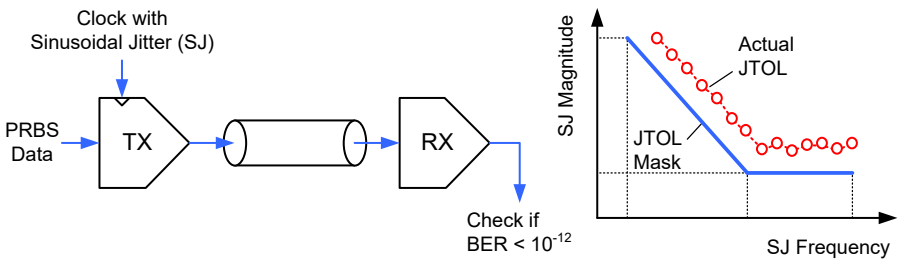
A UVM Reactive Testbench for Jitter Tolerance Measurement of High-Speed Wireline Receivers

Jaeha Kim (jaeha@snu.ac.kr)
Seoul National University, Seoul, Korea

accellera SYSTEMS INITIATIVE
2023 DESIGN AND VERIFICATION™ DVCON CONFERENCE AND EXHIBITION UNITED STATES
SAN JOSE, CA, USA
FEBRUARY 27-MARCH 2, 2023

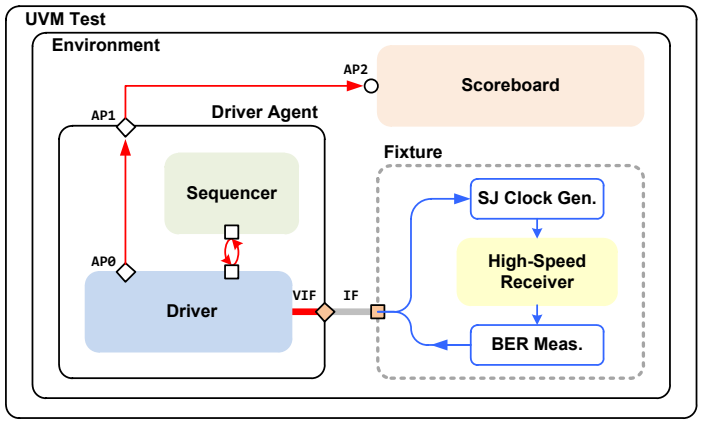
OBJECTIVES

- Jitter tolerance (JTOL) test measures the resilience of a high-speed wireline receiver to the additional sinusoidal jitter (SJ)
- This work demonstrates a UVM testbench performing an iterative search to find the maximum SJ magnitude that can be tolerated for a target BER of 10^{-12} for each SJ frequency



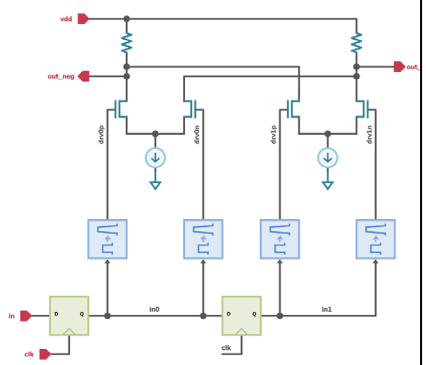
UVM TESTBENCH FOR JTOL MEASUREMENT

- With a well-defined fixture module encapsulating the receiver model and its analog instrumentations, a UVM testbench can be built using standard UVM components



FIXTURE MODULE FOR HIGH-SPEED RECEIVER MODEL

- Using XMODEL primitives, the AMS circuit model and its instrumentations can be described entirely in SystemVerilog
- The XMODEL statistical simulation can simulate BER < 10^{-12}



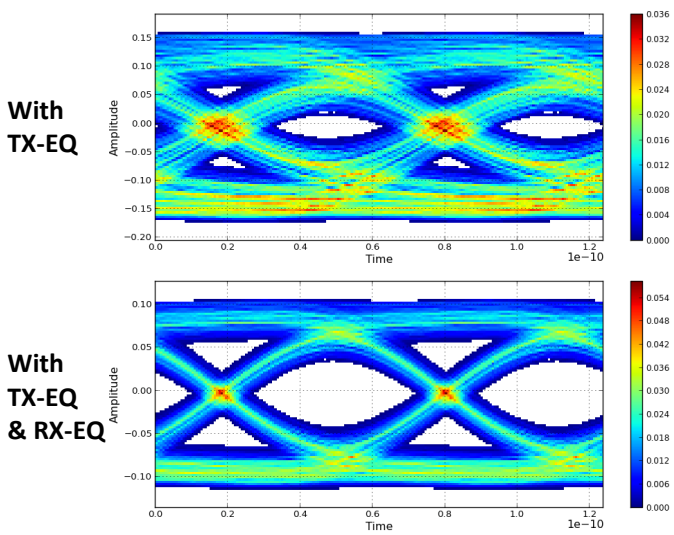

```
module tx_eq (...);

xreal drv0p, drv0n, drv1p, drv1n;
xreal tail0, tail1;
xbit in0, in1;

isource     #(.mode("dc"), .dc(0.00875))
            I0 (.neg(`ground), .pos(tail0), .in(`ground));
isource     #(.mode("dc"), .dc(0.00125))
            I1 (.neg(`ground), .pos(tail1), .in(`ground));
nmosfet     #(.W(2e-05), .L(4e-08), .Kp(2e-05), .Vth(0.5))
            M0 (.g(drv0p), .d(out_neg), .b(`ground), .s(tail0));
nmosfet     #(.W(2e-05), .L(4e-08), .Kp(2e-05), .Vth(0.5))
            M1 (.g(drv0n), .d(out_pos), .b(`ground), .s(tail0));
nmosfet     #(.W(2e-05), .L(4e-08), .Kp(2e-05), .Vth(0.5))
            M2 (.g(drv1p), .d(out_neg), .b(`ground), .s(tail1));
nmosfet     #(.W(2e-05), .L(4e-08), .Kp(2e-05), .Vth(0.5))
            M3 (.g(drv1n), .d(out_pos), .b(`ground), .s(tail1));
resistor    #(.R(50)) R0 (.neg(out_pos), .pos(vdd));
resistor    #(.R(50)) R1 (.neg(out_neg), .pos(vdd));
transition  #(.value0(0.0), .value1(1.0)) XP0 (.out(drv0p), .in(in0));
transition  #(.value0(1.0), .value1(0.0)) XP1 (.out(drv0n), .in(in0));
transition  #(.value0(1.0), .value1(0.0)) XP2 (.out(drv1p), .in(in1));
transition  #(.value0(0.0), .value1(1.0)) XP3 (.out(drv1n), .in(in1));
dff_xbit    XP4 (.clk(clk), .d(in), .q(in0));
dff_xbit    XP5 (.clk(clk), .d(in0), .q(in1));

endmodule   // tx_eq
```

UVM SEQUENCE FOR ITERATIVE SEARCH

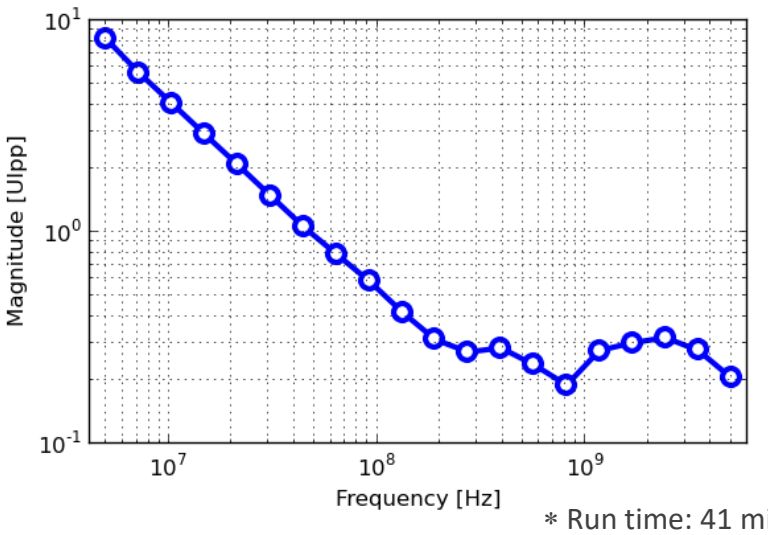
```
class SEQ_JTOL extends uvm_sequence #(PACKET);
    // some details are omitted for brevity ...
    task body();
        PKT = PACKET::type_id::create("PKT");
        for (int i=0; i<freq_numpt; i++) begin:LOOP
            PKT.SJ_freq = (i == 0) ? freq_max : PKT.SJ_freq/freq_ratio;
            PKT.SJ_mag = (i == 0) ? 0.5 : PKT.SJ_mag;
            flag = 0;

            // phase 1: linear search to find the first failing point
            while (1) begin
                start_item(PKT);
                finish_item(PKT);
                get_response(RSP);
                if (RSP.BER < RSP.BER_tol) begin
                    mag_min = PKT.SJ_mag;
                    PKT.SJ_mag += mag_inc;
                    if (flag == -1) break; else flag = 1;
                end
                else begin
                    mag_max = PKT.SJ_mag;
                    PKT.SJ_mag -= mag_inc;
                    if (flag == 1) break; else flag = -1;
                end
            end

            // phase 2: binary search to find the pass/fail boundary
            while (mag_max/mag_min > 1.05) begin
                PKT.SJ_mag = $sqrt(mag_max * mag_min);
                start_item(PKT);
                finish_item(PKT);
                get_response(RSP);
                if (RSP.BER < RSP.BER_tol) mag_min = PKT.SJ_mag;
                else mag_max = PKT.SJ_mag;
            end
        end: LOOP
    endtask: body
endclass: SEQ_JTOL
```

- Sequence component can use **get_response()** method to retrieve the BER result for tried SJ frequency & magnitude
- It finds the maximum SJ meeting BER < 10^{-12} using a combination of linear & binary search

SIMULATED STATISTICAL EYE DIAGRAMS



SIMULATED JITTER TOLERANCE CURVE




```
JITTER TOLERANCE (JTOL)
INDEX     FREQUENCY(Hz)     MAGNITUDE(UIpp)
1         5.0000e+06        8.1488
2         7.1922e+06        5.6295
3         1.0346e+07        4.0418
4         1.4882e+07        2.8870
5         2.1407e+07        2.0728
6         3.0792e+07        1.4806
7         4.4293e+07        1.0576
8         6.3714e+07        0.7851
9         9.1649e+07        0.5858
10        1.3183e+08        0.4184
11        1.8963e+08        0.3106
12        2.7278e+08        0.2709
13        3.9238e+08        0.2834
14        5.6442e+08        0.2362
15        8.1189e+08        0.1894
16        1.1679e+09        0.2756
17        1.6799e+09        0.2996
18        2.4165e+09        0.3168
19        3.4760e+09        0.2763
20        5.0000e+09        0.2051
TOTAL NUMBER OF TRIALS: 106
```

∗ Run time: 41 minutes with 106 BER trials

This work was supported by Samsung Electronics Co., Ltd. and the EDA tools were supported by Scientific Analog, Inc. and IC Design Education Center (IDEC), Korea.

서울대학교 SEOUL NATIONAL UNIVERSITY SAMSUNG scientific analog XMODEL