timeout If changed speed recovery is 1 and previous state was recovery speed
431				Config State after 24MS Timeout and receive 1 TS1 wit			100.00%		Cover: After 24 ms timeout If received atleast 1 TS1 with Link and Lane Number matching with TS OS and changed_speed_recovery bit is 0 alc
432				Config State after 24MS Timeout and receive 1 TS2 wit			100.00%		Cover: After 24 ms timeout If received atleast 1 TS2 with Link and Lane Number matching with TS OS and changed_speed_recovery bit is 0 alc
433				Config State after 24MS Timeout and receive 1 TS1 wit			100.00%		Cover: After 24 ms timeout If received atleast 1 TS1 with Link and Lane Number matching with TS OS ,changed_speed_recovery bit is 0 and cu
434				Config State after 24MS Timeout and receive 1 TS2 wit			100.00%		Cover: After 24 ms timeout If received atleast 1 TS2 with Link and Lane Number matching with TS OS ,changed_speed_recovery bit is 0 and cu
435				Pre Detect Quite after 24ms Timeout			100.00%		Cover : After 24 ms Timeout and none of the above conditions are met
436			transmitter rules					84.21%	Rules that transmitter must obey in this state
437				check_rlock_ts1_link_lane_num				100.00%	Check link and lane numbers transmitted in recovery lock matching with Received TS1/TS2.
438				check_rlock_check_ts1_nfts				100.00%	Check N_FTS value in the TS1 Ordered Set transmitted reflects the number at the current speed of operation.
439				check_rlock_eieos_b4_ts1				100.00%	Check that An EIEOS is required to be sent before the first TS1 after entering LTSSM state Recovery.Lock @ 8 GT/s

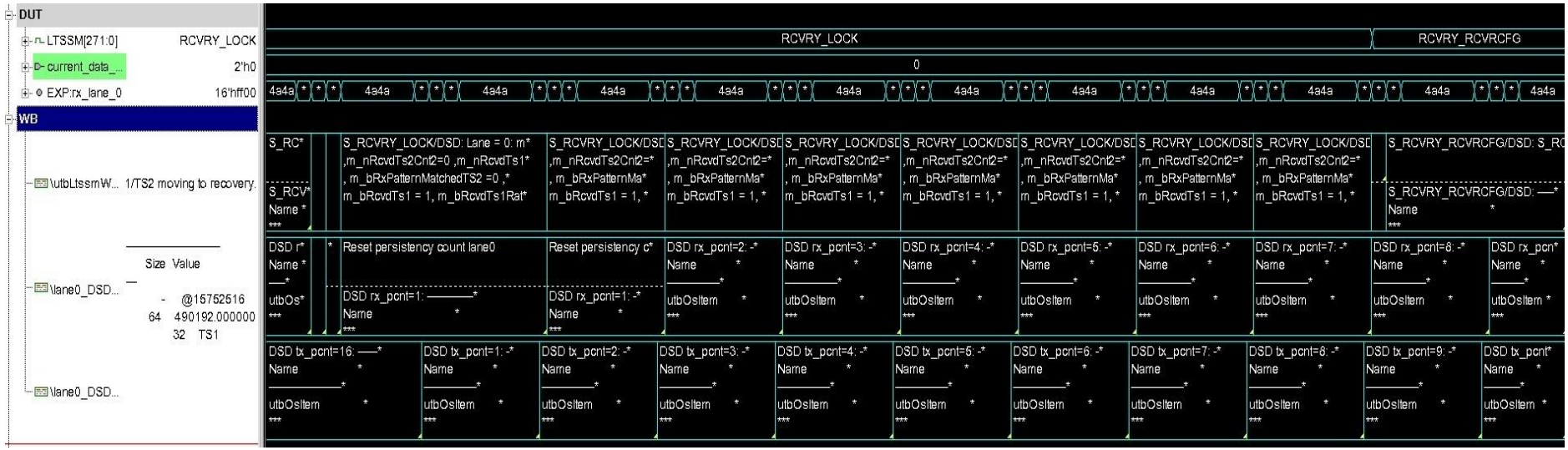
vplan_base_spec_sect_2 | vplan_base_spec_sect_3 | **vplan_base_spec_sect_4** | vplan_databook_sect_4 | vplan_lb_spec_sect_6 | vplan_xadm_wb | vplan

Ready 100%

Coverage & Debug Features (Cont.)

Transaction Logging

- Create a separate transaction log which would just log the required transaction and displays within
- This log can be viewed within the waves along with the design signals
- Helps reducing the debugging time as we can see the design and model behaviour side by side



Results & Summary

- Introduction
- LTSSM Test bench Architecture
- Rx PIPE Agent & Error Stimulus Generation
- Coverage & Debug Features
- **Results & Summary**

Results & Summary

- Once the model is stable after some initial efforts , it was much easier and quicker to verify a complex design like PCIe LTSSM
- With the approach presented in this paper we improved the IP quality
- No. of tests are reduced to around 60 compared to 700 directed test cases
- Auto-extracted coverage reports give a very high level of confidence that strict compliance is met
- This WB approach is not just limited to the LTSSM, but can be re-purposed to verify any other complex state machine

Thank You