2016 DESIGN AND VERIFICATION™ DVCON CONFERENCE AND EXHIBITION UNITED STATES

A Universal DFT Verification Environment: Filling the Gap between Function Simulation and ATE Test

Rui Huang

Advanced Micro Devices

AMD

The Background

- ATEs need test patterns described by STIL (Standard Test Interface Language) or other test languages.
- In DFT domain, the test patterns running on an ATE can be categorized in two types: scan related and none-scan related.
- Scan related test patterns can be generated using ATPG tools.

The Issue with Non-scan Related Test Patterns…

- Non-scan related test patterns are normally created by design verification engineers using languages such as System Verilog or C++.

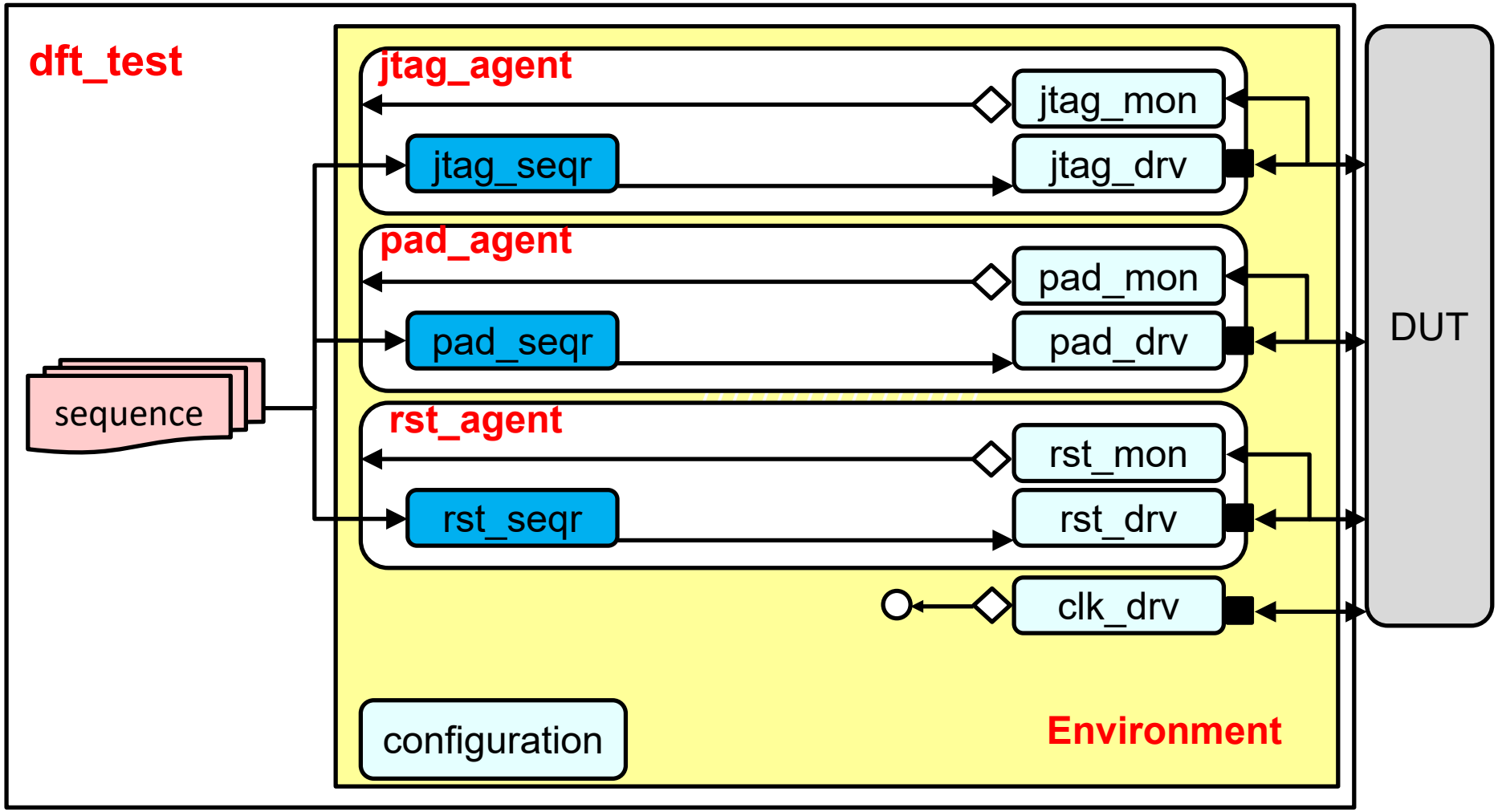
- How this is done in practice?

The Issue with Non-scan Related Test Patterns…

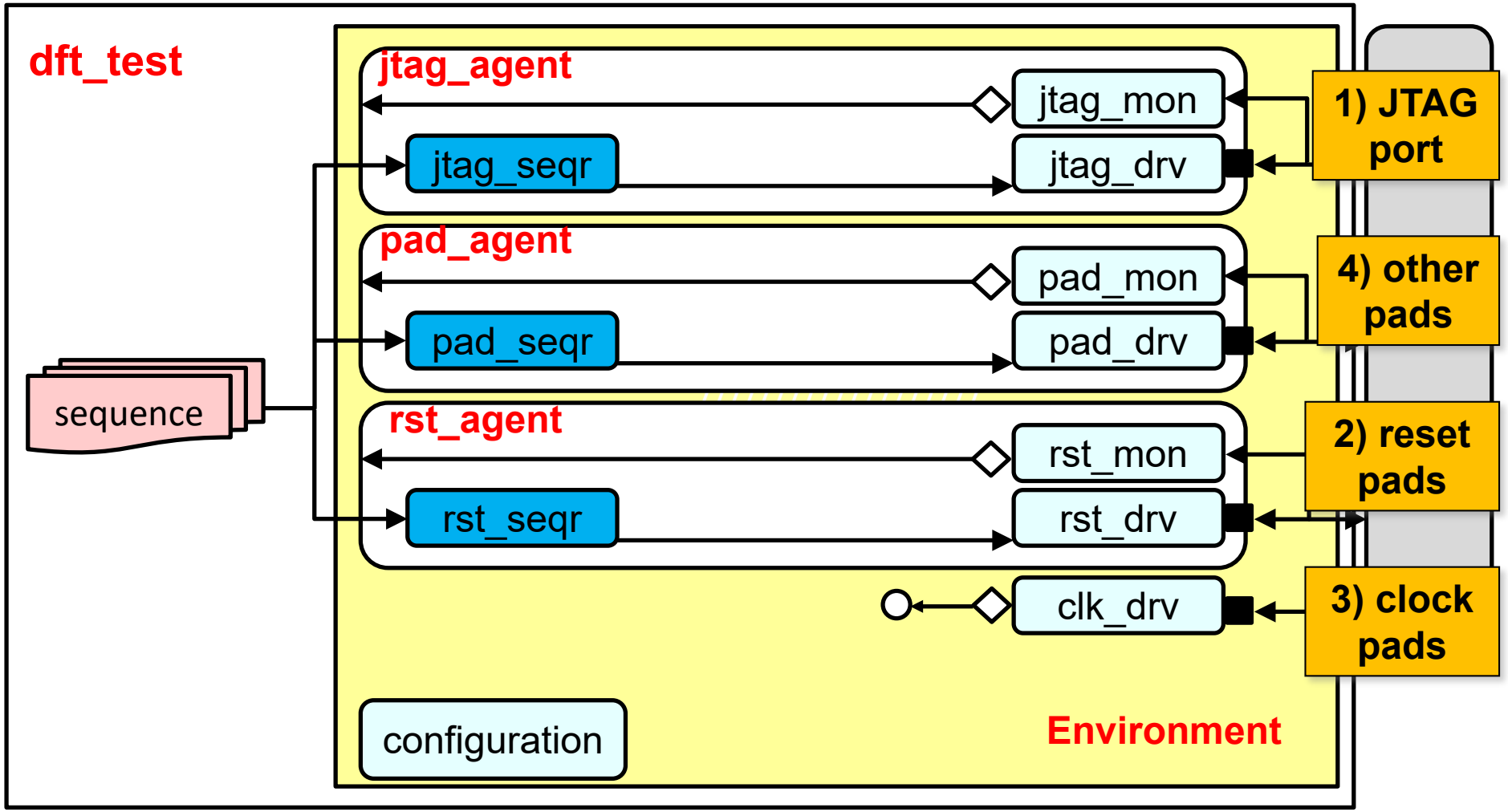
- A dedicated team to transfer function simulation to ATE test environment.
- Some in-house automation flows are developed to enforce complex rules on test writing and register specification documentation.

- Is there a more efficient and universal solution?

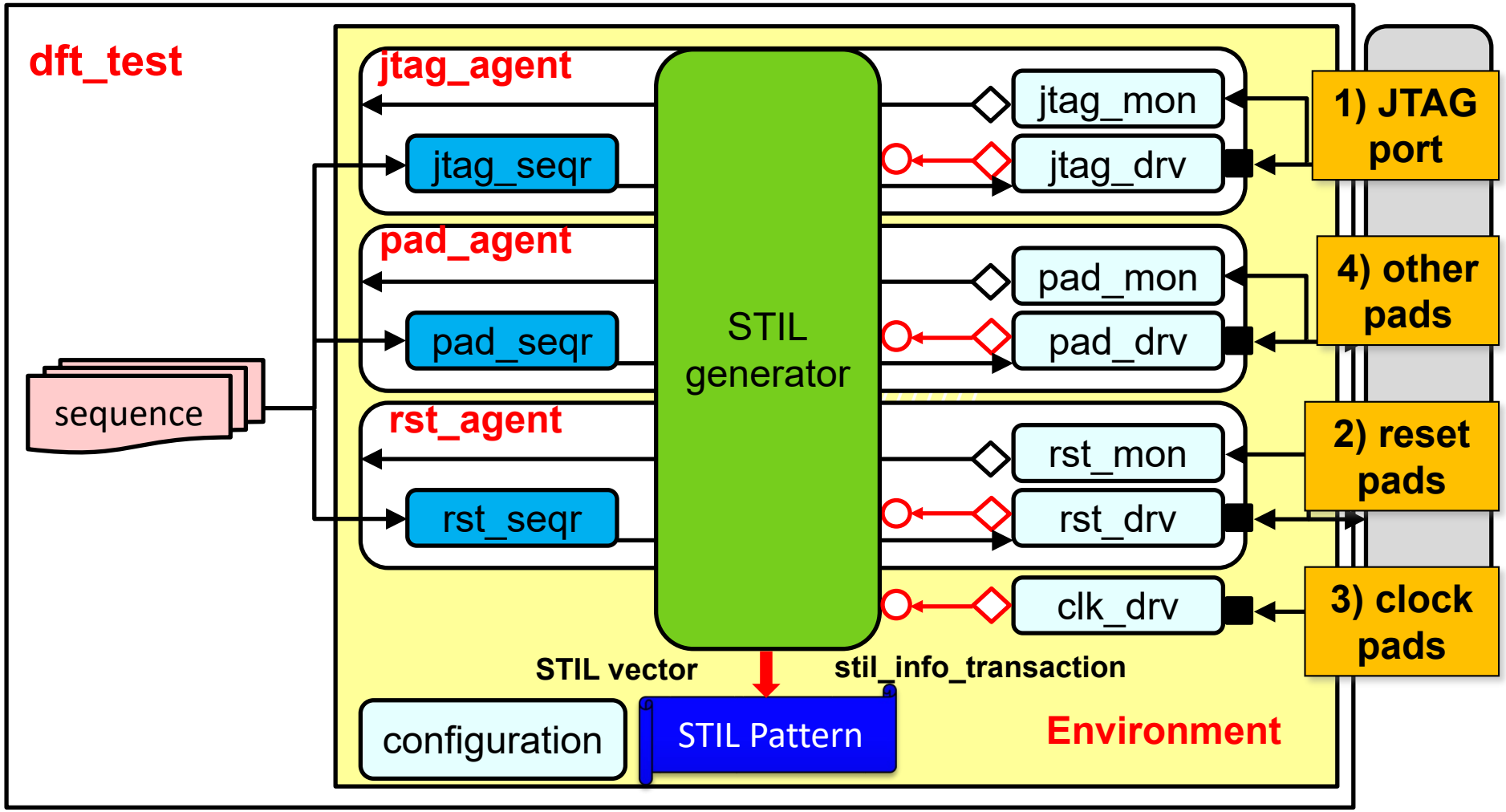
A More Efficient and Universal Solution



A More Efficient and Universal Solution

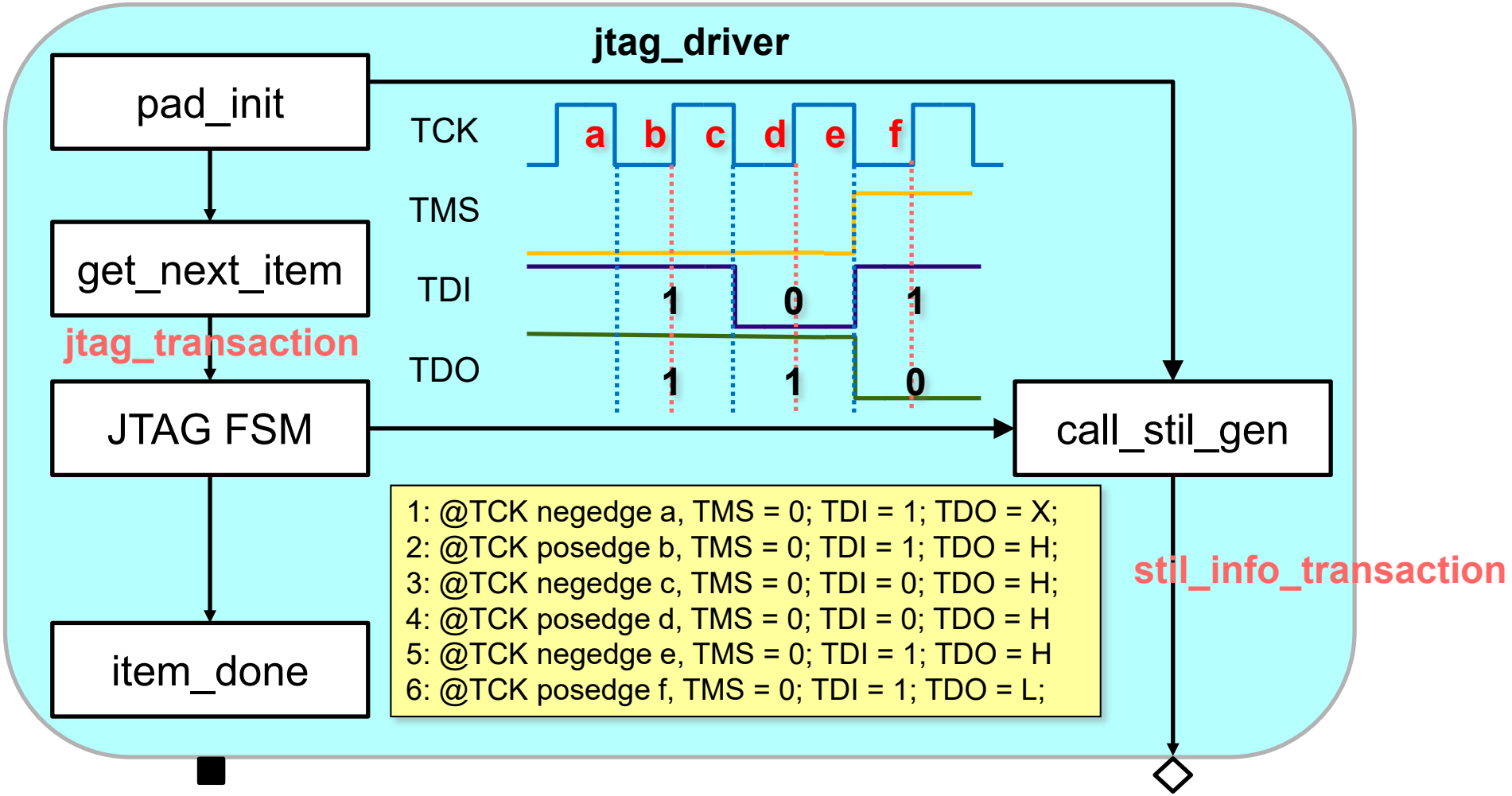


A More Efficient and Universal Solution



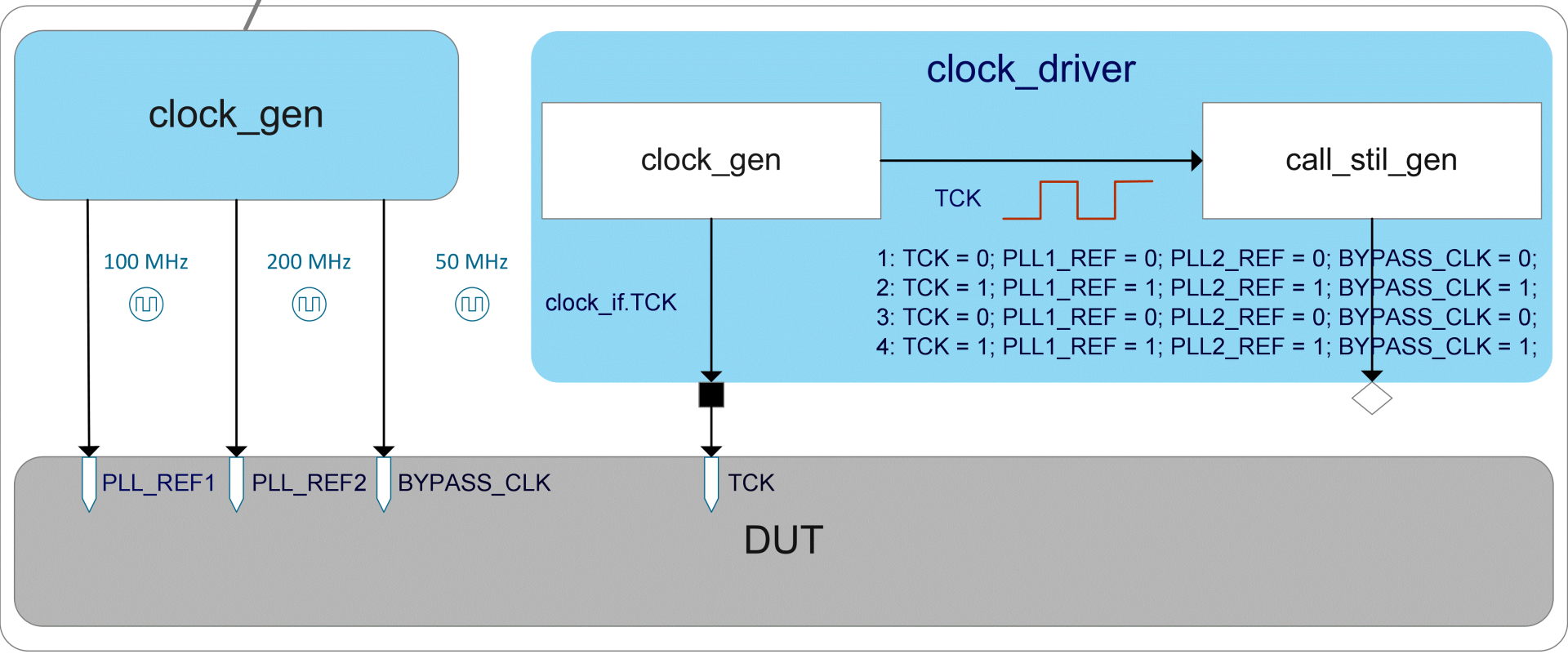
JTAG Driver

JTAG FSM is in shift DR state. It will send three bits 101 on TDI and sample TDO with golden compare data (three bits 110).

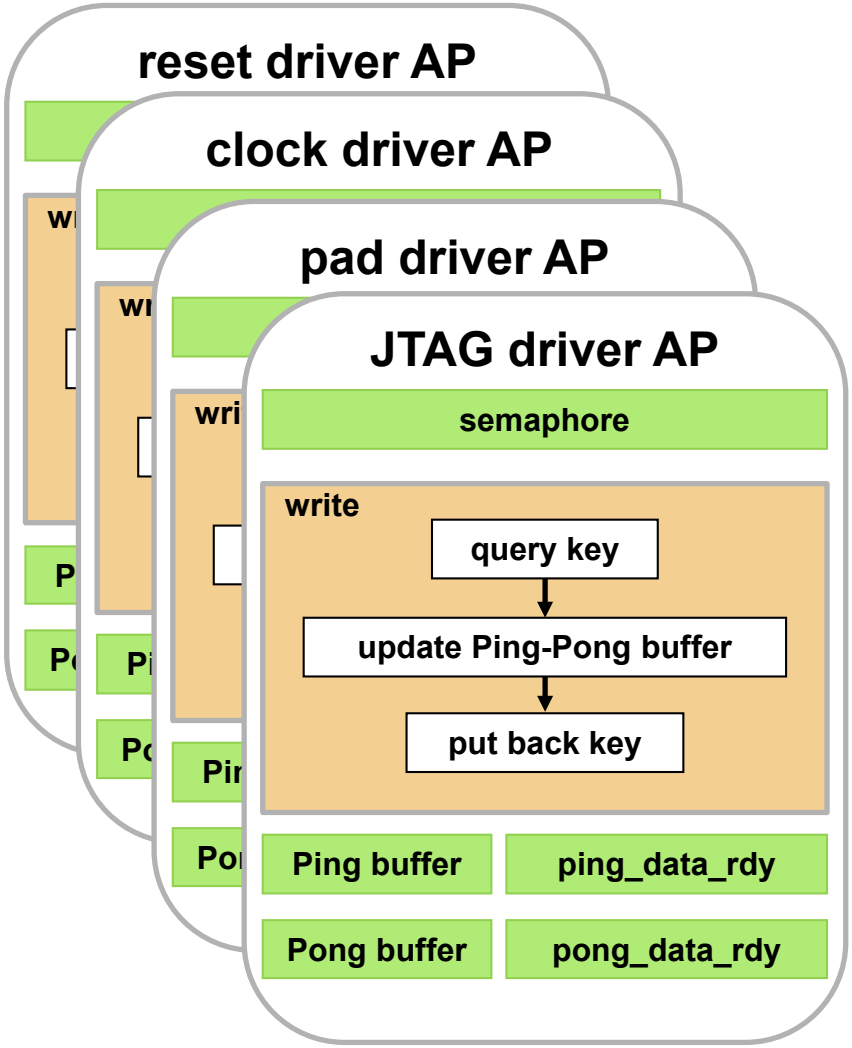


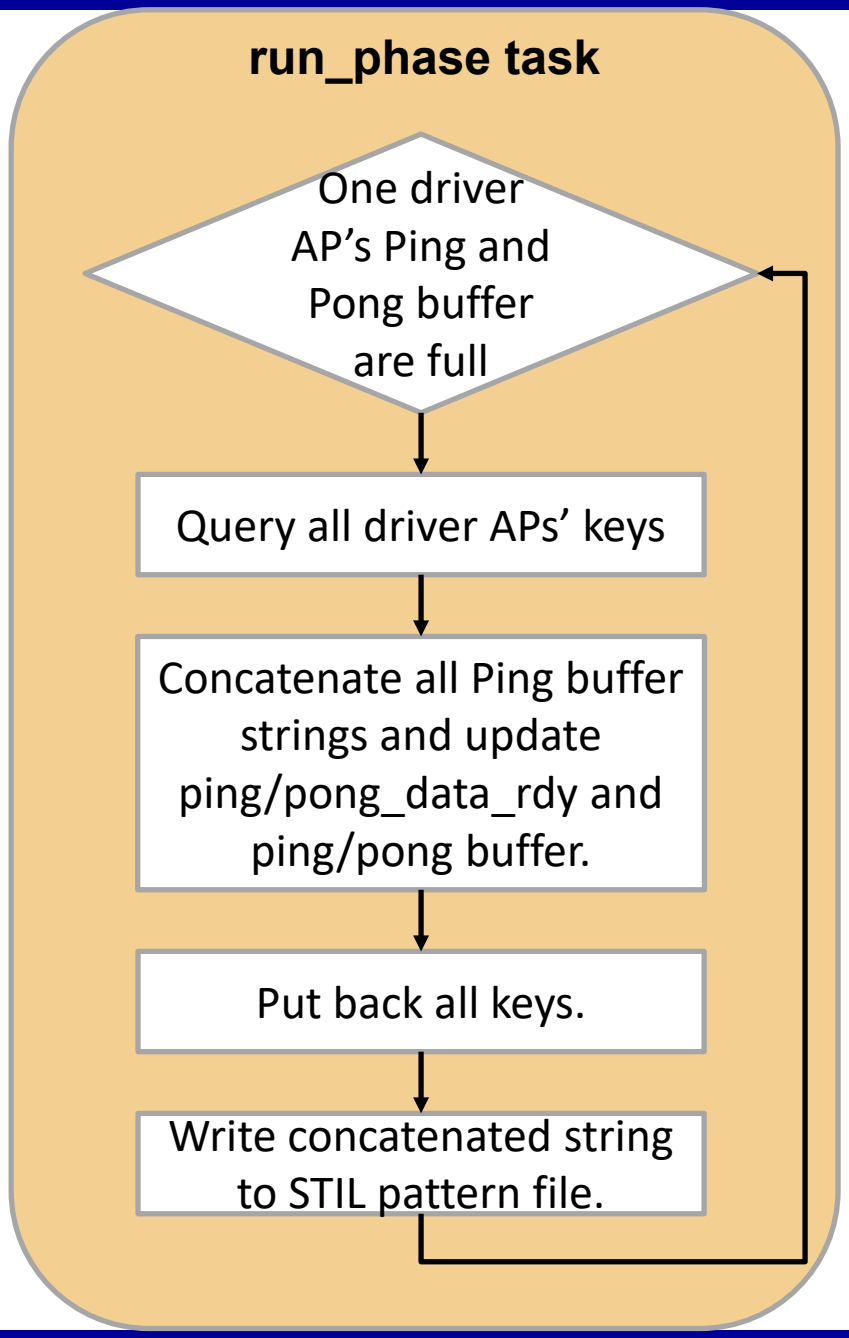
An Example of Clock Pads Connection in Testbench

The only exception: clock_driver only drives TCK, other reference clocks are connected from the testbench.

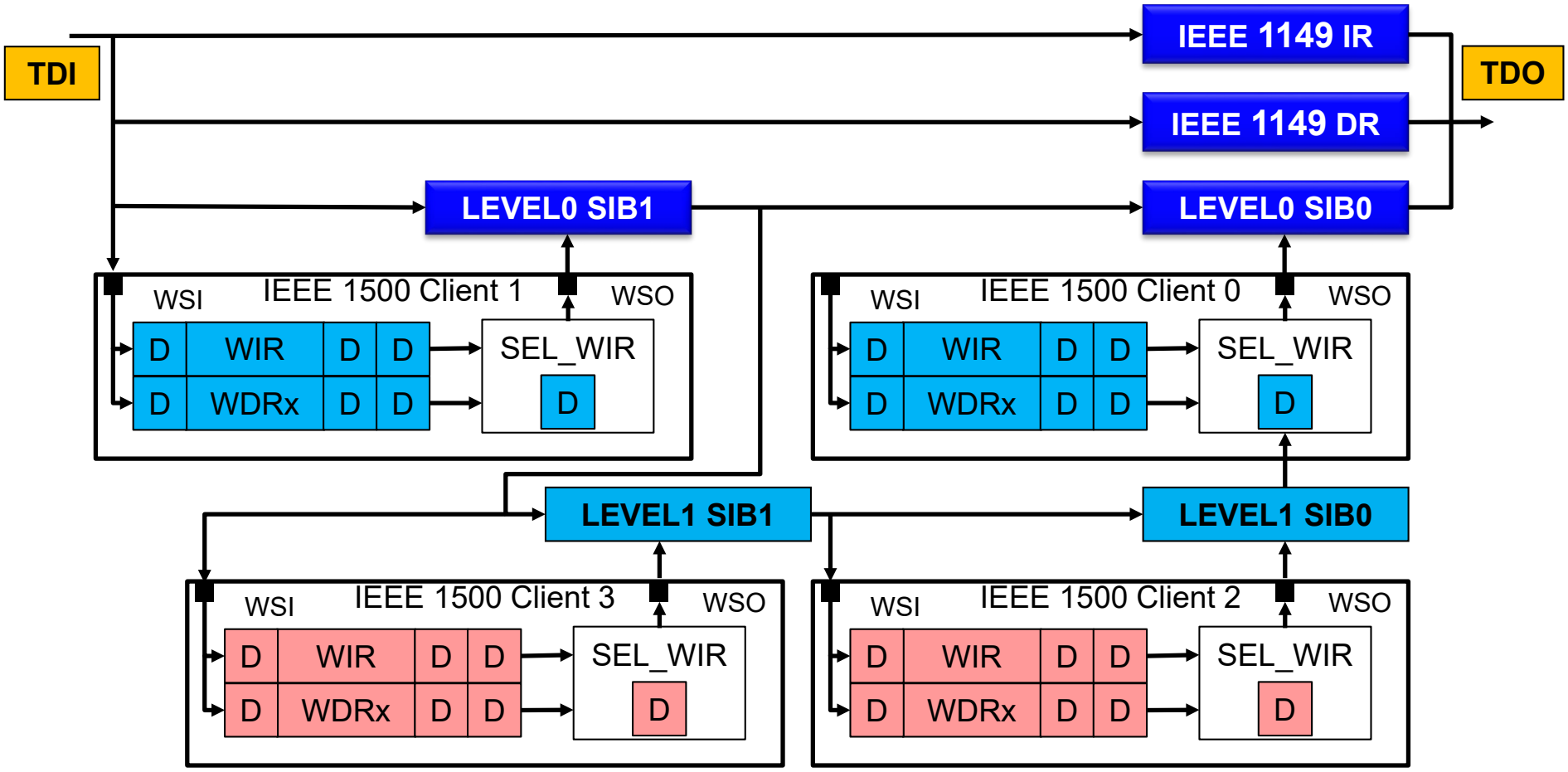


STIL Generator

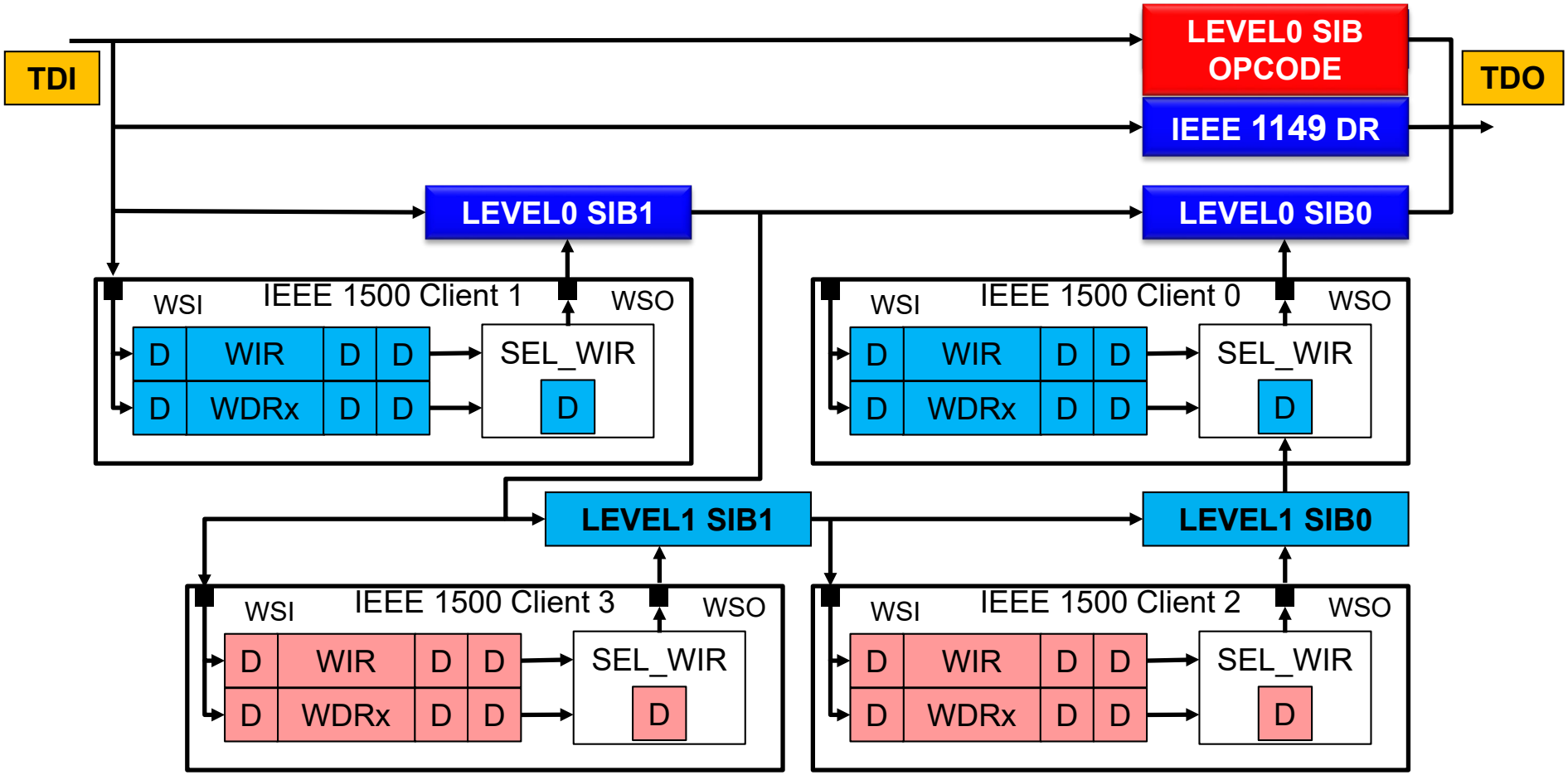




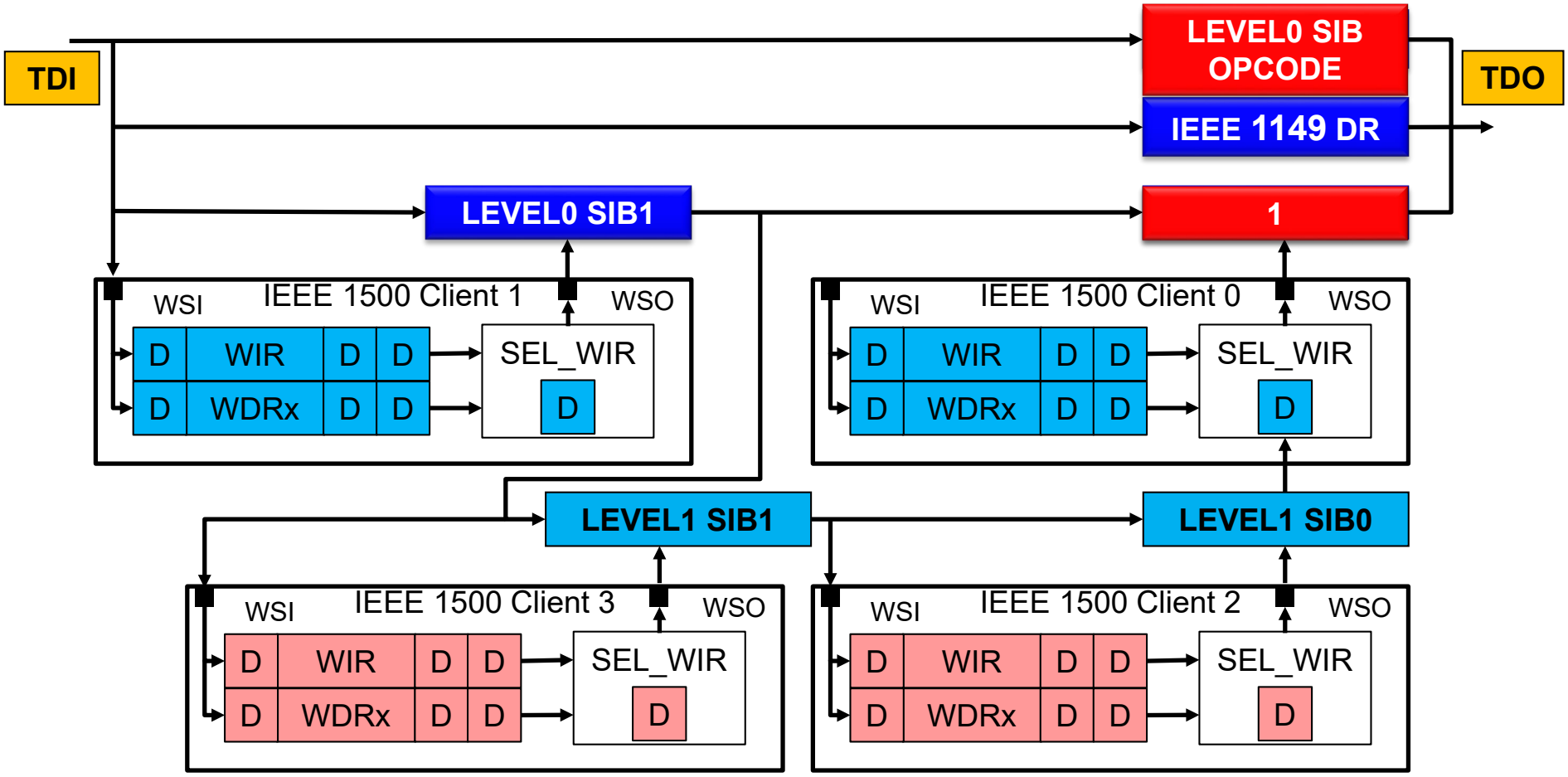
DFT TDR (Test Data Register) Abstraction



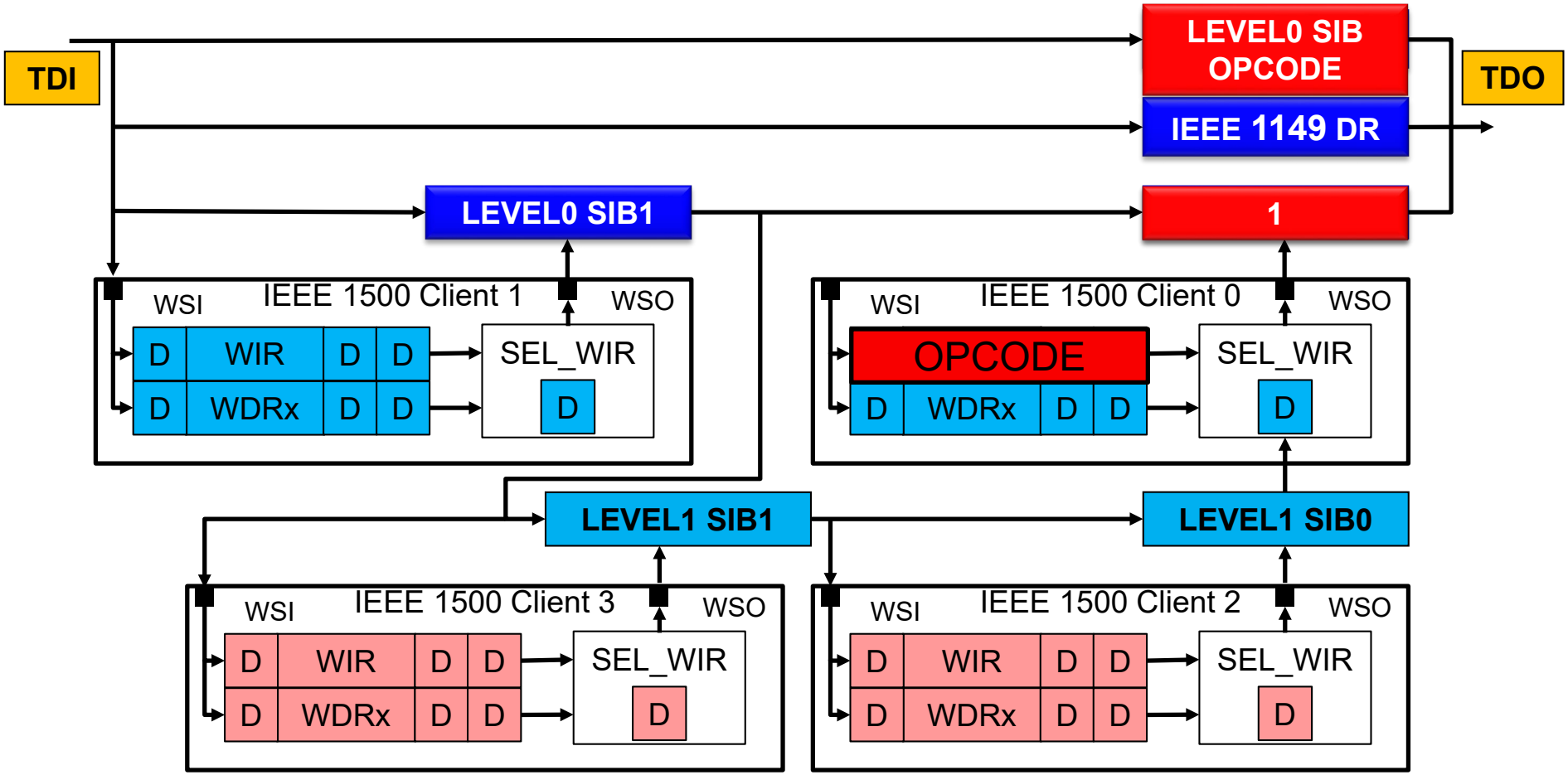
DFT TDR (Test Data Register) Abstraction



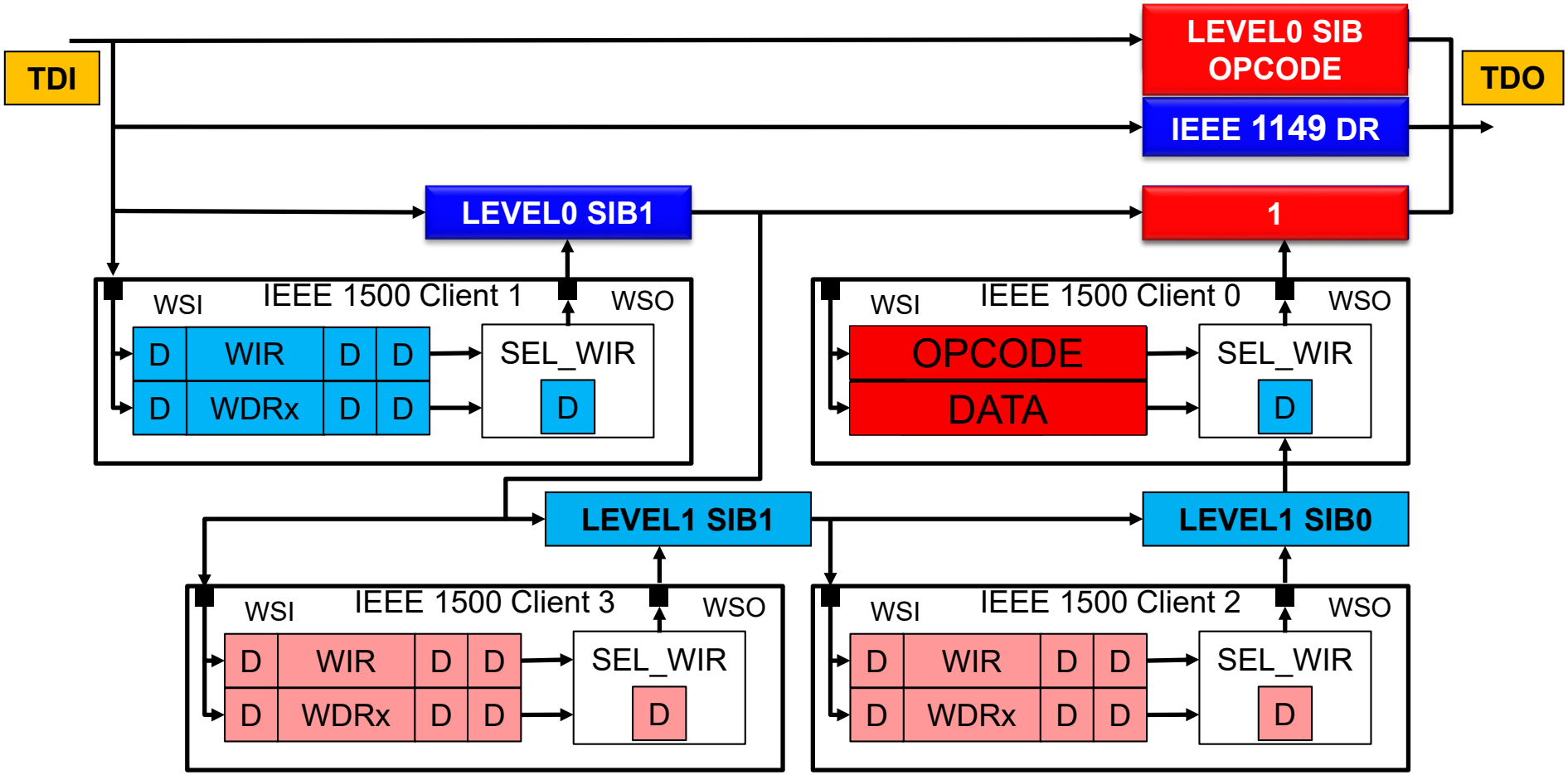
DFT TDR (Test Data Register) Abstraction



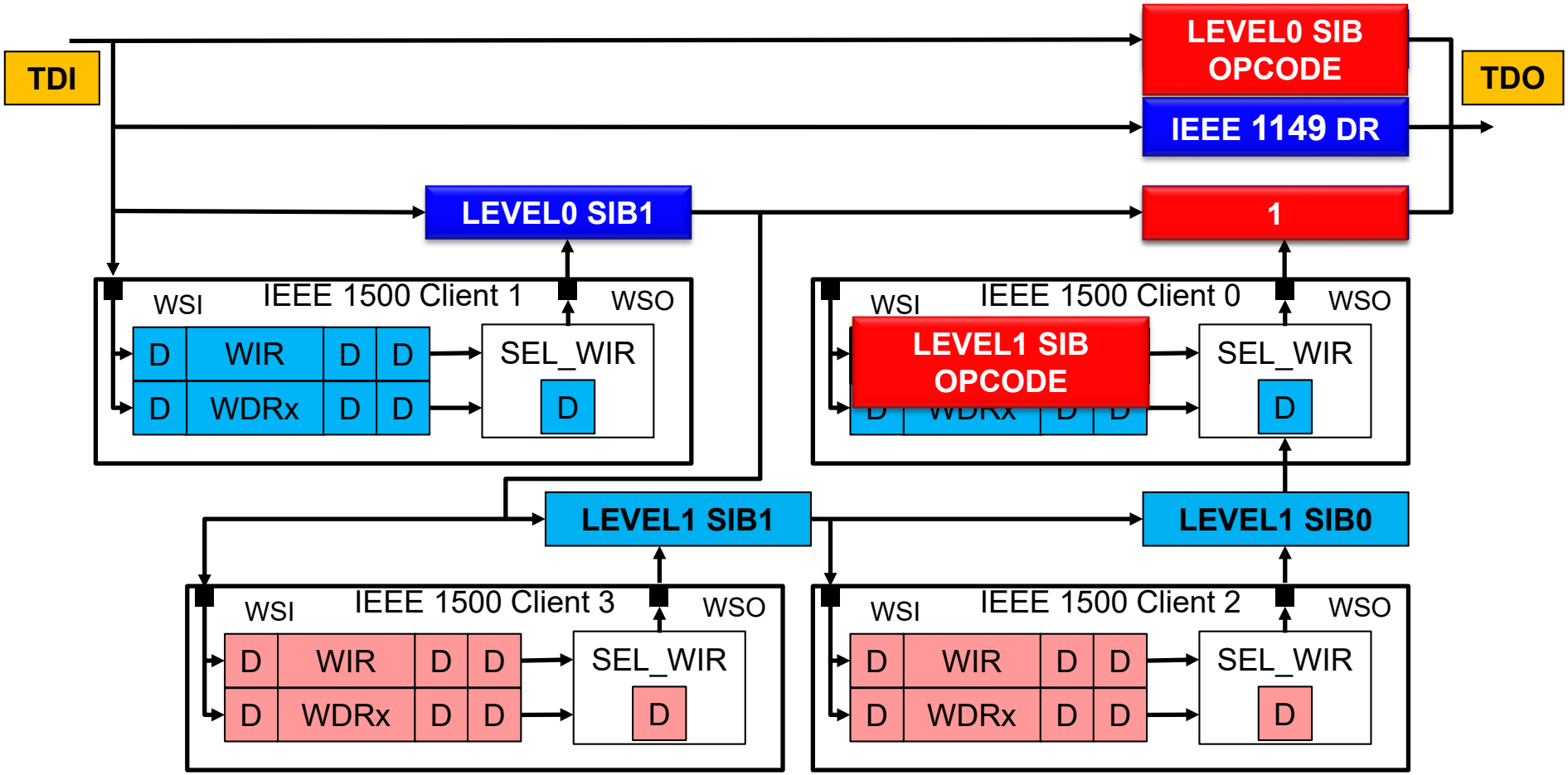
DFT TDR (Test Data Register) Abstraction



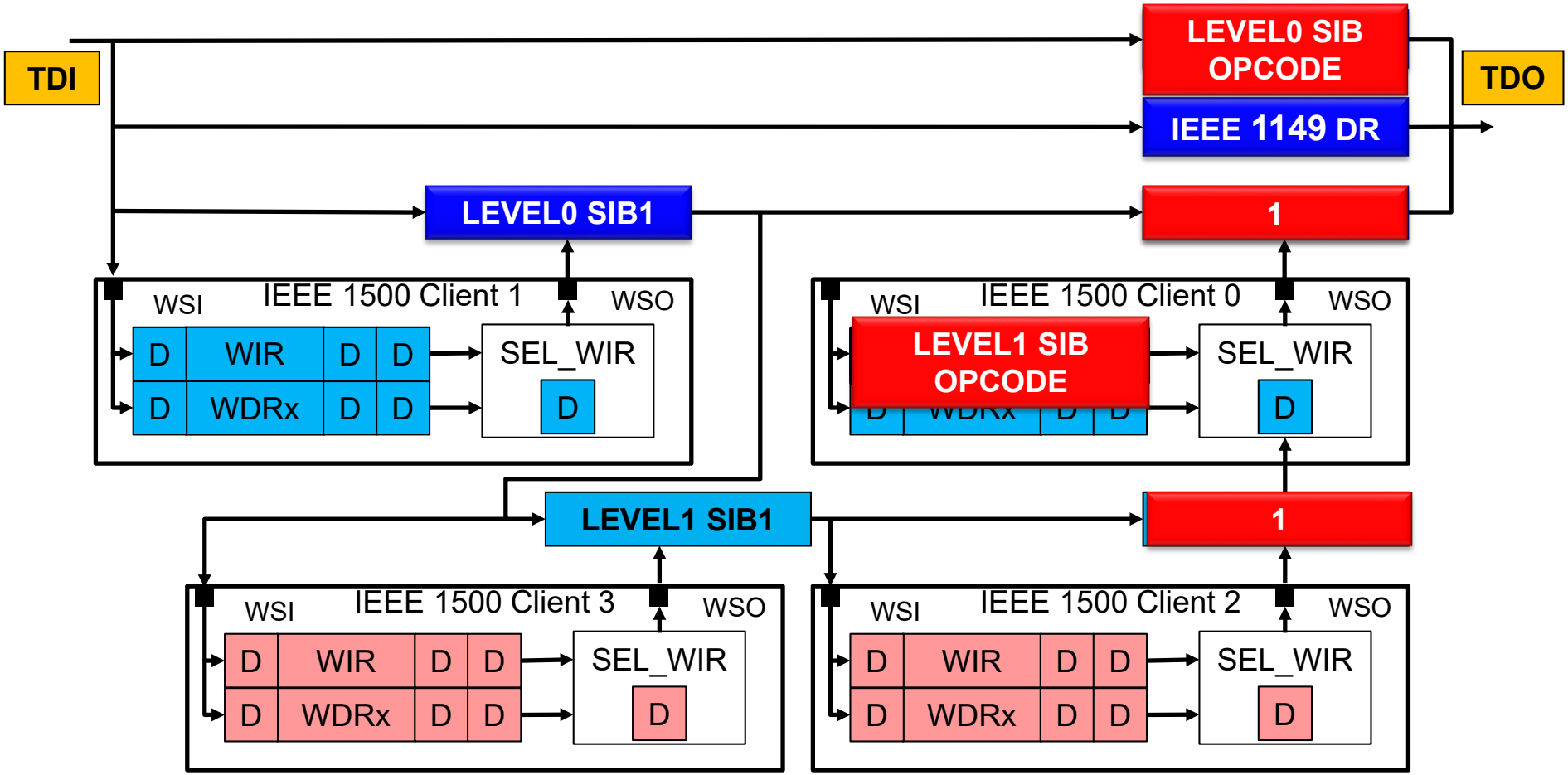
DFT TDR (Test Data Register) Abstraction



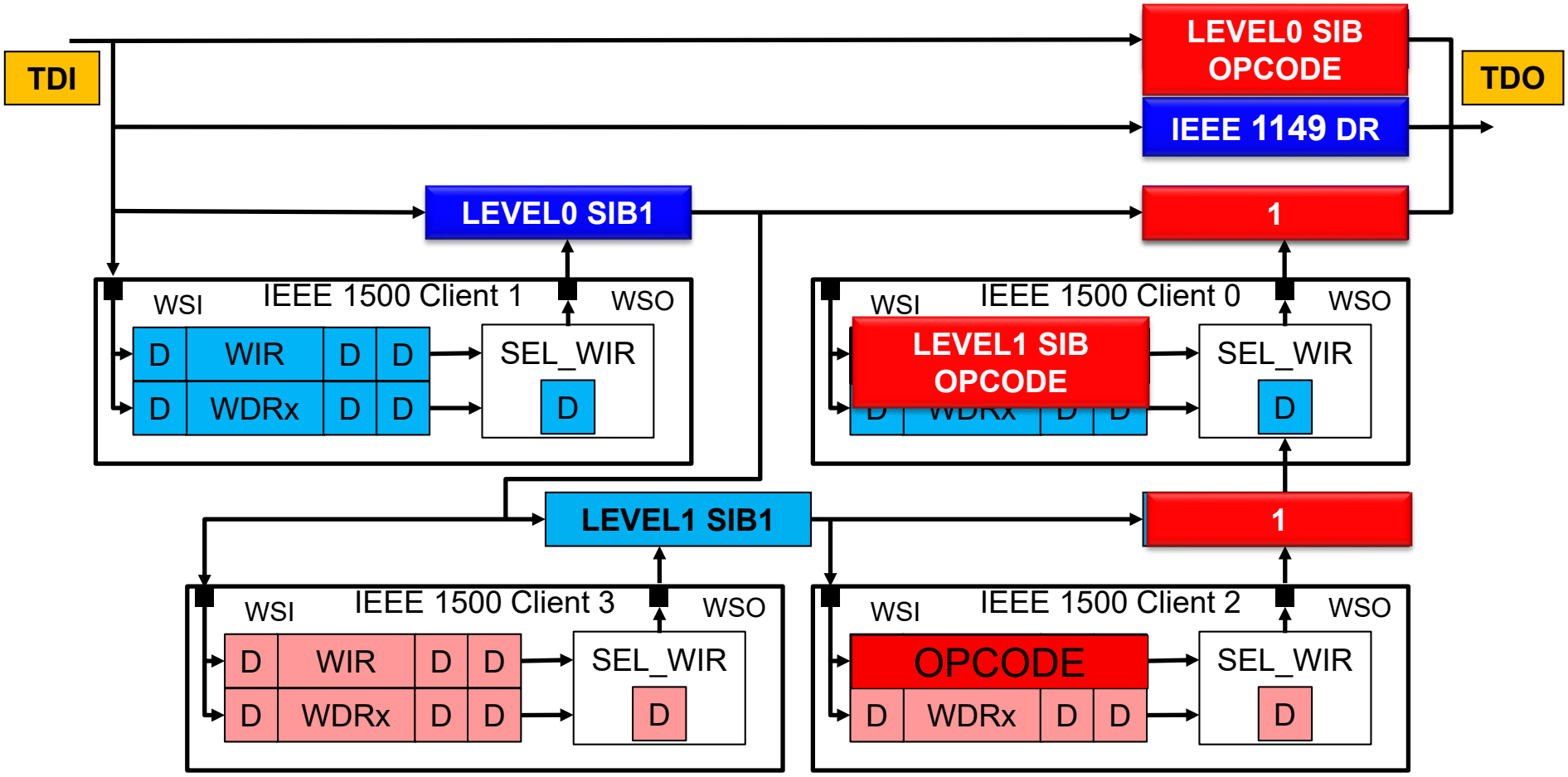
DFT TDR (Test Data Register) Abstraction



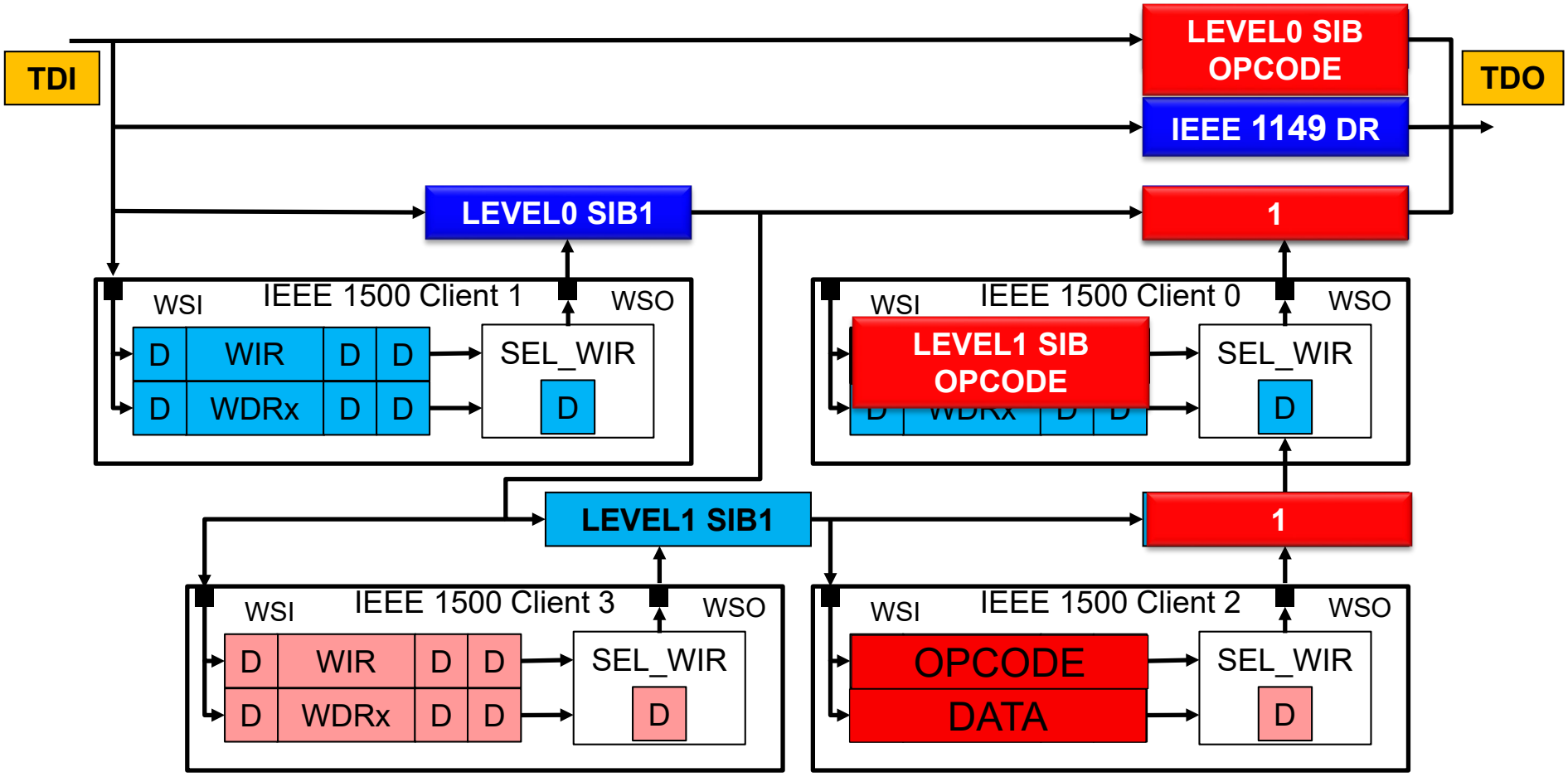
DFT TDR (Test Data Register) Abstraction



DFT TDR (Test Data Register) Abstraction

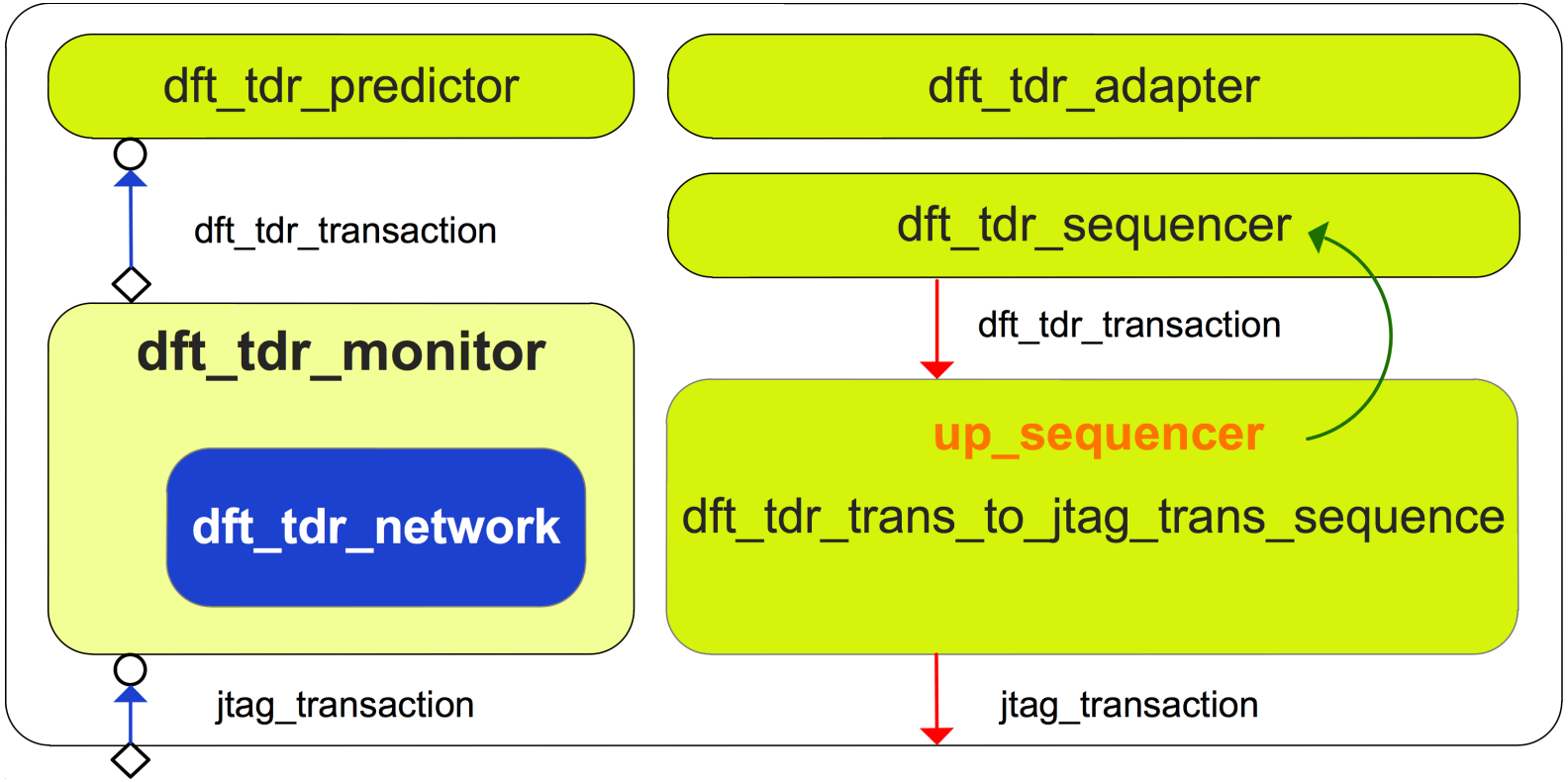


DFT TDR (Test Data Register) Abstraction

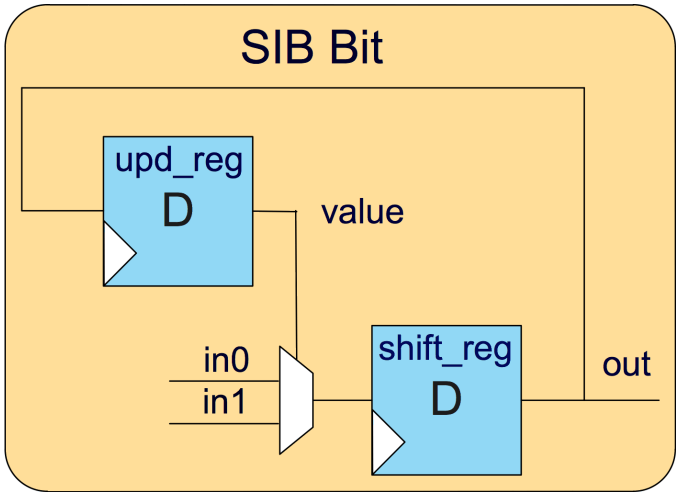


DFT TDR Layer

TDR address encode example



Test Access Network Modelling

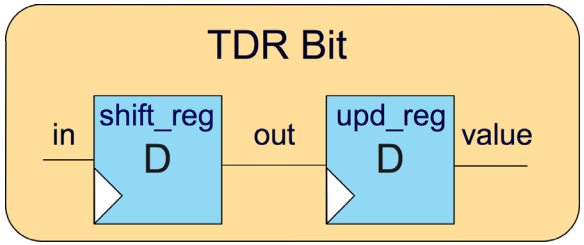



```
class sib_node extends uvm_object;
   `uvm_object_utils(sib_node)
   bit     in0;
   bit     in1;
   bit     value;
   bit     out;
   function new(string name = "sib_node");
     super.new(name);
   endfunction : new
   function void out_update ();
      out = value ? in1 : in0;
   endfunction: out_update
   function void value_update ();
      value = out;
   endfunction: value_update
endclass : sib_node
```

The bit *value* stores update register value and the bit *out* stores shift register value.

The *out_update* function simulates the active clock edge on shift register and the *value_update* function simulates the active clock edge on update register.

Test Access Network Modelling (continued)




```
class reg_node extends uvm_object;
   `uvm_object_utils(reg_node)
   bit    in;
   bit    is_selwir;
   bit    value;
   bit    out;
   function new(string name = "reg_node");
     super.new(name);
   endfunction : new
   function void out_update ();
      out = in;
   endfunction: out_update
   function void value_update ();
      value = out;
   endfunction: value_update
endclass : reg_node
```

Summary

- This UVM-based DFT environment works well in an experiment project and the generated STIL test pattern files pass simulation using STIL Verify.
- The advantages of this environment:
  - It saves usually a team's work to translate DFT function tests to STIL patterns in a project.
  - Avoids errors introduced in the manual translation process.
  - The generated pattern file can be 100% equivalent with simulation.
  - Can be applied to other formats that ATE needs.
- The disadvantage is, the STIL pattern cannot be obtained until the simulation finishes.
  - How to resolve this?

Summary (continued)

- We can add a fake run mode in the environment. In this mode, the DUT is replaced with a shell, which only has input/output pads defined, so that the pattern file can be obtained in minutes.