Bangladesh University of Engineering and Technology
DEPARTMENT OF ELECTRICAL AND ELECTRONIC ENGINEERING

EEE 468 (January 2024)
VLSI Laboratory

Section: G1 Group: 02

16-bit Carry Lookahead Adder with Generate Propagate Logic

Course Instructors:

Nafis Sadik, Lecturer, BUET
Rafid Hassan Palash, Part-time Lecturer, BUET

Signature of Instructor:

Academic Honesty Statement:

IMPORTANT! Please carefully read and sign the Academic Honesty Statement, below. Type the student ID and name, and put your signature. ***You will not receive credit for this project experiment unless this***

"In signing this statement, We hereby certify that the work on this project is our own and that we have not copied the work of any other students (past or present), and cited all relevant sources while completing this project. We understand that if we fail to honor this agreement, We will each receive a score of ZERO for this project and be subject to failure of this course."

Signature: ______________________	**Signature:** ______________________
Full Name: Sheikh Abu Al Raihan	**Full Name: Kazi Abid Hasan**
Student ID: 1906066	**Student ID:1906067**
Signature: ______________________	**Signature:** ______________________
Full Name: Arpa Roy Dastider	**Full Name: Ramisa Tahsin Shreya**
Student ID:1906071	**Student ID: 1906072**

Contents

1. Objectives

- Design and implement a 16-bit Carry Lookahead Adder (CLA) using Generate-Propagate Logic to minimize carry propagation delay and achieve faster addition.
- Optimize critical path delay, power consumption, and area overhead for improved performance compared to traditional ripple carry adders.
- Develop a scalable and modular structure to support the design of higher-bit adders with computational efficiency.
- Conduct thorough simulation and verification using industry-standard VLSI tools to ensure accuracy, functionality, and compliance with design requirements.
- Facilitate seamless integration of the CLA design into larger arithmetic circuits, such as multipliers and ALUs, for digital system applications.

2. Design Steps

- Design Code & Directed Testbench
- Layered Testbench along with Maximum Coverage
- Synthesis
- PnR with DRC
- Optimization of PPA

3. Theory

Theory

The Carry Lookahead Adder (CLA) is a digital circuit designed to perform fast binary addition by overcoming the delay limitations of conventional adders, such as the Ripple Carry Adder. The primary challenge in binary addition is the propagation of carry bits, which increases the computational delay as the number of bits increases. The CLA addresses this challenge by using Generate and Propagate logic to predict the carry at each bit position independently, significantly reducing the delay.

In the CLA, the Generate (G) and Propagate (P) signals are calculated for each bit based on the input bits:

- **Generate (G)**: Indicates if a carry is generated at a specific bit position. It is calculated as $G = A \cdot B$, where A and B are the input bits.
- **Propagate (P)**: Indicates if a carry from a previous bit will propagate through the current bit position. It is calculated as $P = A + B$

Using these signals, the carry for each bit position can be computed directly as: $C_{i+1} = G_i + (P_i \cdot C_i)$. This eliminates the need for the carry to sequentially propagate through all bit positions, as seen in the Ripple Carry Adder. The addition operation is completed by combining the carry signals with the sum logic: $S_i = P_i \oplus C_i$

Design Cosideration

The conventional 16-bit CLA designed with 4 blocks of 4-bit CLAs (chain CLA) introduce a delay of 13 gate level where a block cannot calculate the sum bits before untill the carry out of previous block is calculated. Hence, the circuit is designed in a hierarchical manner to ensure scalability and timing. Groups of bits are processed simultaneously, further reducing the overall delay.

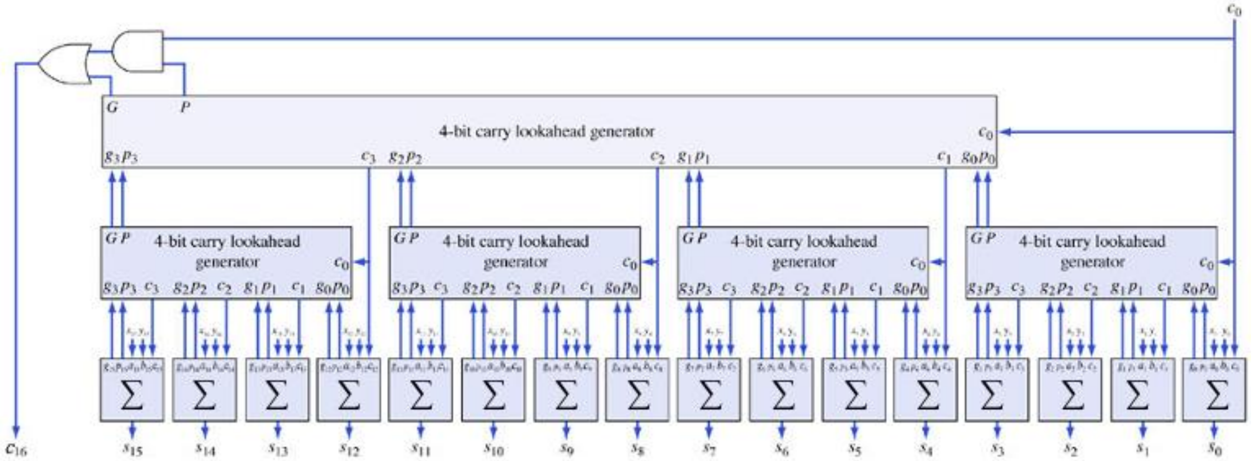
The hierarchical structure of carry lookahead generators is designed to optimize the computation of carries in binary addition. In a 16-bit adder, there are three "floors" of generators:

1. **First Floor (16 Sum Components)**:
   - Each component adds two bits and computes the propagated and generated carry indicators.
   - The result bit is computed once the carry from the previous component is available.
2. **Second Floor (Four 4-bit Carry Lookahead Generators)**:
   - These generators calculate the carries quickly based on the inputs from the previous floor's components.
   - The carries are computed using a two-level logic structure, where the carry propagation and generation indicators from the previous floor determine the carries.
3. **Third Floor (One Carry Lookahead Generator)**:
   - This generator computes the input carries for the second-floor generators and the propagated and generated carry indicators for the entire 16-bit adder.
   - These outputs are used to compute the overall carry of the adder or passed as inputs to higher floors, such as for a 64-bit adder.

The delay of this structure is significantly reduced compared to ripple carry adders and chains of CLA adders. The total delay is 8 gate level (1+2+2+2+1), with each floor contributing a delay of up to 4 gate levels. The delay is proportional to the number of floors, approximately $\log_4 n$ for an n-bit adder, making it much faster.

Design Logic	**Gate level Delay**
Chain CLA	13
Hierarchical CLA	8

Circuit Diagram



4. Step-1 (Design Code & Directed Testbench)

Design Codes

design.sv

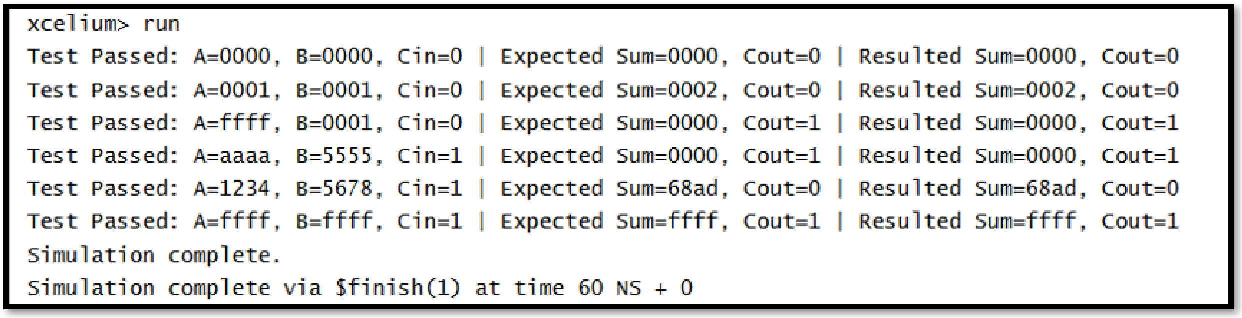
```
module cla16 (
    input clk,
    input [15:0] A,
    input [15:0] B,
    input Cin,
    output reg [15:0] Sum,
    output reg Cout
);

  reg [3:0] P_block, G_block;  // 4-bit Block propagate and generate
  reg [3:0] C_block;           // 4-bit Block carry
  reg [15:0] P, G;             // Bit propagate and generate

  always@(posedge clk) begin
      // Bit Propagate (P) and Generate (G) (1st floor)
              P = A ^ B;
              G = A & B;

      // 4-bit CLA Generators for each block (2nd Floor)
              cla_4bit_generator(P[3:0],G[3:0],P_block[0],G_block[0]);
              cla_4bit_generator(P[7:4],G[7:4],P_block[1],G_block[1]);
              cla_4bit_generator(P[11:8],G[11:8],P_block[2],G_block[2]);
              cla_4bit_generator(P[15:12],G[15:12],P_block[3],G_block[3]);

    // Block and final Carry Logic (3rd floor)
              carry_block_generator(P_block,G_block,Cin,C_block,Cout);

    // Compute carry signals (2nd floor) and sum (1st floor)
              bit_sum(P[3:0],G[3:0],C_block[0],Sum[3:0]);
              bit_sum(P[7:4],G[7:4],C_block[1],Sum[7:4]);
              bit_sum(P[11:8],G[11:8],C_block[2],Sum[11:8]);
              bit_sum(P[15:12],G[15:12],C_block[3],Sum[15:12]);
end

task cla_4bit_generator (
    input [3:0] P,
    input [3:0] G,
    output reg P_block,
    output reg G_block
);
  begin
          P_block = (P[3] & P[2] & P[1] & P[0]);
          G_block = (G[3] | (P[3] & G[2]) | (P[3] & P[2] & G[1]) | (P[3] & P[2] & P[1] & G[0]));
  end
endtask

task carry_block_generator (
    input [3:0] P,
    input [3:0] G,
    input Cin,
    output reg [3:0] C,
    output reg Cout
);
  begin
    // Compute block carries
          C[0] = Cin;
          C[1] = (G[0] | (P[0] & C[0]));
      C[2] = (G[1] | (P[1] & C[1]));
      C[3] = (G[2] | (P[2] & C[2]));

    // Compute final carry-out
          Cout = (G[3] | (P[3] & C[3]));
  end
endtask

task bit_sum (
    input [3:0] P,G,
    input Cin,
    output reg [3:0] Sum
);
```

```
  begin
    Sum[0]=(P[0]^Cin);
    Sum[1]=(P[1]^(G[0]|(P[0]&Cin)));
    Sum[2]=(P[2]^(G[1]|(G[0]&P[1])|(P[0]&P[1]&Cin)));
    Sum[3]=(P[3]^(G[2]|(G[1]&P[2])|(G[0]&P[2]&P[1])|(P[0]&P[1]&P[2]&Cin)));
  end
 endtask
endmodule
```

testbench.sv

```
module cla16_tb;
  reg clk;
  reg [15:0] A, B;
  reg Cin;
  wire [15:0] Sum;
  wire Cout;

  reg [16:0] reference_sum;

       initial begin
   clk=0;
   forever #5 clk=~clk;
  end

  cla16 dut (
    .A(A),
    .B(B),
    .Cin(Cin),
    .clk(clk),
    .Sum(Sum),
    .Cout(Cout)
  );

  initial begin
    test(16'h0000, 16'h0000, 1'b0);
    test(16'h0001, 16'h0001, 1'b0);
    test(16'hFFFF, 16'h0001, 1'b0);
    test(16'hAAAA, 16'h5555, 1'b1);
    test(16'h1234, 16'h5678, 1'b1);
    test(16'hFFFF, 16'hFFFF, 1'b1);
    $display("Simulation complete.");
    $finish;
  end

       initial begin
    $dumpfile("cla16.vcd");
    $dumpvars;
  end
  task test(input [15:0] A_in, B_in, input Cin_in);
    begin
      A = A_in;
      B = B_in;
      Cin = Cin_in;
      #10;
      reference_sum = A + B + Cin;
      if ((Sum!==reference_sum[15:0])||(Cout !== reference_sum[16]))
        $display("Test Failed: A=%h, B=%h, Cin=%b | Expected Sum=%h, Cout=%b | Resulted
Sum=%h, Cout=%b",
                 A, B, Cin, reference_sum[15:0], reference_sum[16], Sum, Cout);
      else
        $display("Test Passed: A=%h, B=%h, Cin=%b | Expected Sum=%h, Cout=%b | Resulted
Sum=%h, Cout=%b",
                 A, B, Cin, reference_sum[15:0], reference_sum[16], Sum, Cout);
    end
  endtask
endmodule
```

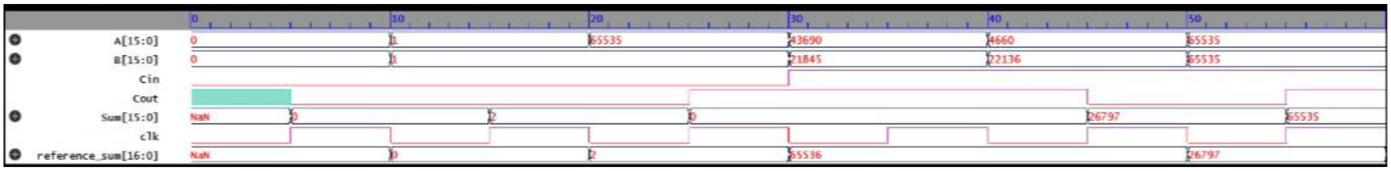
Result



```
xcelium> run
Test Passed: A=0000, B=0000, Cin=0 | Expected Sum=0000, Cout=0 | Resulted Sum=0000, Cout=0
Test Passed: A=0001, B=0001, Cin=0 | Expected Sum=0002, Cout=0 | Resulted Sum=0002, Cout=0
Test Passed: A=ffff, B=0001, Cin=0 | Expected Sum=0000, Cout=1 | Resulted Sum=0000, Cout=1
Test Passed: A=aaaa, B=5555, Cin=1 | Expected Sum=0000, Cout=1 | Resulted Sum=0000, Cout=1
Test Passed: A=1234, B=5678, Cin=1 | Expected Sum=68ad, Cout=0 | Resulted Sum=68ad, Cout=0
Test Passed: A=ffff, B=ffff, Cin=1 | Expected Sum=ffff, Cout=1 | Resulted Sum=ffff, Cout=1
Simulation complete.
Simulation complete via $finish(1) at time 60 NS + 0
```

All test cases passed successfully.

Waveform



5. Step-2 (Layered Testbench)

Design Codes

environment.sv

```
`include "generator.sv"
`include "driver.sv"
`include "monitor.sv"
`include "scoreboard.sv"
class environment;

  mailbox gen2driv;
  mailbox driv2sb;
  mailbox mon2sb;

  generator gen;
  driver drv;
  monitor mon;
  scoreboard scb;

  event driven;
  virtual cla_if claif;
  function new (virtual cla_if claif);
    this.claif=claif;
    gen2driv=new();
    driv2sb=new();
    mon2sb=new();
    gen=new(gen2driv);
    drv=new(gen2driv, driv2sb, claif.DRIVER, driven);
    mon=new (mon2sb, claif.MONITOR, driven);
             scb=new(driv2sb,mon2sb);
  endfunction

  task main(input int count);
    fork gen.main(count);
              drv.main(count);
         mon.main(count);
         scb.main(count);
    join
    $finish;
  endtask:main

endclass: environment
```

interface.sv

```
interface cla_if (input clk);
  logic [15:0] a,b;
  logic cin;
  logic [15:0] sum;
  logic cout;

  clocking driver_cb @(negedge clk); default input #1 output #1;
            output a, b,cin;
  endclocking

  clocking mon_cb @(negedge clk); default input #1 output #1;
    input a,b,cin;
    input sum,cout;
  endclocking

  modport DRIVER (clocking driver_cb, input clk);
  modport MONITOR (clocking mon_cb, input clk);
endinterface
```

generator.sv

```
`include "transaction.sv"
class generator;
mailbox gen2driv;
transaction g_trans, custom_trans;
function new (mailbox gen2driv);
this.gen2driv=gen2driv;
endfunction
task main(input int count);
repeat(count) begin
g_trans=new();
g_trans=new custom_trans;
assert(g_trans.randomize());
gen2driv.put(g_trans);
end
endtask: main
endclass:generator
```

transaction.sv

```
class transaction;
  rand bit [15:0] a;
  rand bit [15:0] b;
  rand bit cin;
  bit [15:0] sum;
  bit cout;
endclass:transaction
```

driver.sv

```
class driver;
  mailbox gen2driv, driv2sb;
  virtual cla_if.DRIVER claif;
  transaction d_trans;
  event driven;

  function new (mailbox gen2driv, driv2sb, virtual cla_if.DRIVER claif, event driven);
    this.gen2driv=gen2driv;
    this.claif=claif;
    this.driven=driven;
    this.driv2sb=driv2sb;
  endfunction

  task main(input int count);
  repeat(count) begin
    d_trans=new();
    gen2driv.get(d_trans);
    @(claif.driver_cb);
    claif.driver_cb.a <= d_trans.a;
    claif.driver_cb.b <= d_trans.b;
    claif.driver_cb.cin <= d_trans.cin;
    driv2sb.put(d_trans);
    -> driven;
  end
  endtask:main
endclass:driver
```

monitor.sv

```
class monitor;

  mailbox mon2sb;
  virtual cla_if.MONITOR claif;
  transaction m_trans;
  event driven;
  function new(mailbox mon2sb, virtual cla_if.MONITOR claif, event driven);
    this.mon2sb=mon2sb;
    this.claif=claif;
    this.driven=driven;
  endfunction

  task main(input int count);
    @(driven);
    @(claif.mon_cb); repeat(count) begin
    m_trans=new();
    @(posedge claif.clk);
    m_trans.cout=claif.mon_cb.cout;
    m_trans.sum=claif.mon_cb.sum;
    mon2sb.put(m_trans);
  end
endtask:main
endclass: monitor
```

scoreboard.sv

```
class scoreboard;

  mailbox driv2sb;
  mailbox mon2sb;

  transaction d_trans;
  transaction m_trans;
  event driven;

  function new(mailbox driv2sb, mon2sb);
    this.driv2sb=driv2sb;
    this.mon2sb=mon2sb;
  endfunction

  task main(input int count);
    $display("--Scoreboard Test Starts- -");
    repeat(count) begin
    m_trans=new();
    mon2sb.get(m_trans);
    report();
    if((m_trans.sum != d_trans.sum)||(m_trans.cout != d_trans.cout))
          $display("Failed : a=%d b=%d cin=%d  Expected sum=%d  Resulted sum=%d Expected
cout=%d  Resulted cout=%d",d_trans.a,d_trans.b, d_trans.cin,d_trans.sum, m_trans.sum,d_trans.cout,
m_trans.cout);
    else
      $display("Passed : a=%d b=%d cin=%d  Expected sum=%d  Resulted sum=%d Expected
cout=%d  Resulted cout=%d",d_trans.a,d_trans.b, d_trans.cin,d_trans.sum, m_trans.sum,d_trans.cout,
m_trans.cout);
    end
    $display("--Scoreboard Test Ends---");
  endtask:main

  task report();
    d_trans=new();
    driv2sb.get(d_trans);
    {d_trans.cout,d_trans.sum}=d_trans.a+d_trans.b+d_trans.cin;
  endtask: report

endclass:scoreboard
```

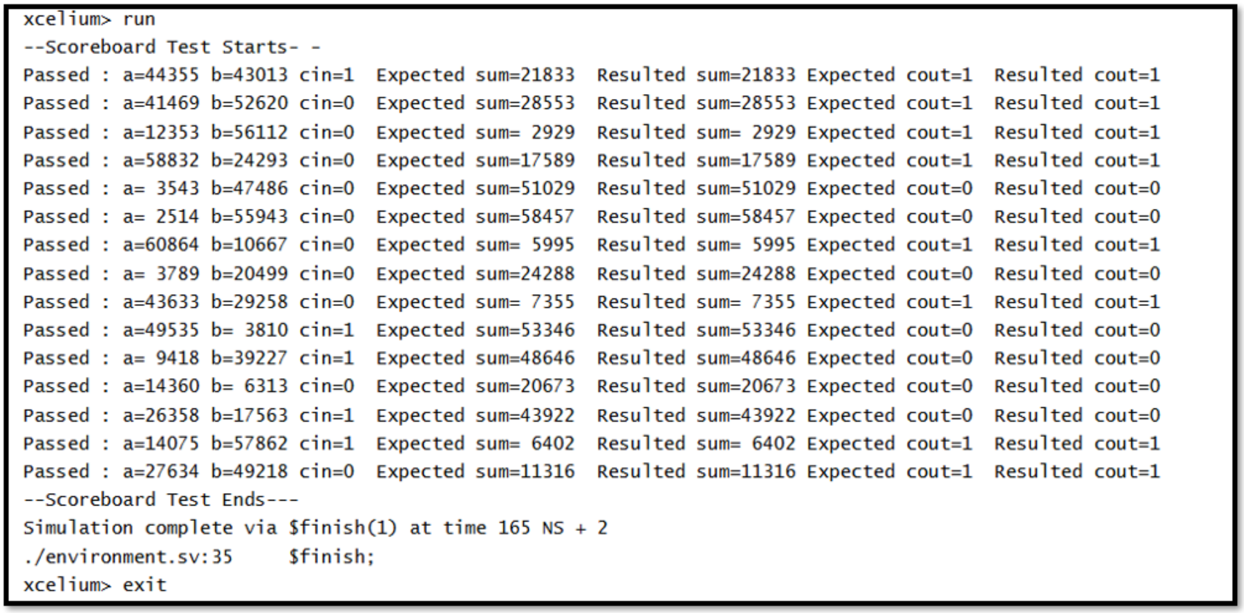
testcases.sv

```
`include "environment.sv"

program test(input int count, cla_if claif);
  environment env;

  class testcase01 extends transaction;
    //constraint c_s {
      //s inside {[0:1], [14:15]};
      //s inside {[0:15]};
    //}
  endclass:testcase01

  initial begin
    testcase01 testcase01handle;
    testcase01handle=new();

    env=new(claif);
    env.gen.custom_trans=testcase01handle;
    env.main(count);
  end

endprogram:test
```

testbench.sv

```
`include "testcase01.sv"
//`include "test.sv"
`include "interface.sv"

module testbench;
  bit clk;


  initial begin
    forever #5 clk =~clk;
  end

  int count=15;
  cla_if claif(clk);

  test test01(count,claif);

  initial begin
    $dumpfile("dump.vcd");
    $dumpvars;
  end

  cla16 DUT (
    .A(claif.a),
    .B(claif.b),
    .Cin(claif.cin),
    .Sum(claif.sum),
    .Cout(claif.cout),
    .clk(clk)
  );

Endmodule
```

Result



```
xcelium> run
--Scoreboard Test Starts- -
Passed : a=44355 b=43013 cin=1  Expected sum=21833  Resulted sum=21833 Expected cout=1  Resulted cout=1
Passed : a=41469 b=52620 cin=0  Expected sum=28553  Resulted sum=28553 Expected cout=1  Resulted cout=1
Passed : a=12353 b=56112 cin=0  Expected sum= 2929  Resulted sum= 2929 Expected cout=1  Resulted cout=1
Passed : a=58832 b=24293 cin=0  Expected sum=17589  Resulted sum=17589 Expected cout=1  Resulted cout=1
Passed : a= 3543 b=47486 cin=0  Expected sum=51029  Resulted sum=51029 Expected cout=0  Resulted cout=0
Passed : a= 2514 b=55943 cin=0  Expected sum=58457  Resulted sum=58457 Expected cout=0  Resulted cout=0
Passed : a=60864 b=10667 cin=0  Expected sum= 5995  Resulted sum= 5995 Expected cout=1  Resulted cout=1
Passed : a= 3789 b=20499 cin=0  Expected sum=24288  Resulted sum=24288 Expected cout=0  Resulted cout=0
Passed : a=43633 b=29258 cin=0  Expected sum= 7355  Resulted sum= 7355 Expected cout=1  Resulted cout=1
Passed : a=49535 b= 3810 cin=1  Expected sum=53346  Resulted sum=53346 Expected cout=0  Resulted cout=0
Passed : a= 9418 b=39227 cin=1  Expected sum=48646  Resulted sum=48646 Expected cout=0  Resulted cout=0
Passed : a=14360 b= 6313 cin=0  Expected sum=20673  Resulted sum=20673 Expected cout=0  Resulted cout=0
Passed : a=26358 b=17563 cin=1  Expected sum=43922  Resulted sum=43922 Expected cout=0  Resulted cout=0
Passed : a=14075 b=57862 cin=1  Expected sum= 6402  Resulted sum= 6402 Expected cout=1  Resulted cout=1
Passed : a=27634 b=49218 cin=0  Expected sum=11316  Resulted sum=11316 Expected cout=1  Resulted cout=1
--Scoreboard Test Ends---
Simulation complete via $finish(1) at time 165 NS + 2
./environment.sv:35     $finish;
xcelium> exit
```

All random test cases passed successfully.

6. Step-3 (Layered Testbench with maximum coverage)

Design Codes

testbench.sv

```
`include "testcase01.sv"
//`include "test.sv"
`include "interface.sv"

module testbench;
  bit clk;


  initial begin
    forever #5 clk =~clk;
  end

  int count=500;
  cla_if claif(clk);

  test test01(count,claif);
  initial begin
    $dumpfile("dump.vcd");
    $dumpvars;
    #10000;
    $finish;
  end

  cla16 DUT (
    .A(claif.a),
    .B(claif.b),
    .Cin(claif.cin),
    .Sum(claif.sum),
    .Cout(claif.cout),
    .clk(clk)
  );

Endmodule
```

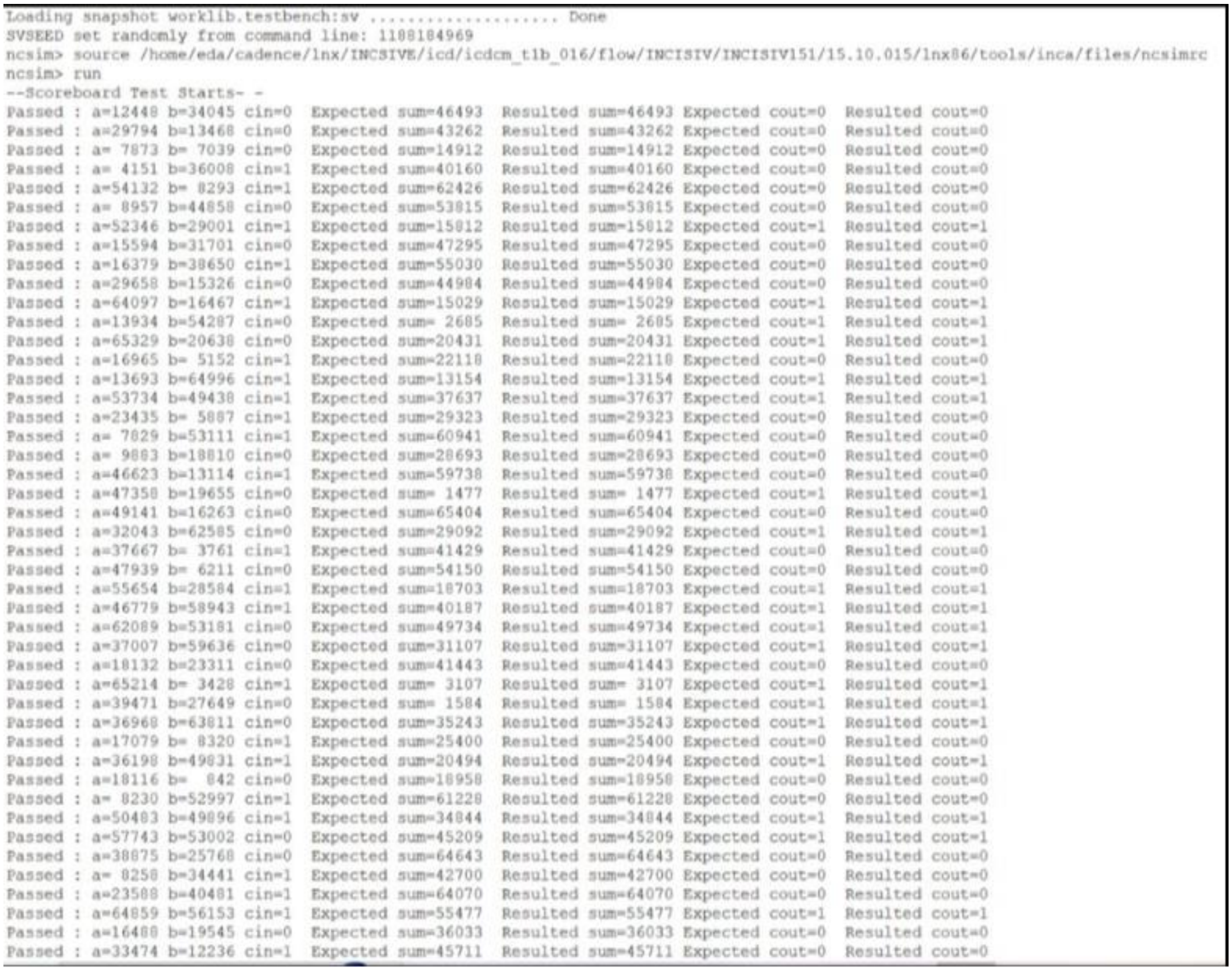
testcase.sv

```
`include "environment.sv"

program test(input int count, cla_if claif);
  environment env;

  class testcase01 extends transaction;
    constraint c_s {
      a inside {[0:65535]};
      b inside {[0:65535]};
      cin inside {[0:1]};
    }



  endclass:testcase01

  initial begin
    testcase01 testcase01handle;
    testcase01handle=new();

    env=new(claif);
    env.gen.custom_trans=testcase01handle;
    env.main(count);
  end

endprogram:test
```

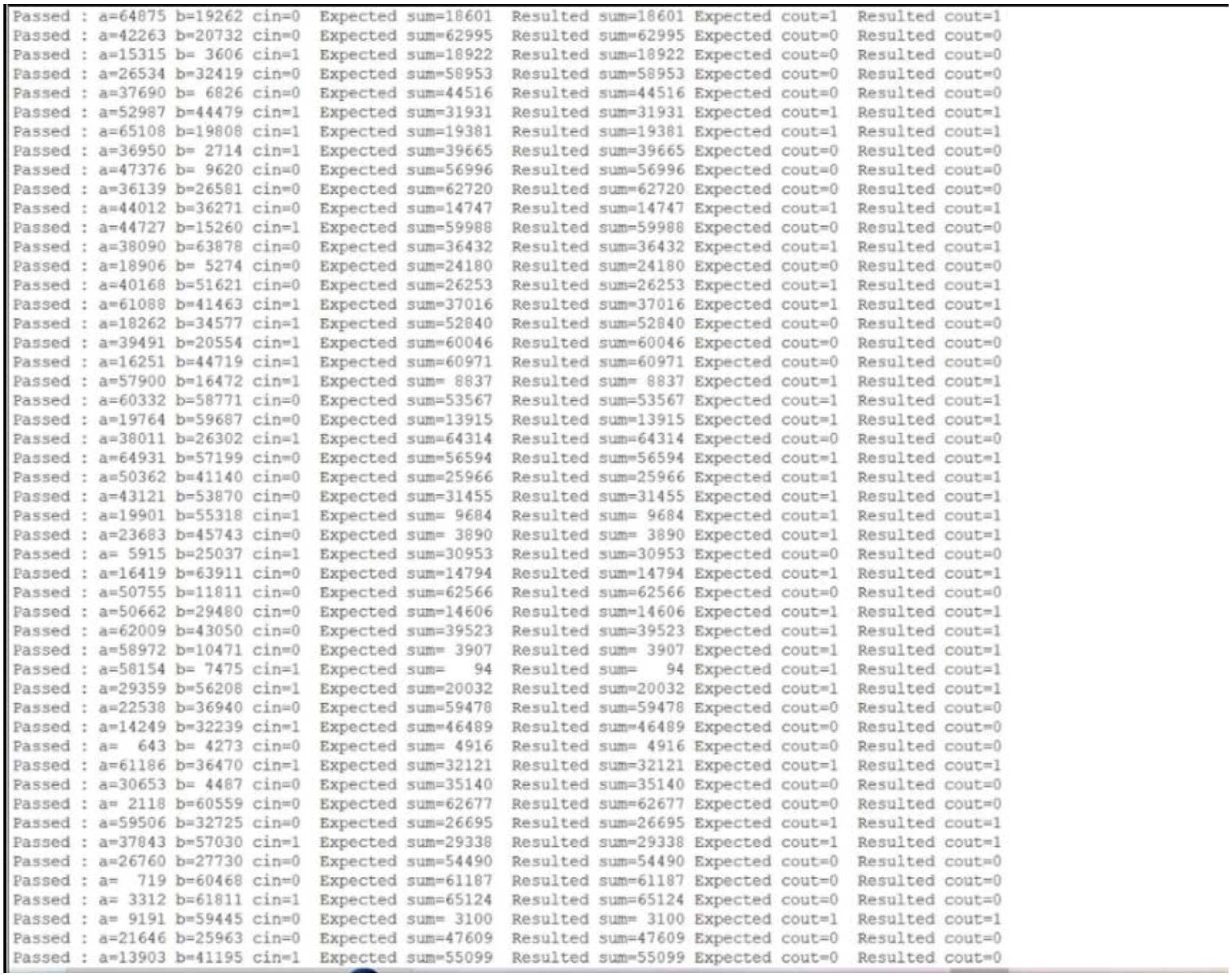
Scoreboard.sv

```
class scoreboard;

  mailbox driv2sb;
  mailbox mon2sb;

  transaction d_trans;
  transaction m_trans;

  logic V;

  real Flags [4] = '{default:64'b0};
  real Pass [4] = '{default:64'b0};
  real Total_Pass;
  real Fail [4] = '{default:64'b0};
  real pc [4], f_pc [4], p_pc [4];
  logic [1:0] X,Y;


  event driven;

  function new(mailbox driv2sb, mon2sb);
    this.driv2sb=driv2sb;
    this.mon2sb=mon2sb;
  endfunction

  task main(input int count);
    $display("--Scoreboard Test Starts- -");
    repeat(count) begin
    m_trans=new();
    mon2sb.get(m_trans);
    report();

    Y = {d_trans.cout, V};
    if((m_trans.sum != d_trans.sum)||(m_trans.cout != d_trans.cout))
      begin
            $display("Failed : a=%d b=%d cin=%d  Expected sum=%d  Resulted sum=%d Expected
cout=%d  Resulted cout=%d",d_trans.a,d_trans.b, d_trans.cin,d_trans.sum, m_trans.sum,d_trans.cout,
m_trans.cout);
            if(Y==2'b00)
         Fail[0] = Fail[0]+1;
            else if(Y==2'b01)
         Fail[1] = Fail[1]+1;
            else if(Y==2'b10)
         Fail[2] = Fail[2]+1;
            else if(Y==2'b11)
         Fail[3] = Fail[3]+1;
      end
    else
      begin
        $display("Passed : a=%d b=%d cin=%d  Expected sum=%d  Resulted sum=%d Expected
cout=%d  Resulted cout=%d",d_trans.a,d_trans.b, d_trans.cin,d_trans.sum, m_trans.sum,d_trans.cout,
m_trans.cout);
        if(Y==2'b00)
          Pass[0] = Pass[0]+1;
        else if(Y==2'b01)
          Pass[1] = Pass[1]+1;
        else if(Y==2'b10)
          Pass[2] = Pass[2]+1;
        else if(Y==2'b11)
          Pass[3] = Pass[3]+1;
      end
    end

    pc[0]=(Flags[0]*100)/count;
    pc[1]=(Flags[1]*100)/count;
    pc[2]=(Flags[2]*100)/count;
    pc[3]=(Flags[3]*100)/count;


    p_pc[0]=(Pass[0]*100)/Flags[0];
    p_pc[1]=(Pass[1]*100)/Flags[1];
    p_pc[2]=(Pass[2]*100)/Flags[2];
    p_pc[3]=(Pass[3]*100)/Flags[3];


    f_pc[0]=(Fail[0]*100)/Flags[0];
    f_pc[1]=(Fail[1]*100)/Flags[1];
    f_pc[2]=(Fail[2]*100)/Flags[2];
    f_pc[3]=(Fail[3]*100)/Flags[3];


    Total_Pass=((Pass[0]+Pass[1]+Pass[2]+Pass[3])*100)/count;


    //$display("--Scoreboard Test Ends---");

    $display("\n\n-------------------Coverage Results---------------------\n\n");

    $display("# of Case Tested for Type 1: {CV}='00'=%0.0f cases with percentage=%f, Pass rate=%f,
Fail rate=%f",Flags[0],pc[0],p_pc[0],f_pc[0]);
    $display("# of Case Tested for Type 2: {CV}='01'=%0.0f cases with percentage=%f, Pass rate=%f,
Fail rate=%f",Flags[1],pc[1],p_pc[1],f_pc[1]);
    $display("# of Case Tested for Type 3: {CV}='10'=%0.0f cases with percentage=%f, Pass rate=%f,
Fail rate=%f",Flags[2],pc[2],p_pc[2],f_pc[2]);
    $display("# of Case Tested for Type 4: {CV}='11'=%0.0f cases with percentage=%f, Pass rate=%f,
Fail rate=%f",Flags[3],pc[3],p_pc[3],f_pc[3]);
    $display("Total Coverage pass rate = %0.2f %%\n",Total_Pass);

    $display("\n\n------------------Coverage Results End--------------------\n\n");

  endtask:main

  task report();
    d_trans=new();
    driv2sb.get(d_trans);
    {d_trans.cout,d_trans.sum}=d_trans.a+d_trans.b+d_trans.cin;

    if(d_trans.cout == 1)
      V = 1'b1;
    else
      V = 1'b0;

    X = {d_trans.cout, V};

    if(X==2'b00)
      Flags[0] = Flags[0]+1;
    else if(X==2'b01)
      Flags[1] = Flags[1]+1;
    else if(X==2'b10)
      Flags[2] = Flags[2]+1;
    else if(X==2'b11)
      Flags[3] = Flags[3]+1;

  endtask: report

endclass:scoreboard
```

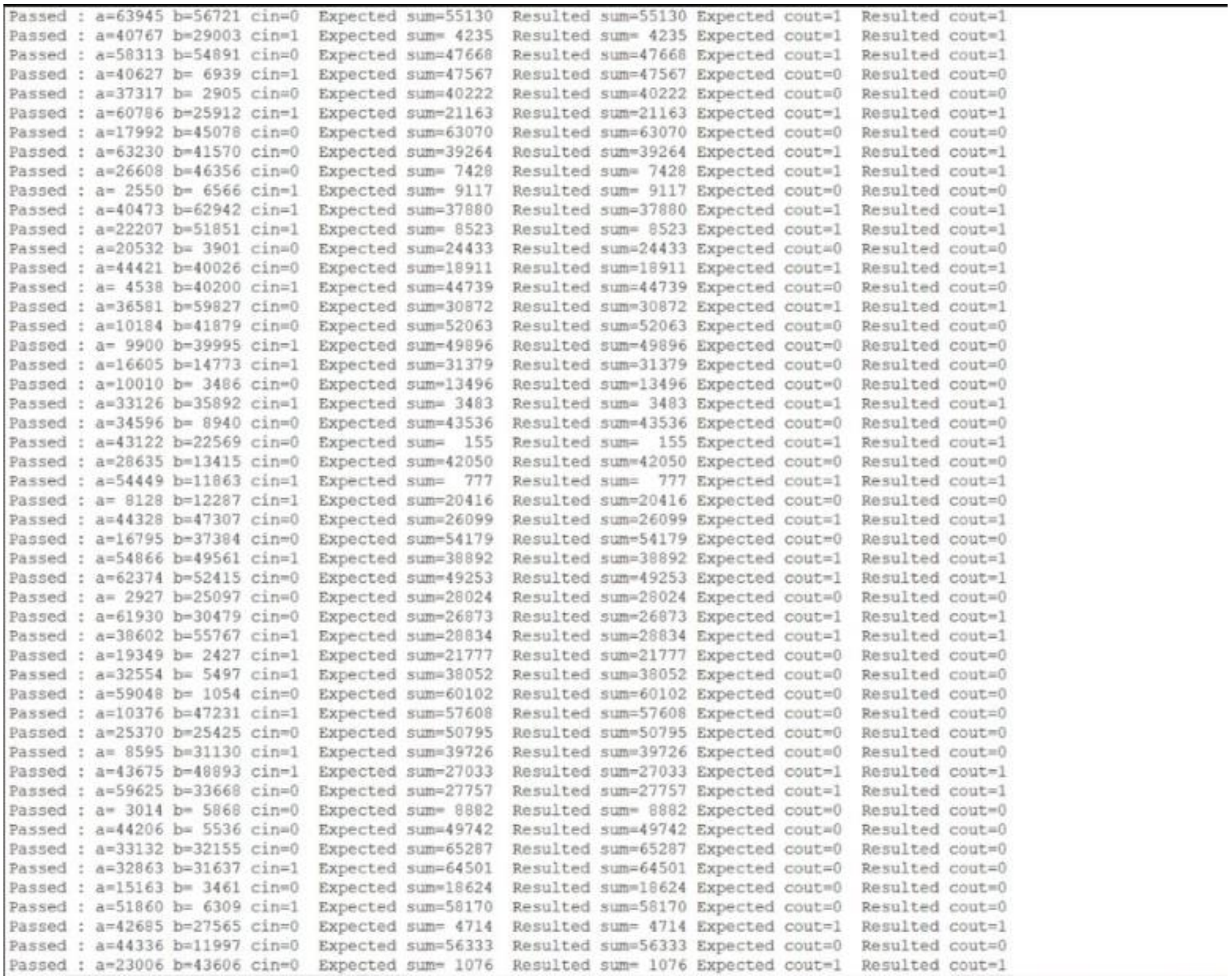
Result



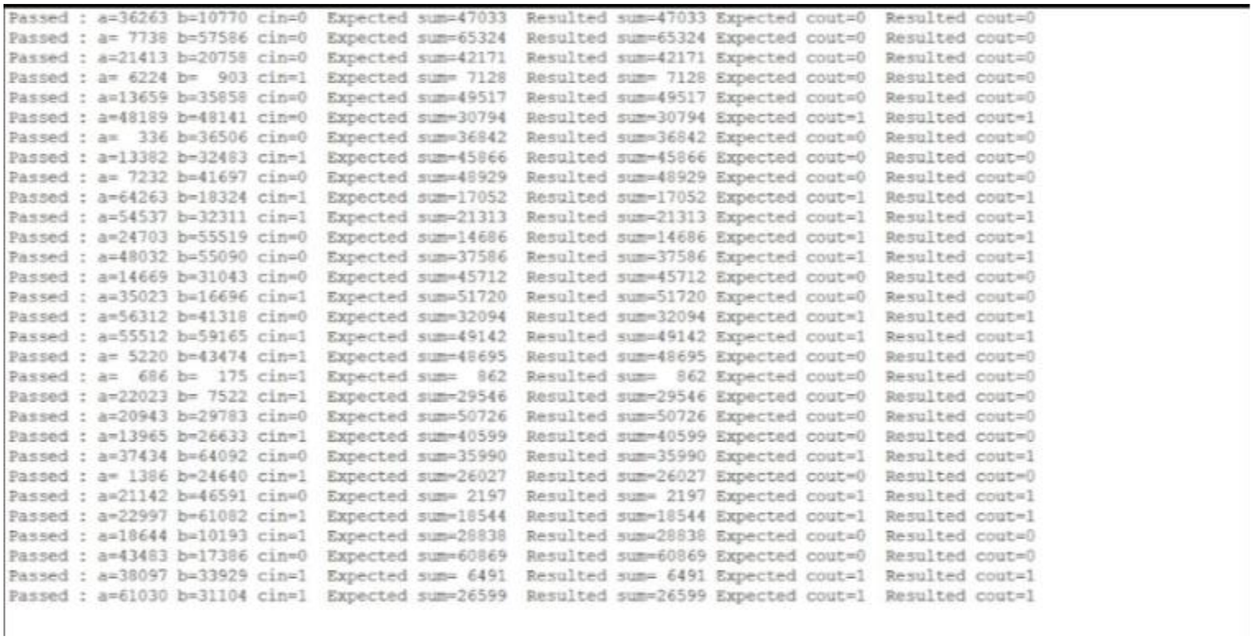
```
Loading snapshot worklib.testbench:sv .................... Done
SVSEED set randomly from command line: 1188184969
ncsim> source /home/eda/cadence/lnx/INCSIVE/icd/icdcm_t1b_016/flow/INCISIV/INCISIV151/15.10.015/lnx86/tools/inca/files/ncsimrc
ncsim> run
--Scoreboard Test Starts- -
Passed : a=12448 b=34045 cin=0  Expected sum=46493  Resulted sum=46493 Expected cout=0  Resulted cout=0
Passed : a=29794 b=13468 cin=0  Expected sum=43262  Resulted sum=43262 Expected cout=0  Resulted cout=0
Passed : a= 7873 b= 7039 cin=0  Expected sum=14912  Resulted sum=14912 Expected cout=0  Resulted cout=0
Passed : a= 4151 b=36008 cin=1  Expected sum=40160  Resulted sum=40160 Expected cout=0  Resulted cout=0
Passed : a=54132 b= 8293 cin=1  Expected sum=62426  Resulted sum=62426 Expected cout=0  Resulted cout=0
Passed : a= 8957 b=44858 cin=0  Expected sum=53815  Resulted sum=53815 Expected cout=0  Resulted cout=0
Passed : a=52346 b=29001 cin=1  Expected sum=15812  Resulted sum=15812 Expected cout=1  Resulted cout=1
Passed : a=15594 b=31701 cin=0  Expected sum=47295  Resulted sum=47295 Expected cout=0  Resulted cout=0
Passed : a=16379 b=38650 cin=1  Expected sum=55030  Resulted sum=55030 Expected cout=0  Resulted cout=0
Passed : a=29658 b=15326 cin=0  Expected sum=44984  Resulted sum=44984 Expected cout=0  Resulted cout=0
Passed : a=64097 b=16467 cin=1  Expected sum=15029  Resulted sum=15029 Expected cout=1  Resulted cout=1
Passed : a=13934 b=54287 cin=0  Expected sum= 2685  Resulted sum= 2685 Expected cout=1  Resulted cout=1
Passed : a=65329 b=20638 cin=0  Expected sum=20431  Resulted sum=20431 Expected cout=1  Resulted cout=1
Passed : a=16965 b= 5152 cin=1  Expected sum=22118  Resulted sum=22118 Expected cout=0  Resulted cout=0
Passed : a=13693 b=64996 cin=1  Expected sum=13154  Resulted sum=13154 Expected cout=1  Resulted cout=1
Passed : a=53734 b=49438 cin=1  Expected sum=37637  Resulted sum=37637 Expected cout=1  Resulted cout=1
Passed : a=23435 b= 5887 cin=1  Expected sum=29323  Resulted sum=29323 Expected cout=0  Resulted cout=0
Passed : a= 7829 b=53111 cin=1  Expected sum=60941  Resulted sum=60941 Expected cout=0  Resulted cout=0
Passed : a= 9883 b=18810 cin=0  Expected sum=28693  Resulted sum=28693 Expected cout=0  Resulted cout=0
Passed : a=46623 b=13114 cin=1  Expected sum=59738  Resulted sum=59738 Expected cout=0  Resulted cout=0
Passed : a=47358 b=19655 cin=0  Expected sum= 1477  Resulted sum= 1477 Expected cout=1  Resulted cout=1
Passed : a=49141 b=16263 cin=0  Expected sum=65404  Resulted sum=65404 Expected cout=0  Resulted cout=0
Passed : a=32043 b=62585 cin=0  Expected sum=29092  Resulted sum=29092 Expected cout=1  Resulted cout=1
Passed : a=37667 b= 3761 cin=1  Expected sum=41429  Resulted sum=41429 Expected cout=0  Resulted cout=0
Passed : a=47939 b= 6211 cin=0  Expected sum=54150  Resulted sum=54150 Expected cout=0  Resulted cout=0
Passed : a=55654 b=28584 cin=1  Expected sum=18703  Resulted sum=18703 Expected cout=1  Resulted cout=1
Passed : a=46779 b=58943 cin=1  Expected sum=40187  Resulted sum=40187 Expected cout=1  Resulted cout=1
Passed : a=62089 b=53181 cin=0  Expected sum=49734  Resulted sum=49734 Expected cout=1  Resulted cout=1
Passed : a=37007 b=59636 cin=0  Expected sum=31107  Resulted sum=31107 Expected cout=1  Resulted cout=1
Passed : a=18132 b=23311 cin=0  Expected sum=41443  Resulted sum=41443 Expected cout=0  Resulted cout=0
Passed : a=65214 b= 3428 cin=1  Expected sum= 3107  Resulted sum= 3107 Expected cout=1  Resulted cout=1
Passed : a=39471 b=27649 cin=0  Expected sum= 1584  Resulted sum= 1584 Expected cout=1  Resulted cout=1
Passed : a=36968 b=63811 cin=0  Expected sum=35243  Resulted sum=35243 Expected cout=1  Resulted cout=1
Passed : a=17079 b= 8320 cin=1  Expected sum=25400  Resulted sum=25400 Expected cout=0  Resulted cout=0
Passed : a=36198 b=49831 cin=1  Expected sum=20494  Resulted sum=20494 Expected cout=1  Resulted cout=1
Passed : a=18116 b=  842 cin=0  Expected sum=18958  Resulted sum=18958 Expected cout=0  Resulted cout=0
Passed : a= 8230 b=52997 cin=1  Expected sum=61228  Resulted sum=61228 Expected cout=0  Resulted cout=0
Passed : a=50483 b=49896 cin=1  Expected sum=34844  Resulted sum=34844 Expected cout=1  Resulted cout=1
Passed : a=57743 b=53002 cin=0  Expected sum=45209  Resulted sum=45209 Expected cout=1  Resulted cout=1
Passed : a=38875 b=25768 cin=0  Expected sum=64643  Resulted sum=64643 Expected cout=0  Resulted cout=0
Passed : a= 8258 b=34441 cin=1  Expected sum=42700  Resulted sum=42700 Expected cout=0  Resulted cout=0
Passed : a=23588 b=40481 cin=1  Expected sum=64070  Resulted sum=64070 Expected cout=0  Resulted cout=0
Passed : a=64859 b=56153 cin=1  Expected sum=55477  Resulted sum=55477 Expected cout=1  Resulted cout=1
Passed : a=16488 b=19545 cin=0  Expected sum=36033  Resulted sum=36033 Expected cout=0  Resulted cout=0
Passed : a=33474 b=12236 cin=1  Expected sum=45711  Resulted sum=45711 Expected cout=0  Resulted cout=0
```



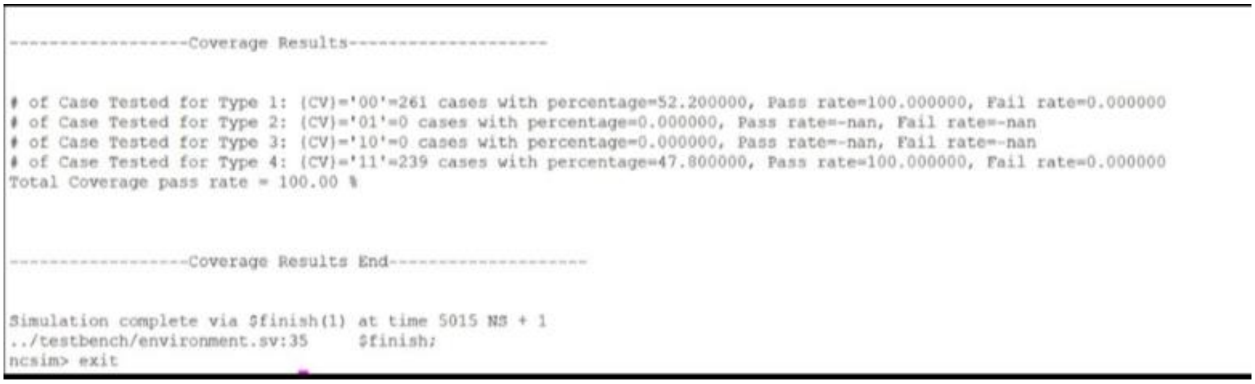
```
Passed : a=64875 b=19262 cin=0  Expected sum=18601  Resulted sum=18601 Expected cout=1  Resulted cout=1
Passed : a=42263 b=20732 cin=0  Expected sum=62995  Resulted sum=62995 Expected cout=0  Resulted cout=0
Passed : a=15315 b= 3606 cin=1  Expected sum=18922  Resulted sum=18922 Expected cout=0  Resulted cout=0
Passed : a=26534 b=32419 cin=0  Expected sum=58953  Resulted sum=58953 Expected cout=0  Resulted cout=0
Passed : a=37690 b= 6826 cin=0  Expected sum=44516  Resulted sum=44516 Expected cout=0  Resulted cout=0
Passed : a=52987 b=44479 cin=1  Expected sum=31931  Resulted sum=31931 Expected cout=1  Resulted cout=1
Passed : a=65108 b=19808 cin=1  Expected sum=19381  Resulted sum=19381 Expected cout=1  Resulted cout=1
Passed : a=36950 b= 2714 cin=1  Expected sum=39665  Resulted sum=39665 Expected cout=0  Resulted cout=0
Passed : a=47376 b= 9620 cin=0  Expected sum=56996  Resulted sum=56996 Expected cout=0  Resulted cout=0
Passed : a=36139 b=26581 cin=0  Expected sum=62720  Resulted sum=62720 Expected cout=0  Resulted cout=0
Passed : a=44012 b=36271 cin=0  Expected sum=14747  Resulted sum=14747 Expected cout=1  Resulted cout=1
Passed : a=44727 b=15260 cin=1  Expected sum=59988  Resulted sum=59988 Expected cout=0  Resulted cout=0
Passed : a=38090 b=63878 cin=0  Expected sum=36432  Resulted sum=36432 Expected cout=1  Resulted cout=1
Passed : a=18906 b= 5274 cin=0  Expected sum=24180  Resulted sum=24180 Expected cout=0  Resulted cout=0
Passed : a=40168 b=51621 cin=0  Expected sum=26253  Resulted sum=26253 Expected cout=1  Resulted cout=1
Passed : a=61088 b=41463 cin=1  Expected sum=37016  Resulted sum=37016 Expected cout=1  Resulted cout=1
Passed : a=18262 b=34577 cin=1  Expected sum=52840  Resulted sum=52840 Expected cout=0  Resulted cout=0
Passed : a=39491 b=20554 cin=1  Expected sum=60046  Resulted sum=60046 Expected cout=0  Resulted cout=0
Passed : a=16251 b=44719 cin=1  Expected sum=60971  Resulted sum=60971 Expected cout=0  Resulted cout=0
Passed : a=57900 b=16472 cin=1  Expected sum= 8837  Resulted sum= 8837 Expected cout=1  Resulted cout=1
Passed : a=60332 b=58771 cin=0  Expected sum=53567  Resulted sum=53567 Expected cout=1  Resulted cout=1
Passed : a=19764 b=59687 cin=0  Expected sum=13915  Resulted sum=13915 Expected cout=1  Resulted cout=1
Passed : a=38011 b=26302 cin=1  Expected sum=64314  Resulted sum=64314 Expected cout=0  Resulted cout=0
Passed : a=64931 b=57199 cin=0  Expected sum=56594  Resulted sum=56594 Expected cout=1  Resulted cout=1
Passed : a=50362 b=41140 cin=0  Expected sum=25966  Resulted sum=25966 Expected cout=1  Resulted cout=1
Passed : a=43121 b=53870 cin=0  Expected sum=31455  Resulted sum=31455 Expected cout=1  Resulted cout=1
Passed : a=19901 b=55318 cin=1  Expected sum= 9684  Resulted sum= 9684 Expected cout=1  Resulted cout=1
Passed : a=23683 b=45743 cin=0  Expected sum= 3890  Resulted sum= 3890 Expected cout=1  Resulted cout=1
Passed : a= 5915 b=25037 cin=1  Expected sum=30953  Resulted sum=30953 Expected cout=0  Resulted cout=0
Passed : a=16419 b=63911 cin=0  Expected sum=14794  Resulted sum=14794 Expected cout=1  Resulted cout=1
Passed : a=50755 b=11811 cin=0  Expected sum=62566  Resulted sum=62566 Expected cout=0  Resulted cout=0
Passed : a=50662 b=29480 cin=0  Expected sum=14606  Resulted sum=14606 Expected cout=1  Resulted cout=1
Passed : a=62009 b=43050 cin=0  Expected sum=39523  Resulted sum=39523 Expected cout=1  Resulted cout=1
Passed : a=58972 b=10471 cin=0  Expected sum= 3907  Resulted sum= 3907 Expected cout=1  Resulted cout=1
Passed : a=58154 b= 7475 cin=1  Expected sum=   94  Resulted sum=   94 Expected cout=1  Resulted cout=1
Passed : a=29359 b=56208 cin=1  Expected sum=20032  Resulted sum=20032 Expected cout=1  Resulted cout=1
Passed : a=22538 b=36940 cin=0  Expected sum=59478  Resulted sum=59478 Expected cout=0  Resulted cout=0
Passed : a=14249 b=32239 cin=1  Expected sum=46489  Resulted sum=46489 Expected cout=0  Resulted cout=0
Passed : a=  643 b= 4273 cin=0  Expected sum= 4916  Resulted sum= 4916 Expected cout=0  Resulted cout=0
Passed : a=61186 b=36470 cin=1  Expected sum=32121  Resulted sum=32121 Expected cout=1  Resulted cout=1
Passed : a=30653 b= 4487 cin=0  Expected sum=35140  Resulted sum=35140 Expected cout=0  Resulted cout=0
Passed : a= 2118 b=60559 cin=0  Expected sum=62677  Resulted sum=62677 Expected cout=0  Resulted cout=0
Passed : a=59506 b=32725 cin=0  Expected sum=26695  Resulted sum=26695 Expected cout=1  Resulted cout=1
Passed : a=37843 b=57030 cin=1  Expected sum=29338  Resulted sum=29338 Expected cout=1  Resulted cout=1
Passed : a=26760 b=27730 cin=0  Expected sum=54490  Resulted sum=54490 Expected cout=0  Resulted cout=0
Passed : a=  719 b=60468 cin=0  Expected sum=61187  Resulted sum=61187 Expected cout=0  Resulted cout=0
Passed : a= 3312 b=61811 cin=1  Expected sum=65124  Resulted sum=65124 Expected cout=0  Resulted cout=0
Passed : a= 9191 b=59445 cin=0  Expected sum= 3100  Resulted sum= 3100 Expected cout=1  Resulted cout=1
Passed : a=21646 b=25963 cin=0  Expected sum=47609  Resulted sum=47609 Expected cout=0  Resulted cout=0
Passed : a=13903 b=41195 cin=1  Expected sum=55099  Resulted sum=55099 Expected cout=0  Resulted cout=0
```



```
Passed : a=63945 b=56721 cin=0  Expected sum=55130  Resulted sum=55130 Expected cout=1  Resulted cout=1
Passed : a=40767 b=29003 cin=1  Expected sum= 4235  Resulted sum= 4235 Expected cout=1  Resulted cout=1
Passed : a=58313 b=54891 cin=0  Expected sum=47668  Resulted sum=47668 Expected cout=1  Resulted cout=1
Passed : a=40627 b= 6939 cin=1  Expected sum=47567  Resulted sum=47567 Expected cout=0  Resulted cout=0
Passed : a=37317 b= 2905 cin=0  Expected sum=40222  Resulted sum=40222 Expected cout=0  Resulted cout=0
Passed : a=60786 b=25912 cin=1  Expected sum=21163  Resulted sum=21163 Expected cout=1  Resulted cout=1
Passed : a=17992 b=45078 cin=0  Expected sum=63070  Resulted sum=63070 Expected cout=0  Resulted cout=0
Passed : a=63230 b=41570 cin=0  Expected sum=39264  Resulted sum=39264 Expected cout=1  Resulted cout=1
Passed : a=26608 b=46356 cin=0  Expected sum= 7428  Resulted sum= 7428 Expected cout=1  Resulted cout=1
Passed : a= 2550 b= 6566 cin=1  Expected sum= 9117  Resulted sum= 9117 Expected cout=0  Resulted cout=0
Passed : a=40473 b=62942 cin=1  Expected sum=37880  Resulted sum=37880 Expected cout=1  Resulted cout=1
Passed : a=22207 b=51851 cin=1  Expected sum= 8523  Resulted sum= 8523 Expected cout=1  Resulted cout=1
Passed : a=20532 b= 3901 cin=0  Expected sum=24433  Resulted sum=24433 Expected cout=0  Resulted cout=0
Passed : a=44421 b=40026 cin=0  Expected sum=18911  Resulted sum=18911 Expected cout=1  Resulted cout=1
Passed : a= 4538 b=40200 cin=1  Expected sum=44739  Resulted sum=44739 Expected cout=0  Resulted cout=0
Passed : a=36581 b=59827 cin=0  Expected sum=30872  Resulted sum=30872 Expected cout=1  Resulted cout=1
Passed : a=10184 b=41879 cin=0  Expected sum=52063  Resulted sum=52063 Expected cout=0  Resulted cout=0
Passed : a= 9900 b=39995 cin=1  Expected sum=49896  Resulted sum=49896 Expected cout=0  Resulted cout=0
Passed : a=16605 b=14773 cin=1  Expected sum=31379  Resulted sum=31379 Expected cout=0  Resulted cout=0
Passed : a=10010 b= 3486 cin=0  Expected sum=13496  Resulted sum=13496 Expected cout=0  Resulted cout=0
Passed : a=33126 b=35892 cin=1  Expected sum= 3483  Resulted sum= 3483 Expected cout=1  Resulted cout=1
Passed : a=34596 b= 8940 cin=0  Expected sum=43536  Resulted sum=43536 Expected cout=0  Resulted cout=0
Passed : a=43122 b=22569 cin=0  Expected sum=  155  Resulted sum=  155 Expected cout=1  Resulted cout=1
Passed : a=28635 b=13415 cin=0  Expected sum=42050  Resulted sum=42050 Expected cout=0  Resulted cout=0
Passed : a=54449 b=11863 cin=1  Expected sum=  777  Resulted sum=  777 Expected cout=1  Resulted cout=1
Passed : a= 8128 b=12287 cin=1  Expected sum=20416  Resulted sum=20416 Expected cout=0  Resulted cout=0
Passed : a=44328 b=47307 cin=0  Expected sum=26099  Resulted sum=26099 Expected cout=1  Resulted cout=1
Passed : a=16795 b=37384 cin=0  Expected sum=54179  Resulted sum=54179 Expected cout=0  Resulted cout=0
Passed : a=54866 b=49561 cin=1  Expected sum=38892  Resulted sum=38892 Expected cout=1  Resulted cout=1
Passed : a=62374 b=52415 cin=0  Expected sum=49253  Resulted sum=49253 Expected cout=1  Resulted cout=1
Passed : a= 2927 b=25097 cin=0  Expected sum=28024  Resulted sum=28024 Expected cout=0  Resulted cout=0
Passed : a=61930 b=30479 cin=0  Expected sum=26873  Resulted sum=26873 Expected cout=1  Resulted cout=1
Passed : a=38602 b=55767 cin=1  Expected sum=28834  Resulted sum=28834 Expected cout=1  Resulted cout=1
Passed : a=19349 b= 2427 cin=1  Expected sum=21777  Resulted sum=21777 Expected cout=0  Resulted cout=0
Passed : a=32554 b= 5497 cin=1  Expected sum=38052  Resulted sum=38052 Expected cout=0  Resulted cout=0
Passed : a=59048 b= 1054 cin=0  Expected sum=60102  Resulted sum=60102 Expected cout=0  Resulted cout=0
Passed : a=10376 b=47231 cin=1  Expected sum=57608  Resulted sum=57608 Expected cout=0  Resulted cout=0
Passed : a=25370 b=25425 cin=0  Expected sum=50795  Resulted sum=50795 Expected cout=0  Resulted cout=0
Passed : a= 8595 b=31130 cin=1  Expected sum=39726  Resulted sum=39726 Expected cout=0  Resulted cout=0
Passed : a=43675 b=48893 cin=1  Expected sum=27033  Resulted sum=27033 Expected cout=1  Resulted cout=1
Passed : a=59625 b=33668 cin=0  Expected sum=27757  Resulted sum=27757 Expected cout=1  Resulted cout=1
Passed : a= 3014 b= 5868 cin=0  Expected sum= 8882  Resulted sum= 8882 Expected cout=0  Resulted cout=0
Passed : a=44206 b= 5536 cin=0  Expected sum=49742  Resulted sum=49742 Expected cout=0  Resulted cout=0
Passed : a=33132 b=32155 cin=0  Expected sum=65287  Resulted sum=65287 Expected cout=0  Resulted cout=0
Passed : a=32863 b=31637 cin=1  Expected sum=64501  Resulted sum=64501 Expected cout=0  Resulted cout=0
Passed : a=15163 b= 3461 cin=0  Expected sum=18624  Resulted sum=18624 Expected cout=0  Resulted cout=0
Passed : a=51860 b= 6309 cin=1  Expected sum=58170  Resulted sum=58170 Expected cout=0  Resulted cout=0
Passed : a=42685 b=27565 cin=0  Expected sum= 4714  Resulted sum= 4714 Expected cout=1  Resulted cout=1
Passed : a=44336 b=11997 cin=0  Expected sum=56333  Resulted sum=56333 Expected cout=0  Resulted cout=0
Passed : a=23006 b=43606 cin=0  Expected sum= 1076  Resulted sum= 1076 Expected cout=1  Resulted cout=1
```



```
Passed : a=36263 b=10770 cin=0  Expected sum=47033  Resulted sum=47033 Expected cout=0  Resulted cout=0
Passed : a= 7738 b=57586 cin=0  Expected sum=65324  Resulted sum=65324 Expected cout=0  Resulted cout=0
Passed : a=21413 b=20758 cin=0  Expected sum=42171  Resulted sum=42171 Expected cout=0  Resulted cout=0
Passed : a= 6224 b=  903 cin=1  Expected sum= 7128  Resulted sum= 7128 Expected cout=0  Resulted cout=0
Passed : a=13659 b=35858 cin=0  Expected sum=49517  Resulted sum=49517 Expected cout=0  Resulted cout=0
Passed : a=48189 b=48141 cin=0  Expected sum=30794  Resulted sum=30794 Expected cout=1  Resulted cout=1
Passed : a=  336 b=36506 cin=0  Expected sum=36842  Resulted sum=36842 Expected cout=0  Resulted cout=0
Passed : a=13382 b=32483 cin=1  Expected sum=45866  Resulted sum=45866 Expected cout=0  Resulted cout=0
Passed : a= 7232 b=41697 cin=0  Expected sum=48929  Resulted sum=48929 Expected cout=0  Resulted cout=0
Passed : a=64263 b=18324 cin=1  Expected sum=17052  Resulted sum=17052 Expected cout=1  Resulted cout=1
Passed : a=54537 b=32311 cin=1  Expected sum=21313  Resulted sum=21313 Expected cout=1  Resulted cout=1
Passed : a=24703 b=55519 cin=0  Expected sum=14686  Resulted sum=14686 Expected cout=1  Resulted cout=1
Passed : a=48032 b=55090 cin=0  Expected sum=37586  Resulted sum=37586 Expected cout=1  Resulted cout=1
Passed : a=14669 b=31043 cin=0  Expected sum=45712  Resulted sum=45712 Expected cout=0  Resulted cout=0
Passed : a=35023 b=16696 cin=1  Expected sum=51720  Resulted sum=51720 Expected cout=0  Resulted cout=0
Passed : a=56312 b=41318 cin=0  Expected sum=32094  Resulted sum=32094 Expected cout=1  Resulted cout=1
Passed : a=55512 b=59165 cin=1  Expected sum=49142  Resulted sum=49142 Expected cout=1  Resulted cout=1
Passed : a= 5220 b=43474 cin=1  Expected sum=48695  Resulted sum=48695 Expected cout=0  Resulted cout=0
Passed : a=  686 b=  175 cin=1  Expected sum=  862  Resulted sum=  862 Expected cout=0  Resulted cout=0
Passed : a=22023 b= 7522 cin=1  Expected sum=29546  Resulted sum=29546 Expected cout=0  Resulted cout=0
Passed : a=20943 b=29783 cin=0  Expected sum=50726  Resulted sum=50726 Expected cout=0  Resulted cout=0
Passed : a=13965 b=26633 cin=1  Expected sum=40599  Resulted sum=40599 Expected cout=0  Resulted cout=0
Passed : a=37434 b=64092 cin=0  Expected sum=35990  Resulted sum=35990 Expected cout=1  Resulted cout=1
Passed : a= 1386 b=24640 cin=1  Expected sum=26027  Resulted sum=26027 Expected cout=0  Resulted cout=0
Passed : a=21142 b=46591 cin=0  Expected sum= 2197  Resulted sum= 2197 Expected cout=1  Resulted cout=1
Passed : a=22997 b=61082 cin=1  Expected sum=18544  Resulted sum=18544 Expected cout=1  Resulted cout=1
Passed : a=18644 b=10193 cin=1  Expected sum=28838  Resulted sum=28838 Expected cout=0  Resulted cout=0
Passed : a=43483 b=17386 cin=0  Expected sum=60869  Resulted sum=60869 Expected cout=0  Resulted cout=0
Passed : a=38097 b=33929 cin=1  Expected sum= 6491  Resulted sum= 6491 Expected cout=1  Resulted cout=1
Passed : a=61030 b=31104 cin=1  Expected sum=26599  Resulted sum=26599 Expected cout=1  Resulted cout=1
```



```
------------------Coverage Results--------------------


# of Case Tested for Type 1: (CV)='00'=261 cases with percentage=52.200000, Pass rate=100.000000, Fail rate=0.000000
# of Case Tested for Type 2: (CV)='01'=0 cases with percentage=0.000000, Pass rate=-nan, Fail rate=-nan
# of Case Tested for Type 3: (CV)='10'=0 cases with percentage=0.000000, Pass rate=-nan, Fail rate=-nan
# of Case Tested for Type 4: (CV)='11'=239 cases with percentage=47.800000, Pass rate=100.000000, Fail rate=0.000000
Total Coverage pass rate = 100.00 %



------------------Coverage Results End--------------------


Simulation complete via $finish(1) at time 5015 NS + 1
../testbench/environment.sv:35     $finish;
ncsim> exit
```

The Report explicitly shows that a coverage of 100% is achieved.

7. Step-4 (Synthesis with Optimized Power, Performance & Area)

.tcl code for varying input delay

```
# Run Genus in Legacy UI if Genus is invoked with Common UI

::legacy::set_attribute common_ui false /
```

```
# CPU Info check (optional)

if {[file exists /proc/cpuinfo]} {

   sh grep "model name" /proc/cpuinfo

   sh grep "cpu MHz" /proc/cpuinfo

}


# Print hostname (optional)

puts "Hostname : [info hostname]"


##########################################################################################

### Preset global variables and attributes

##########################################################################################

# Set input_delay times to iterate over

set input_delays {0.2 0.3 0.4 0.5 0.6}


set DESIGN cla16

set SYN_EFF low

set MAP_EFF low

set OPT_EFF low


# Directory of PDK

set pdk_dir /home/cad/VLSI2Lab/Digital/library/

set_attribute init_lib_search_path $pdk_dir


# Set synthesizing effort for each synthesis stage

set_attribute syn_generic_effort $SYN_EFF

set_attribute syn_map_effort $MAP_EFF

set_attribute syn_opt_effort $OPT_EFF

set_attribute library "slow_vdd1v0_basicCells.lib"
```

```
# Avoid using specific library cells
set_dont_use [get_lib_cells CLK*]
set_dont_use [get_lib_cells SDFF*]
set_dont_use [get_lib_cells DLY*]
set_dont_use [get_lib_cells HOLD*]


# Load Design
read_hdl "${DESIGN}.v"
elaborate $DESIGN
check_design -unresolved


# Iterate over each input delay
foreach input_delay $input_delays {
    # Create a new SDC file for each input_delay time
    set sdc_file "${DESIGN}_input_delay_${input_delay}_low.sdc"


    # Open the SDC file for writing
    set output [open $sdc_file "w"]
    puts $output "set setup_time 0.3"
    puts $output "create_clock -period 10 -waveform {0 6} -name func_clk [get_ports clk]"


    # Writing SDC constraints to the file
    #puts $output "set input_delay ${input_delay}"
    puts $output "set_input_delay ${input_delay} -clock [get_clocks func_clk] {A B Cin}"
    puts $output "set_output_delay 0.6 -clock [get_clocks func_clk] {Cout Sum}"


    puts $output "set_hold_time 0.2 -clock [get_clocks func_clk] {A B Cin}"


    # Closing the SDC file after writing
    close $output
    # Read the newly created SDC file
    read_sdc $sdc_file


    # Run synthesis with the newly set constraints
    syn_generic
    puts "Runtime & Memory after 'syn_generic' for input delay time ${input_delay} ns"
    time_info GENERIC


    # Generate report after synthesis
    report_power -verbose -detail >> reports/power_report_input_delay_${input_delay}_low.txt
    report_area >> reports/area_report_input_delay_${input_delay}_low.txt


    # Synthesize to mapped design
    synthesize -to_mapped
    write -mapped > ${DESIGN}_synth_input_delay_${input_delay}_low.v
}


# Final script for execution
write_script > script
```

Data for Low Effort

Setup Time(.3)

Value	Area	Leakage Power	Internal Power	Net Power	Dynamic Power	Total Power
.1	612	5.625	5046.076	4345.144	9391.22	9396.845
.2	478	4.899	4832.767	3737.644	8570.411	8575.309
.3	487	5.072	5099.365	3714.862	8814.228	8819.30
.4	288	4.245	5849.826	2572.003	8421.829	8426.073
.5	275	4.201	5945.181	2466.956	8412.137	8416.339

Hold Time(ns)

Value	Area	Leakage Power	Internal Power	Net Power	Dynamic Power	Total Power
.1	612	5.625	5046.076	4345.144	9391.22	9396.845
.2	478	4.899	4832.767	3737.644	8570.411	8575.309
.3	487	5.072	5099.365	3714.862	8814.228	8819.30
.4	288	4.245	5849.826	2572.003	8421.829	8426.073
.5	275	4.201	5945.181	2466.956	8412.137	8416.339

Clock Period(ns)

Value	Area	Leakage Power	Internal Power	Net Power	Dynamic Power	Total Power
5	612	5.625	5046.076	4345.144	9391.22	9396.845
10	478	4.899	4832.767	3737.644	8570.411	8575.309
15	487	5.072	5099.365	3714.862	8814.228	8819.30
20	288	4.245	5849.826	2572.003	8421.829	8426.073
25	275	4.201	5945.181	2466.956	8412.137	8416.339

Input Delay(ns)

Value	Area	Leakage	Internal	Net	Dynamic	Total Power
.3	612	5.625	5046.076	4345.144	9391.22	9396.845
.4	478	4.899	7142.693	5636.081	12778.774	12783.673
.5	487	5.072	7583.01	5613.933	13196.943	13202.015
.6	288	4.245	8615.366	3602.222	12217.587	12221.832
.7	275	4.201	8742.557	3471.23	12213.787	12217.988

Output Delay(ns)

Value	Area	Leakage Power	Internal Power	Net Power	Dynamic Power	Total Power
.5	612	5.625	5046.076	4345.144	9391.22	9396.845
.6	478	4.899	4832.767	3737.644	8570.411	8575.309
.7	487	5.072	5099.365	3714.862	8814.228	8819.30
.8	288	4.245	5849.826	2572.003	8421.829	8426.073
.9	275	4.201	5945.181	2466.956	8412.137	8416.339

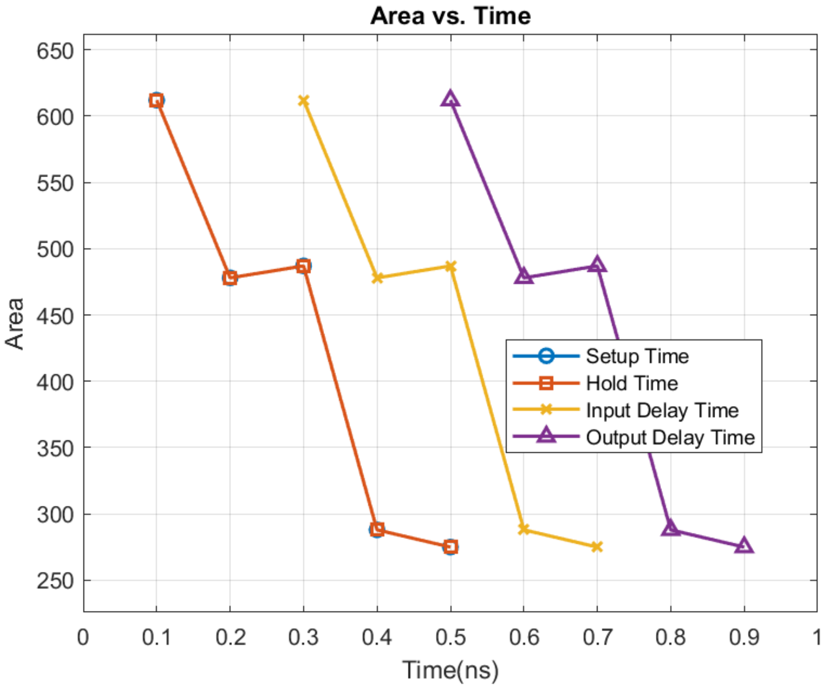
*All Power components are in nW unit

Data for High & Medium Effort

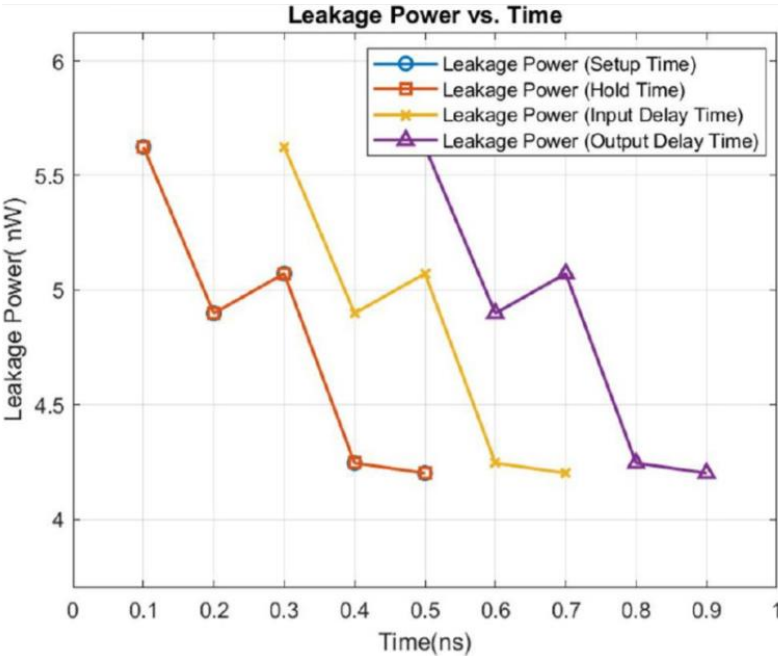
We have achieved the same sort of data for low, Medium and high effort.

Plots

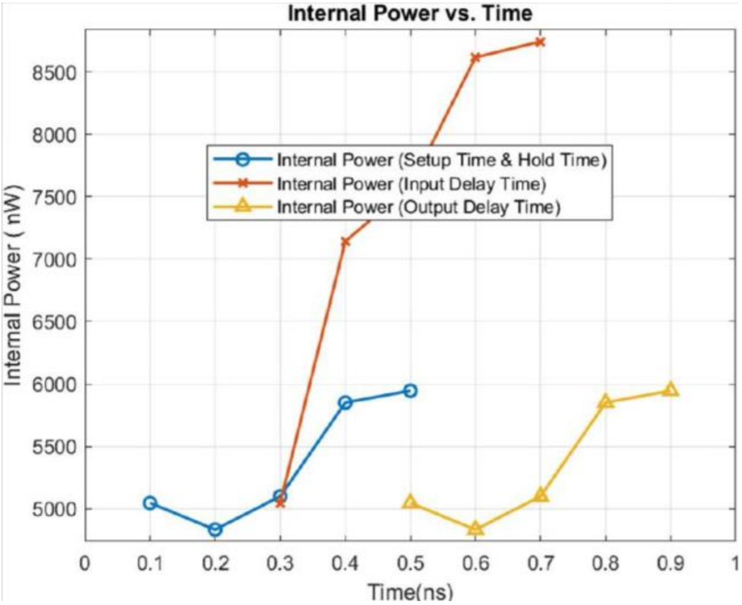
Area vs. Time:



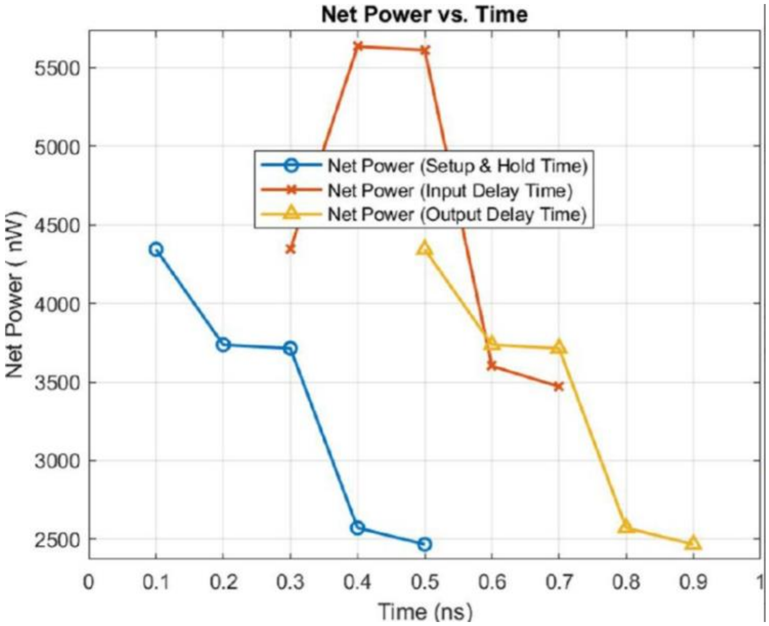
Leakage Power vs. Time:



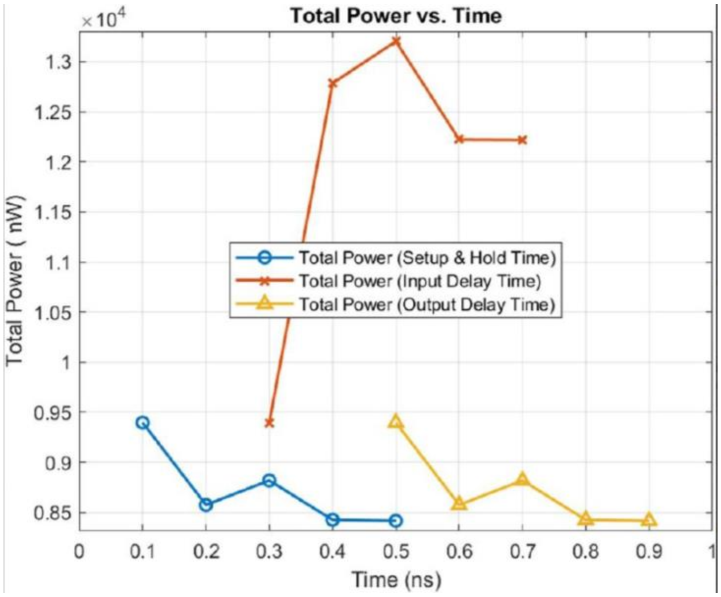
Internal Power vs. Time



Net Power vs. Time

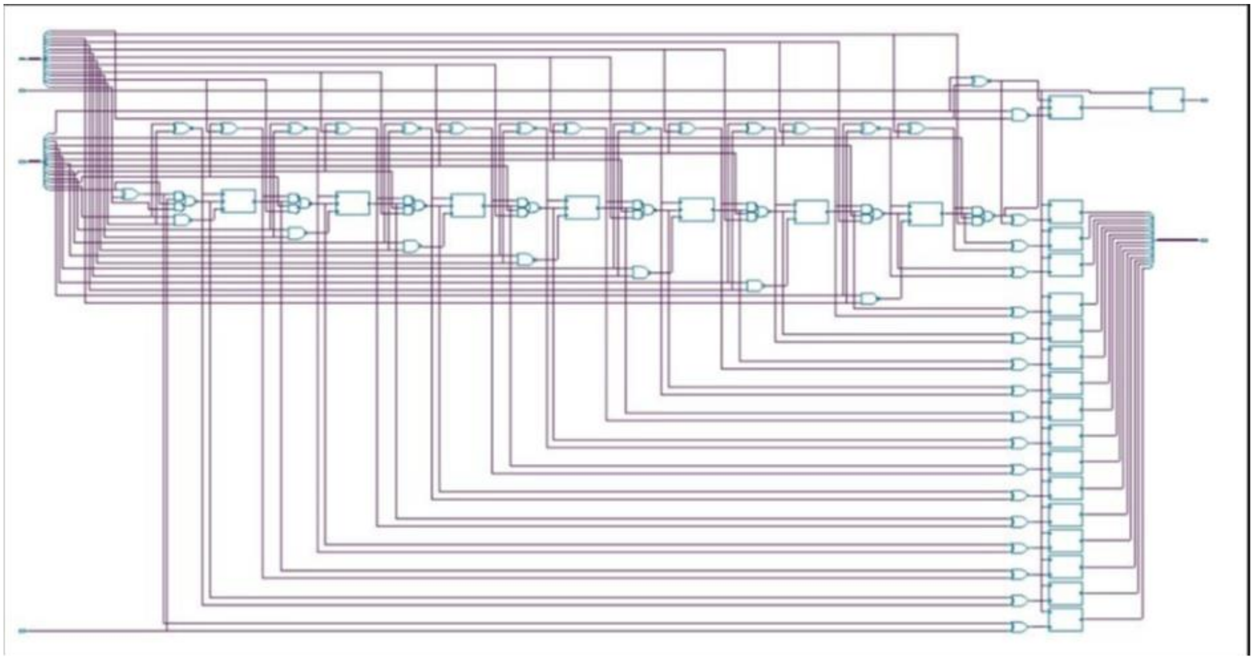


Total Power vs. Time:



Analyzing these data, we have selected the following parameters for better PPA optimization: Hold time = 0.2ns, Setup time = 0.3ns, Clock Frequency = 100MHz, Input delay = 0.4ns, Output delay = 0.6ns.

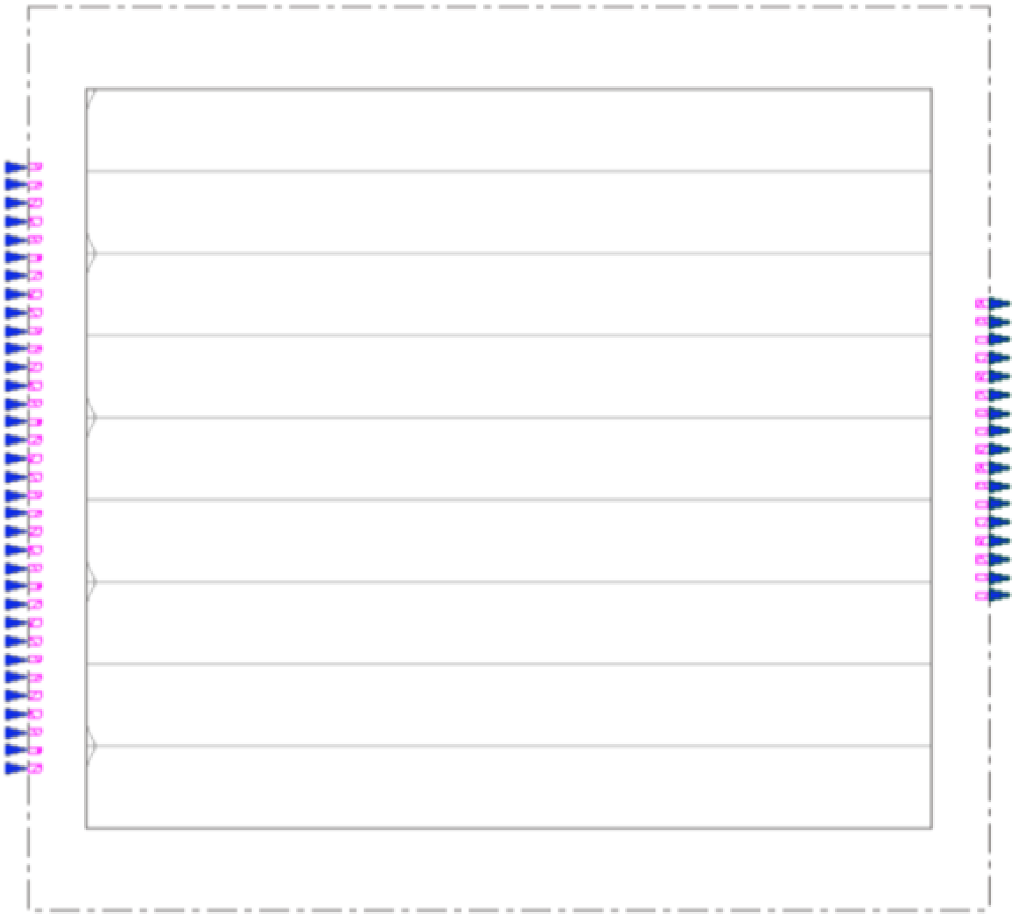
Synthesized Design



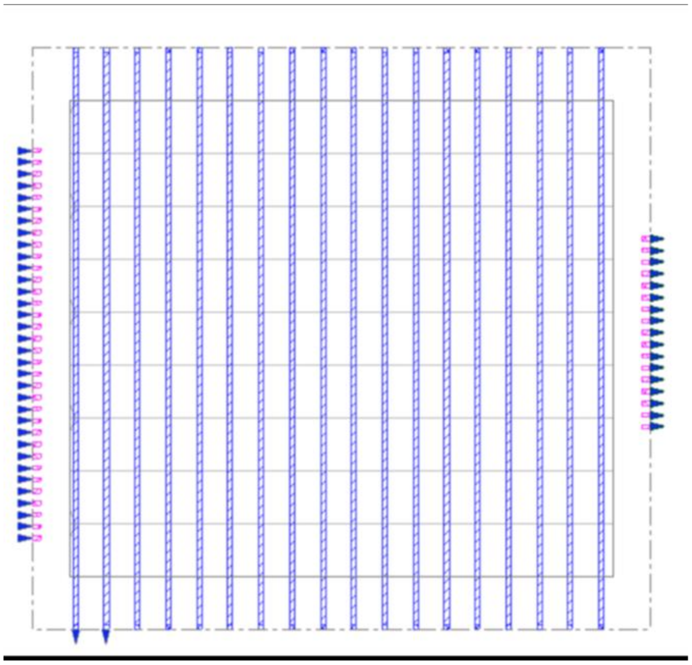
8. Step-5 (Optimized Physical Design with respect to Die Size, Pin Planning, and Power Rails)

Final design

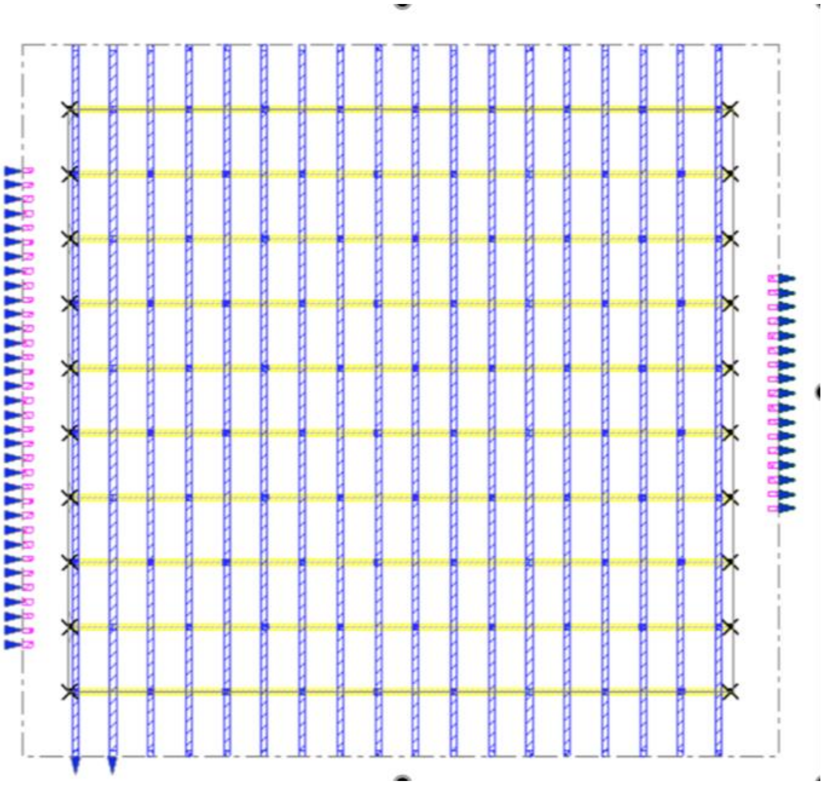
Layout (After Adding Pins)



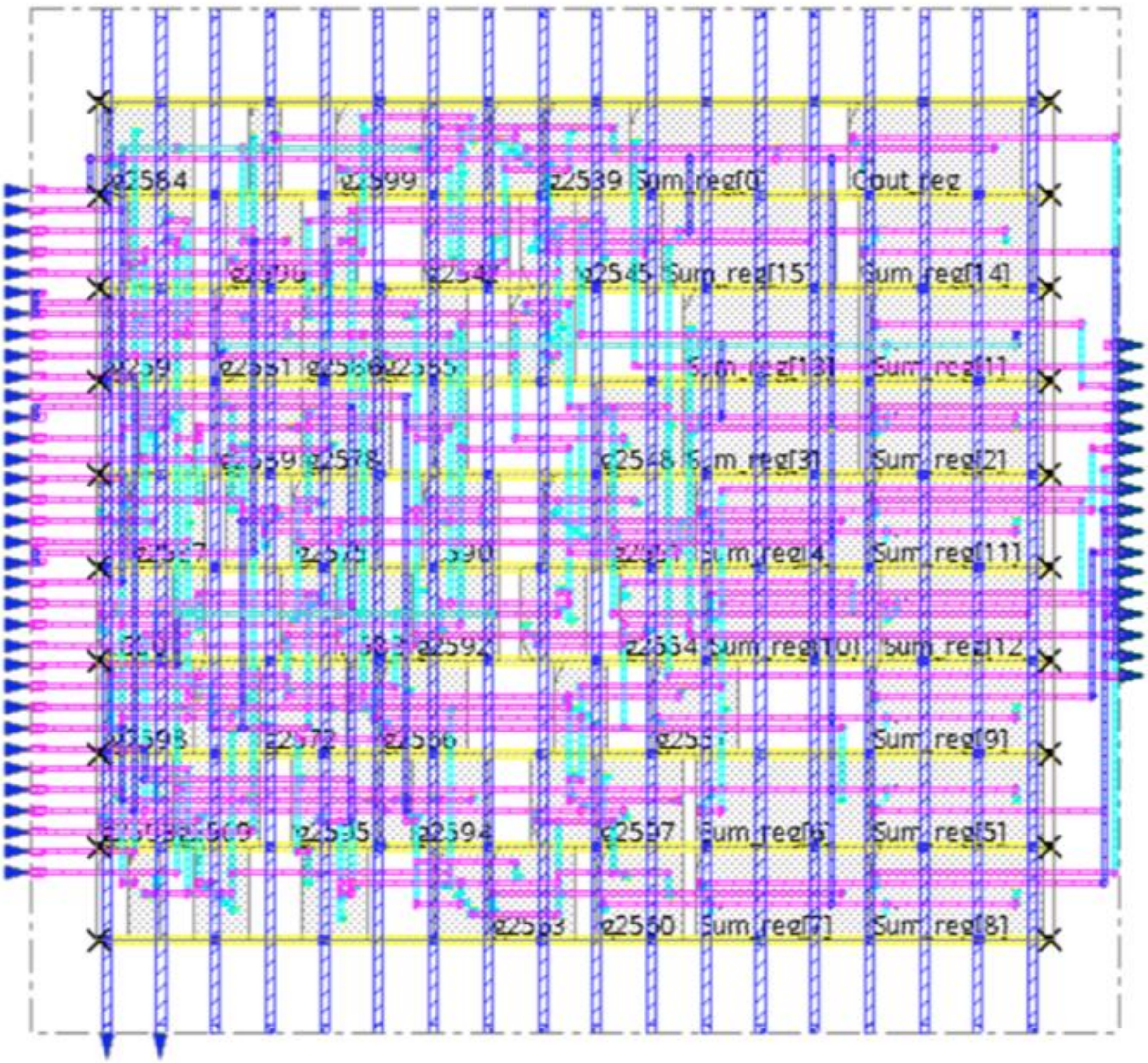
Layout (After Adding Power Strips)



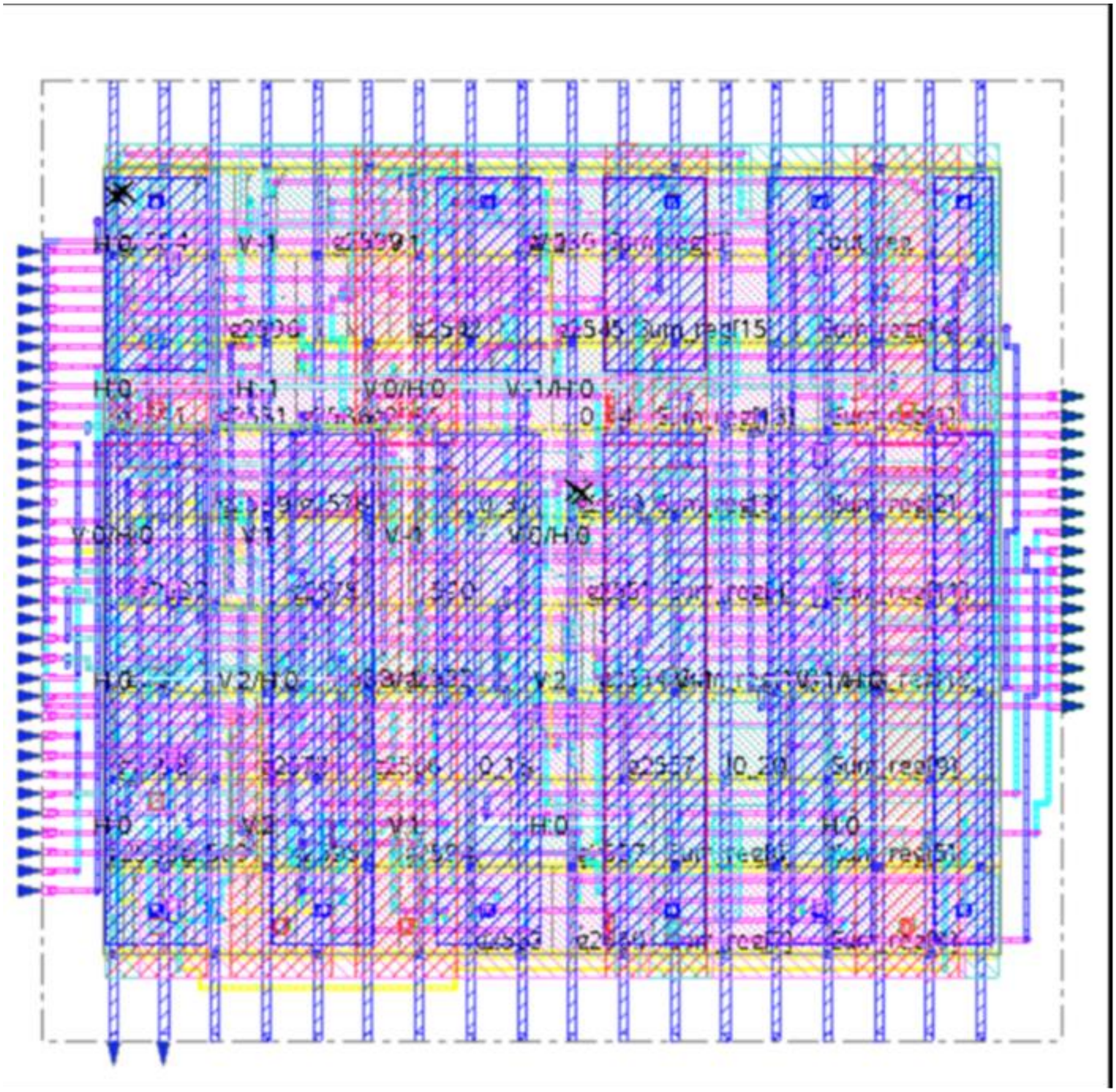
Layout (Adding Rails)



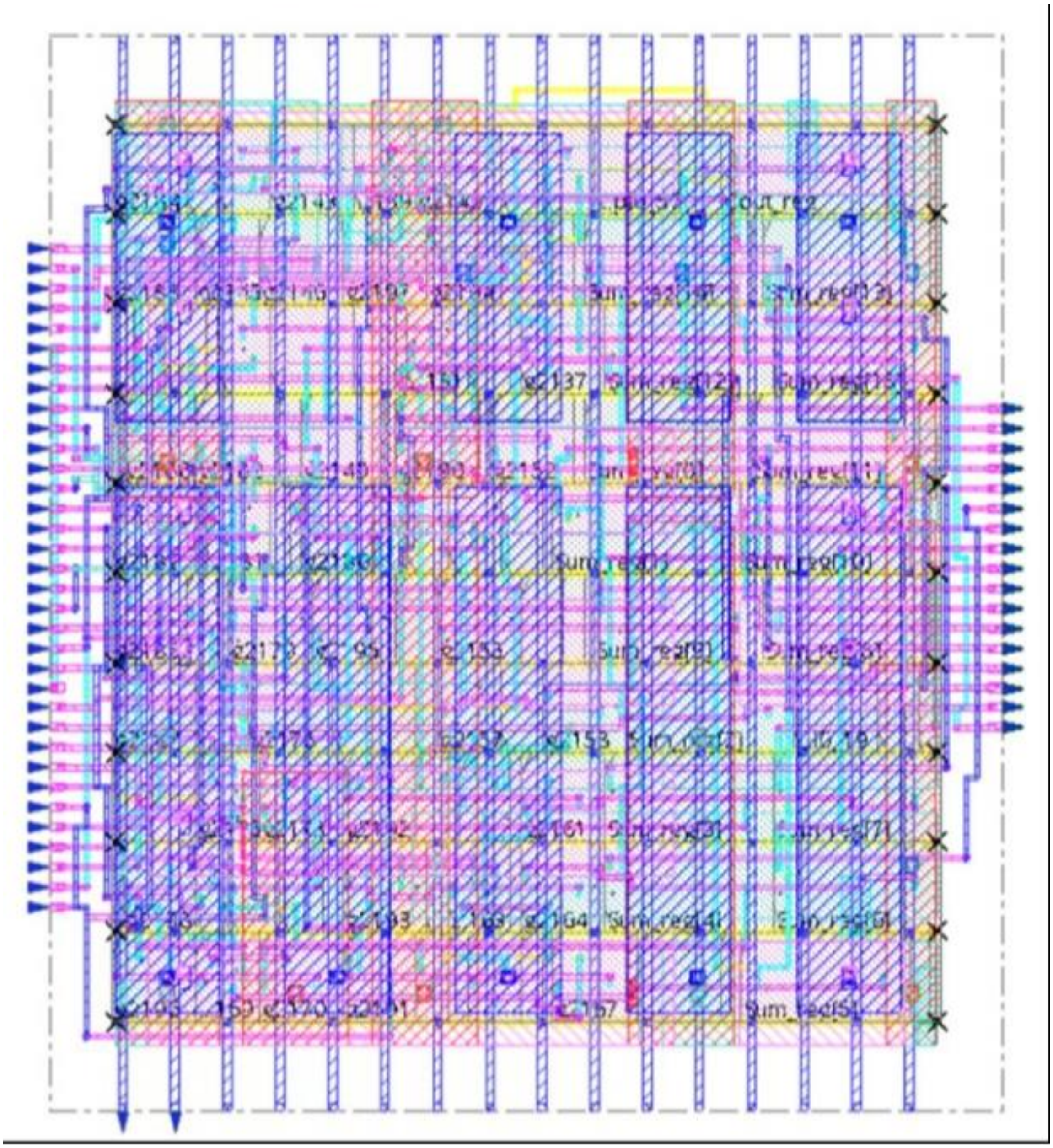
Layout (Cell Placement)



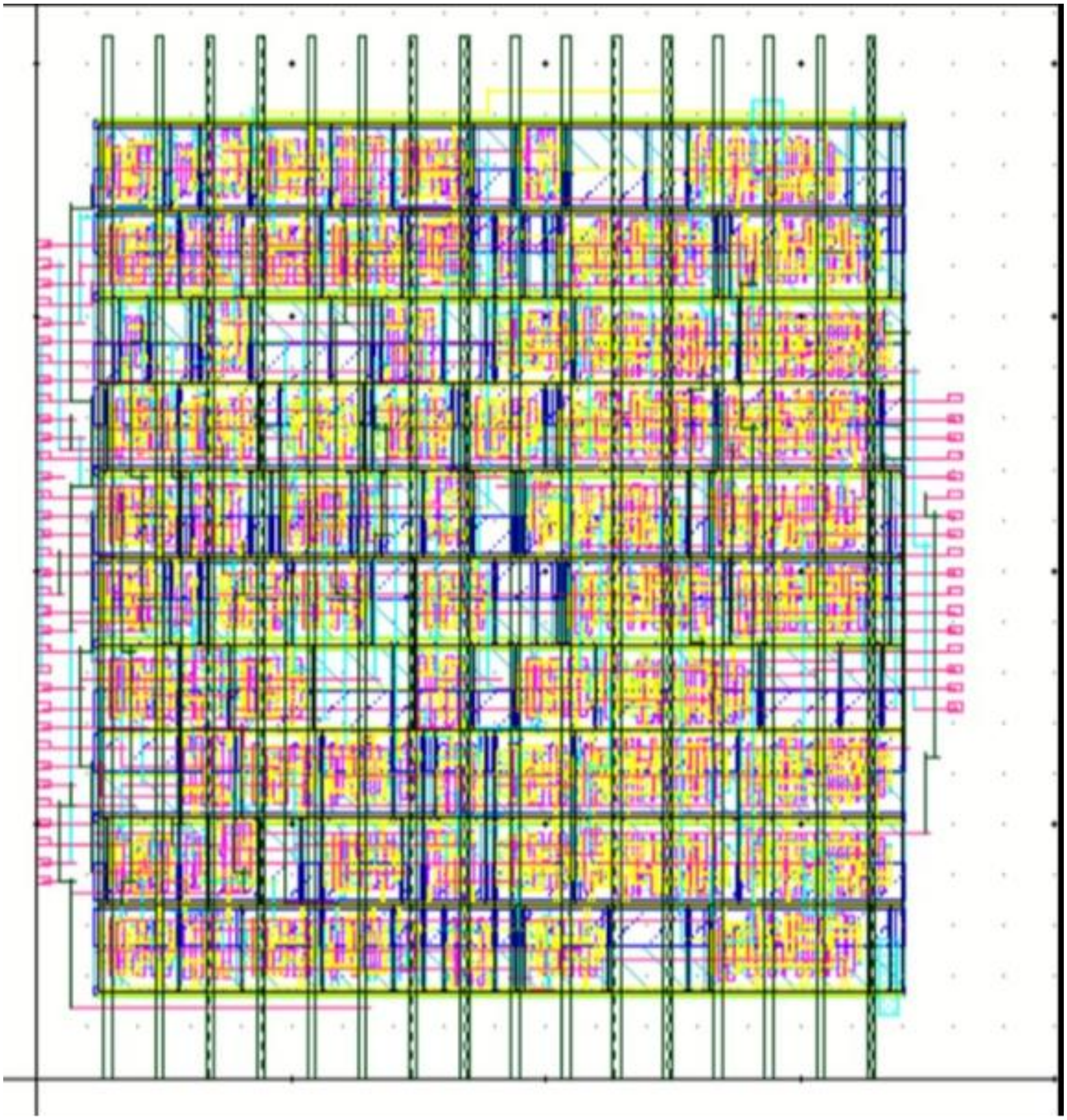
Layout (After Nano Routing)



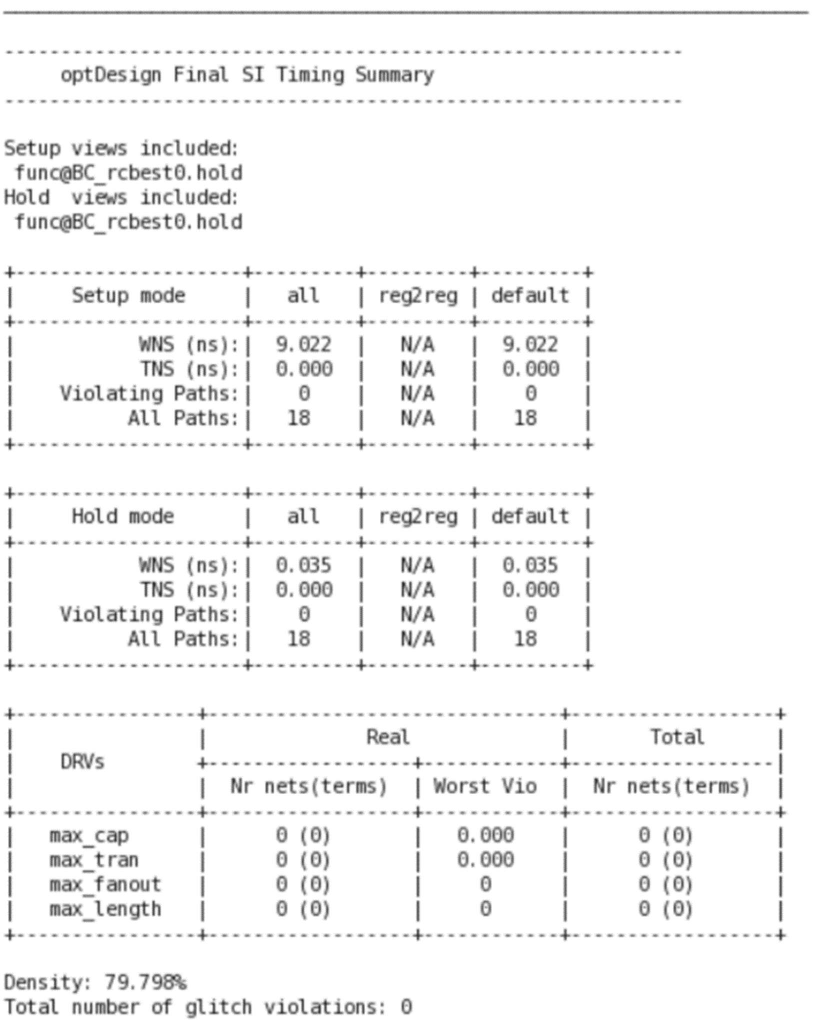
Layout (After Metal Fill)



Layout (Final)



Summary Reports

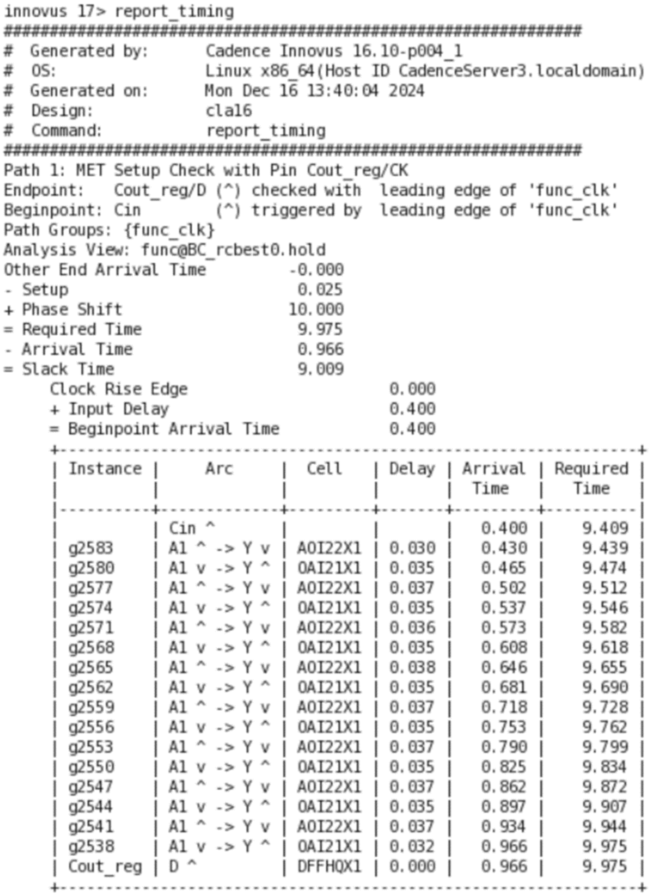


```
---------------------------------------------------------
     optDesign Final SI Timing Summary
---------------------------------------------------------

Setup views included:
 func@BC_rcbest0.hold
Hold  views included:
 func@BC_rcbest0.hold

+--------------------+---------+---------+---------+
|     Setup mode     |   all   | reg2reg | default |
+--------------------+---------+---------+---------+
|           WNS (ns):|  9.022  |   N/A   |  9.022  |
|           TNS (ns):|  0.000  |   N/A   |  0.000  |
|    Violating Paths:|    0    |   N/A   |    0    |
|          All Paths:|   18    |   N/A   |   18    |
+--------------------+---------+---------+---------+

+--------------------+---------+---------+---------+
|     Hold mode      |   all   | reg2reg | default |
+--------------------+---------+---------+---------+
|           WNS (ns):|  0.035  |   N/A   |  0.035  |
|           TNS (ns):|  0.000  |   N/A   |  0.000  |
|    Violating Paths:|    0    |   N/A   |    0    |
|          All Paths:|   18    |   N/A   |   18    |
+--------------------+---------+---------+---------+

+----------------+------------------------------+------------------+
|                |             Real             |       Total      |
|    DRVs        +------------------+-----------+------------------|
|                |  Nr nets(terms)  | Worst Vio |  Nr nets(terms)  |
+----------------+------------------+-----------+------------------+
|   max_cap      |      0 (0)       |   0.000   |      0 (0)       |
|   max_tran     |      0 (0)       |   0.000   |      0 (0)       |
|   max_fanout   |      0 (0)       |     0     |      0 (0)       |
|   max_length   |      0 (0)       |     0     |      0 (0)       |
+----------------+------------------+-----------+------------------+

Density: 79.798%
Total number of glitch violations: 0
```

This report shows that –

- Setup and hold time constraints are met (as there is no negative slack time).
- There are no DRV (Design Rule Violations).
- The density of **79.798%** indicates a moderately high utilization of the available chip area. While further optimization could potentially increase density, keeping it below **80%** ensures practical design feasibility.
- No glitch violations detected, meaning stable and glitch-free logic transitions.

Timing Report

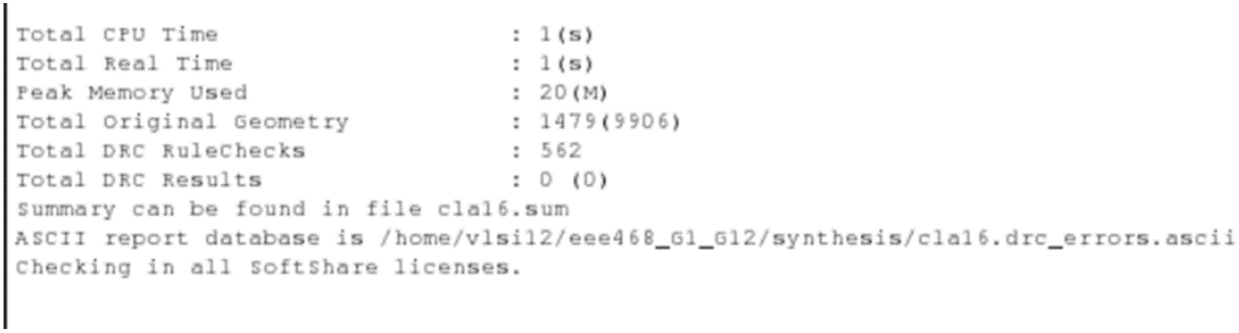


```
innovus 17> report_timing
###############################################################
#  Generated by:      Cadence Innovus 16.10-p004_1
#  OS:                Linux x86_64(Host ID CadenceServer3.localdomain)
#  Generated on:      Mon Dec 16 13:40:04 2024
#  Design:            cla16
#  Command:           report_timing
###############################################################
Path 1: MET Setup Check with Pin Cout_reg/CK
Endpoint:   Cout_reg/D (^) checked with  leading edge of 'func_clk'
Beginpoint: Cin        (^) triggered by  leading edge of 'func_clk'
Path Groups: {func_clk}
Analysis View: func@BC_rcbest0.hold
Other End Arrival Time        -0.000
- Setup                        0.025
+ Phase Shift                 10.000
= Required Time                9.975
- Arrival Time                 0.966
= Slack Time                   9.009
     Clock Rise Edge                      0.000
     + Input Delay                        0.400
     = Beginpoint Arrival Time            0.400
     +---------------------------------------------------------------+
     | Instance |     Arc     |  Cell   | Delay | Arrival | Required |
     |          |             |         |       |  Time   |   Time   |
     |----------+-------------+---------+-------+---------+----------|
     |          | Cin ^       |         |       |   0.400 |    9.409 |
     | g2583    | A1 ^ -> Y v | AOI22X1 | 0.030 |   0.430 |    9.439 |
     | g2580    | A1 v -> Y ^ | OAI21X1 | 0.035 |   0.465 |    9.474 |
     | g2577    | A1 ^ -> Y v | AOI22X1 | 0.037 |   0.502 |    9.512 |
     | g2574    | A1 v -> Y ^ | OAI21X1 | 0.035 |   0.537 |    9.546 |
     | g2571    | A1 ^ -> Y v | AOI22X1 | 0.036 |   0.573 |    9.582 |
     | g2568    | A1 v -> Y ^ | OAI21X1 | 0.035 |   0.608 |    9.618 |
     | g2565    | A1 ^ -> Y v | AOI22X1 | 0.038 |   0.646 |    9.655 |
     | g2562    | A1 v -> Y ^ | OAI21X1 | 0.035 |   0.681 |    9.690 |
     | g2559    | A1 ^ -> Y v | AOI22X1 | 0.037 |   0.718 |    9.728 |
     | g2556    | A1 v -> Y ^ | OAI21X1 | 0.035 |   0.753 |    9.762 |
     | g2553    | A1 ^ -> Y v | AOI22X1 | 0.037 |   0.790 |    9.799 |
     | g2550    | A1 v -> Y ^ | OAI21X1 | 0.035 |   0.825 |    9.834 |
     | g2547    | A1 ^ -> Y v | AOI22X1 | 0.037 |   0.862 |    9.872 |
     | g2544    | A1 v -> Y ^ | OAI21X1 | 0.035 |   0.897 |    9.907 |
     | g2541    | A1 ^ -> Y v | AOI22X1 | 0.037 |   0.934 |    9.944 |
     | g2538    | A1 v -> Y ^ | OAI21X1 | 0.032 |   0.966 |    9.975 |
     | Cout_reg | D ^         | DFFHQX1 | 0.000 |   0.966 |    9.975 |
     +---------------------------------------------------------------+
```

This report shows that –

- Clock propagation is detected successfully.
- Setup timing is matches as WNS (Worst Negative Slack) is positive.

DRC result



```
Total CPU Time                  : 1(s)
Total Real Time                 : 1(s)
Peak Memory Used                : 20(M)
Total Original Geometry         : 1479(9906)
Total DRC RuleChecks            : 562
Total DRC Results               : 0 (0)
Summary can be found in file cla16.sum
ASCII report database is /home/vlsi12/eee468_G1_G12/synthesis/cla16.drc_errors.ascii
Checking in all SoftShare licenses.
```

This result indicates –

- A clean layout as no DRC violations were found.
- The design is efficient in computation and moderate in complexity.

9. Conclusion

In conclusion, the 16-bit Carry Lookahead Adder with Generate-Propagate Logic was successfully designed and implemented to reduce the delay caused by carry propagation, making it faster than traditional adders. The design was optimized for delay, power use, and area, making it suitable for high-performance digital systems. Simulations and tests confirmed that the adder works correctly and efficiently, showing its potential for use in larger circuits like multipliers and ALUs. This project demonstrates how advanced design methods can improve the performance of modern VLSI systems.

10. Links of Codes and Files

- Layered Testbench: https://edaplayground.com/x/s4Q
- Directed Testbench: https://edaplayground.com/x/HAfM
- Theory: *Hierarchical carry lookahead adder*. https://www2.cs.sfu.ca/CourseCentral/150/eyal/lectures/CLA.pdf?fbclid=IwY2xjawHPRzxleHRuA2FlbQIxMAABHevx9r-I4Z8FY82nj37Ccxvs5QwHa25t1dropoFzzizV3uBok5qZ9WZESA_aem_5uCWPh-03y4GOrDOEDN2-A