A REPORT ON

Logic Function Realization With 2 Different Designs

2017B5A30983P	SWAGAT PANDA
2017B5A30935P	DEEPAK JAIN
2017B5A30944P	PIYUSH JAIN
2017B5A30624P	NEELANCHAL JOSHI

Submitted in partial fulfillment of the course

EEE F313/ INSTRF313 – Analog &Digital VLSI Design

Under the Guidance of

Dr. Anu Gupta

Birla Institute of Technology and Science, Pilani

August - December 2020

PROBLEM STATEMENT 6

(a) Logic Function Realization

F= AB +AD +BD+ CD

Realize the above logic function using CMOS logic style. (use less than 21 transistors including both NMOS and PMOS in your design) (use load of 100fF)

(b) Verilog

Two eight-bit numbers need to be added using a serial bit adder. A serial bit adder is a circuit has only one full adder cell and a memory element to hold the carry. The two eight-bit numbers are to be stored in two eight-bit registers. The registers and any other components necessary can be implemented using modeling style of your choice.

- # Reference Inverter:

The aspect ratios of the NMOS and PMOS are to be made such that their drive currents are equal. The drive currents come out to be in the ratio 3.987 (I_n:I_p), which is to be compensated by their aspect ratios. Hence, our reference inverter has W_p:W_n = 3.987 and L_n = L_p = 0.18um.

For the sake of convenience, the value of the W_p:W_n ratio has been taken to be as **4**.

Table of Contents

PART (A) -- Logic Function Realization:

Logical function to be realized**:** F= AB +AD +BD+ CD

We are presenting two designs for the same logic function:
Design 1 - Single Finger Layout
Design 2 – Inter-digitized Multi Finger Layout

1. DESIGN 1 - Single Finger layout

- **Design parameters:**

- General parameters:

PARAMETERS	VALUES
Vdc	1.8V
Technology	180 nm
Load Capacitance	100 fF
L	180 nm

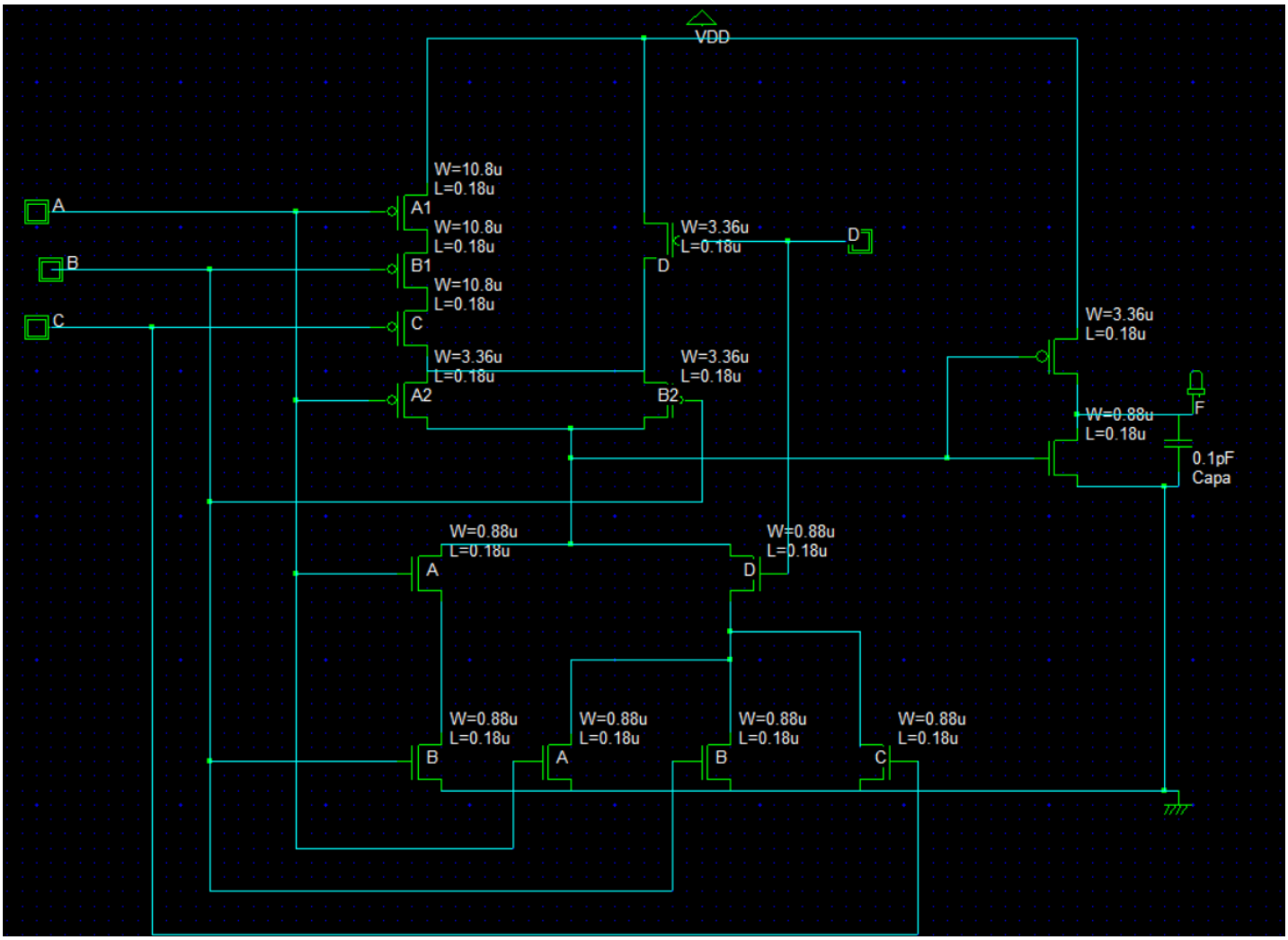
- PMOS parameters:

PARAMETERS	VALUES
Wa1 pmos	10.08 um
Wb1 pmos	10.08 um
Wa2 pmos	3.36 um
Wb2 pmos	3.36 um
Wc pmos	10.08 um
Wd pmos	3.36 um
Winv pmos	3.36 um

- NMOS parameters

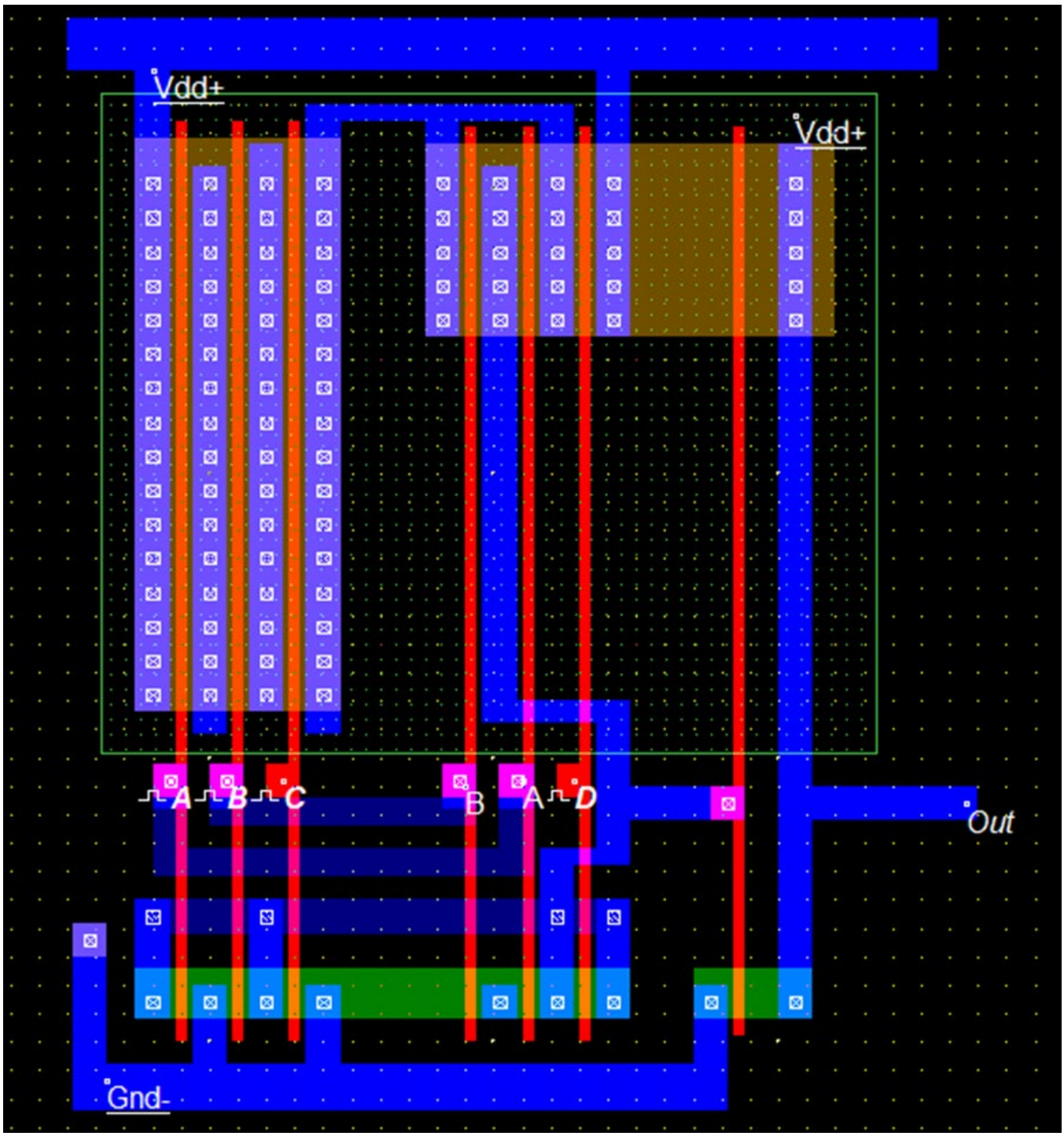
PARAMETERS	VALUES
Wa1 nmos	0.88 um
Wb1 nmos	0.88 um
Wa2 nmos	0.88 um
Wb2 nmos	0.88 um
Wc nmos	0.88 um
Wd nmos	0.88 um
Winv nmos	0.88 um

- **Schematic**



- **Layout**

Note: Virtual Capacitance of 100fF to be attached to the output. Considered for the output plot.



- **OUTPUT**

- **DELAY:**

  T_{PHL} = 245ps

  T_{PLH} = 90ps

  $T_D = (T_{PHL}+T_{PLH})/2$ = 167.5ps

- **AREA**

  The area of the design (excluding Vdc and GND area) is : 307 um^2

- **POWER**

  $\underline{Power = \alpha_{0->1} * Vdd^2 * C_L * f}$

  $\alpha_{0->1}$ = 0.24609

  C_L = 100fF + 4.72 fF (from parasitic extraction at node O)

  f= $1/2T_D$ (FOR MIN. POWER CALCULATION)

  Vdd=1.8V

  Power(min.) = $\underline{2.492x10^{-4}\ Watt}$

- **Parasitic Capacitance:** 4.72 fF

2. DESIGN 2 – Inter-digitized Multi-finger Layout

- **Design parameters:**

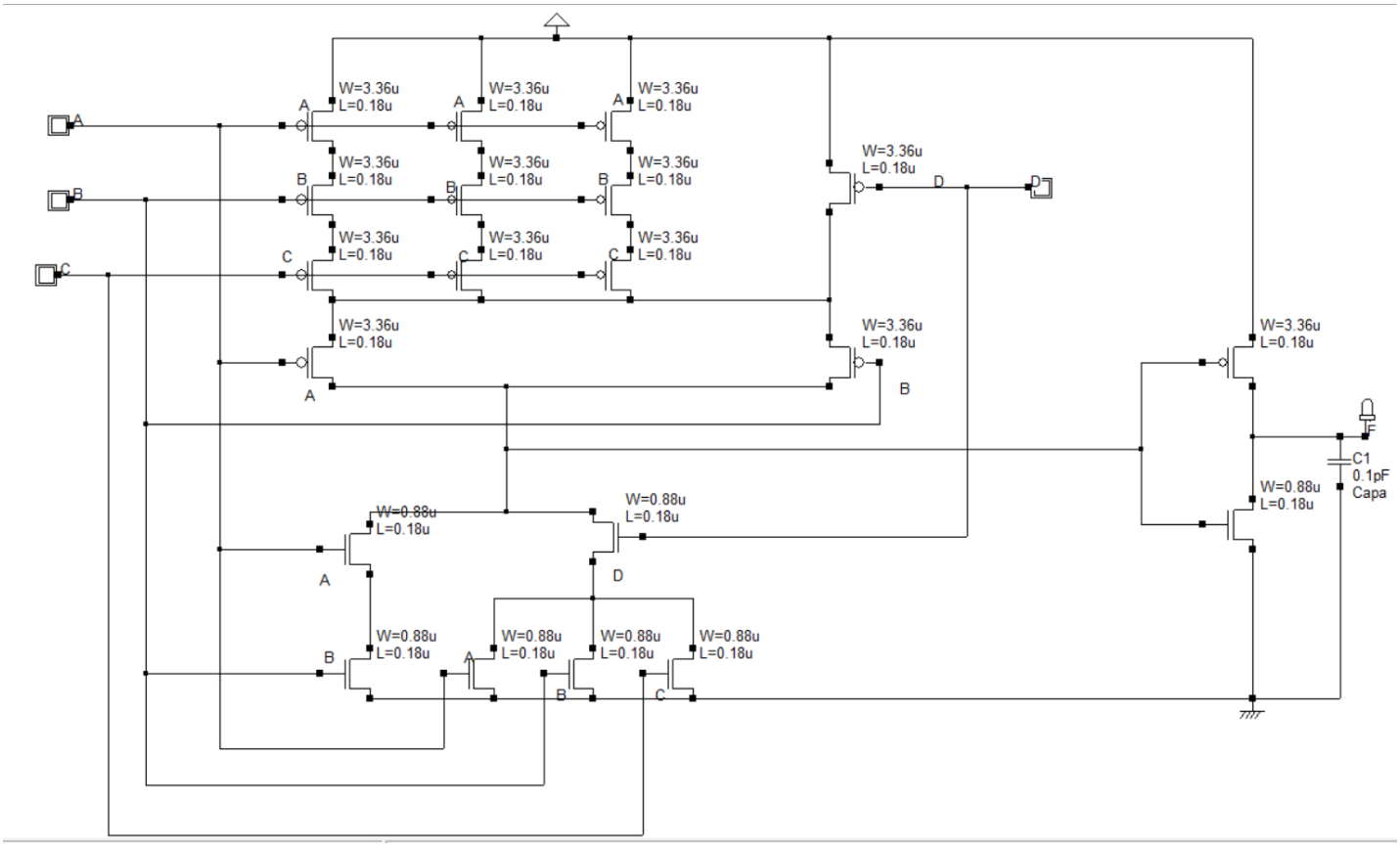
General parameters:

PARAMETERS	VALUES
Vdc	1.8V
Technology	180 nm
Load Capacitance	100 fF
L	180 nm

MOS Parameters:

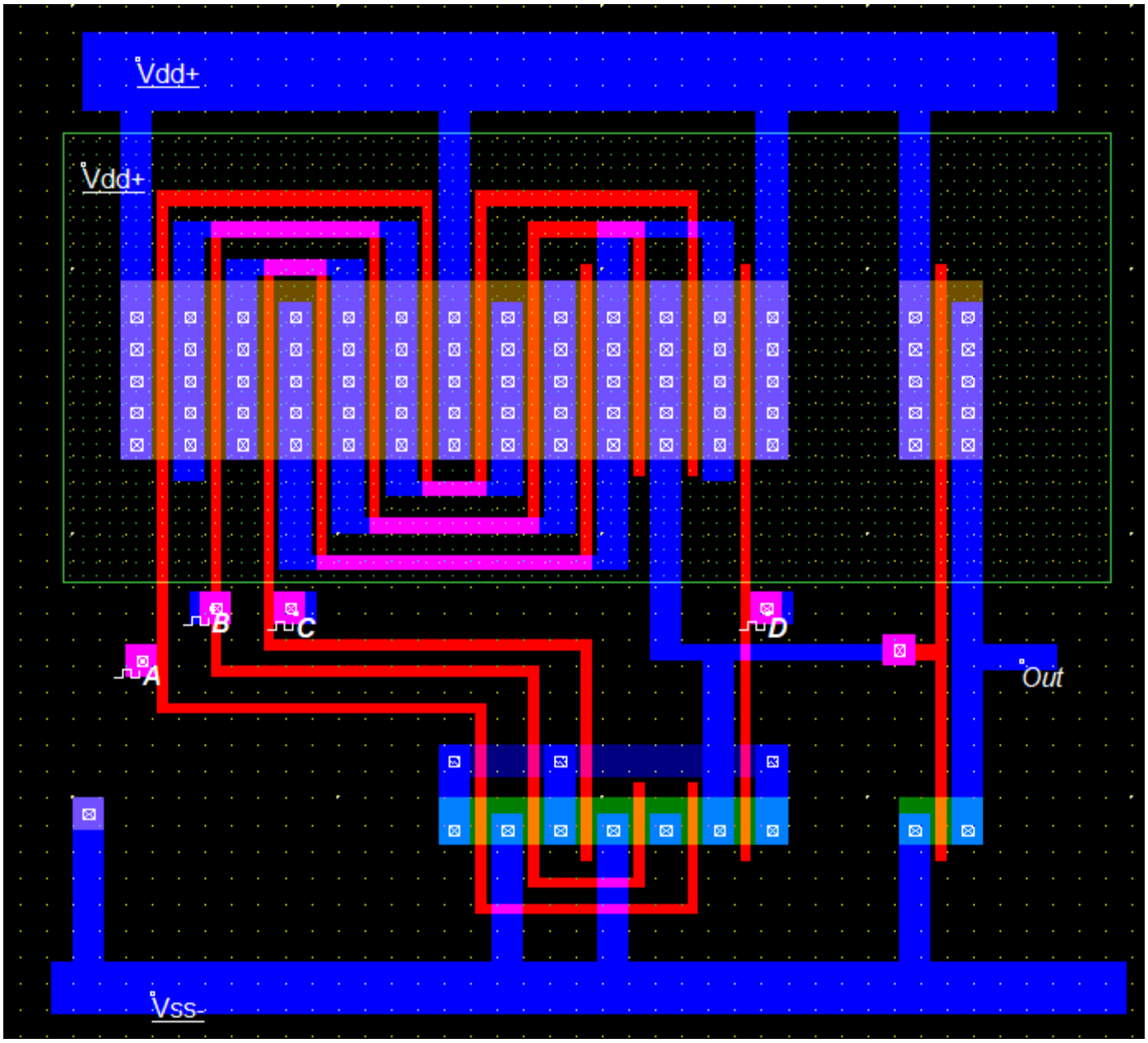
PARAMETERS	VALUES
Wpmos	3.36um
Wnmos	0.880um
L	180

- **Schematic**

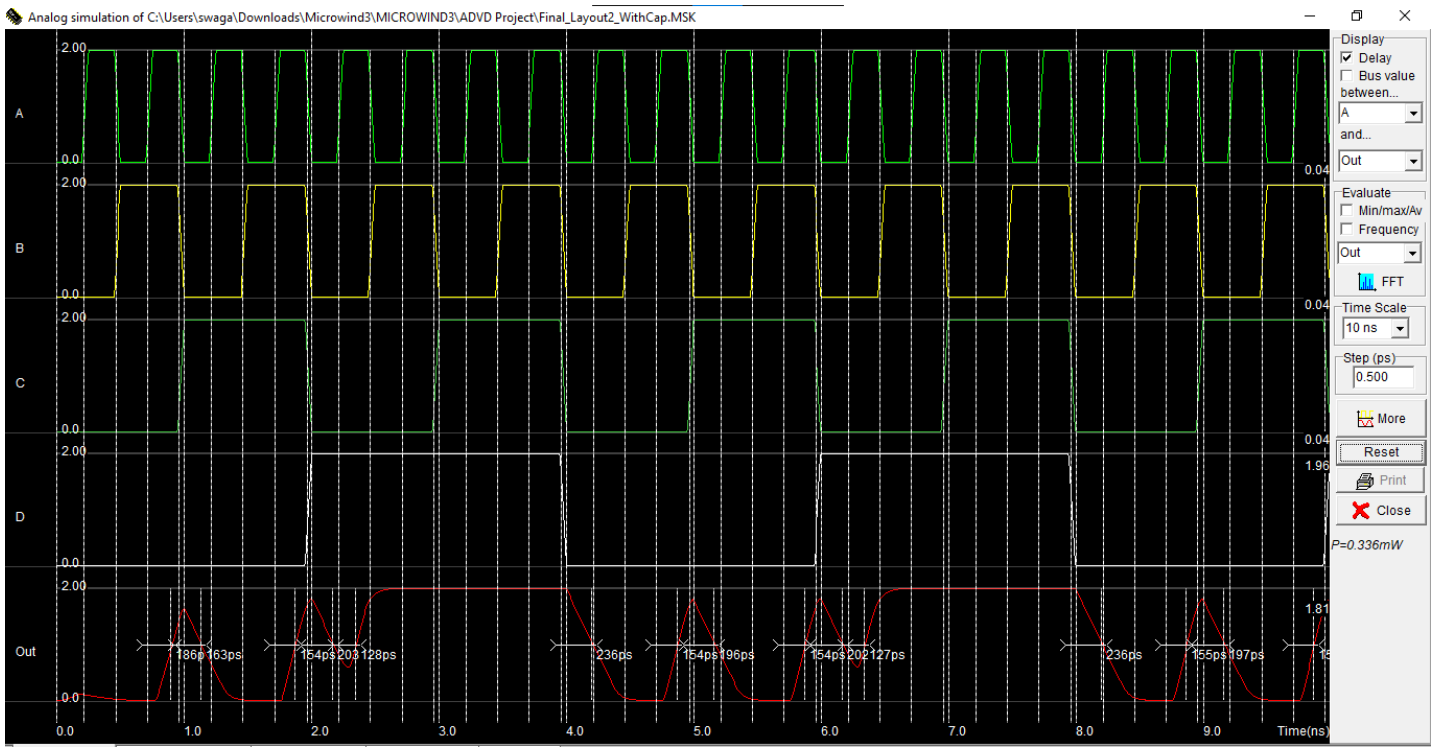


- **Layout**

Note: Virtual Capacitance of 100fF to be attached to the output. Considered for the output plot.



- **OUTPUT**



- **DELAY**

T_{PLH} = 232ps
T_{PHL} = 90ps

$$T_D = (T_{PHL} + T_{PLH})/2 = 161ps$$

- **AREA**

The area of the design (excluding Vdc and GND area) is: 259.23 um^2

- **POWER**

$$\underline{Power = \alpha_{0->1} * Vdd^2 * C_L * f}$$

$\alpha_{0->1}$ = 0.24609
C_L = 100fF + 2.83fF (from parasitic extraction at node O)
f= $1/2T_D$ (FOR MIN. POWER CALCULATION)
Vdd=1.8V

Power(min.) = $\underline{2.546x10^{-4}\ Watt}$

- **Parasitic Capacitance**: 2.83 fF

3. Comparison Table:

Parameters	Design 1 – Single Finger Layout	Design 2 – Interdigitized Multi Finger Layout
$T_{D\ (Delay)}$	167.5ps	161ps
Parasitics	4.72fF	2.83fF
Area	307 um^2	259.23 um^2
Minimum Power	$2.492x10^{-4}$ Watt	$2.546x10^{-4}$ Watt

As we can see, the second design yields lower area and lesser delay, but more power when compared to the first design.

4. Problems Faced:

- Finding out the area of the layout was tough without disturbing it
- Calculated power and power displayed by Microwind do not agree with each other
- Waveform heavily distorted by the load capacitance

5. Innovation and Optimization

The problems statement required the design of two different layout approaches to implement the given function. We went with Single Finger and Multifinger Inter-digitized layouts. One approach gives us an optimized area, while the other gives us optimized power. The layouts should be used according to the application requirements.

6. Part (B) -- Verilog :

Problem statement :

Two eight bit numbers need to be added using a serial bit adder. A serial bit adder is a circuit has only one full adder cell and a memory element to hold the carry. The two eight bit numbers are to be stored in two eight bit registers. The registers and any other components necessary can be implemented using modeling style of your choice.

VERILOG CODE :

```
module serial_adder(clk,DA,DB,Cin1,Cout,ld,Sum);
        input [7:0] DA,DB;
        input clk;
        input Cin1;
        reg Cin;
        output Cout;
        output [7:0]Sum;
        wire S;
        wire Cout;
        reg [7:0] A,B;
        inout ld;
        wire w1,w2,w3;
        assign Sum=A;

        xor ( w1,A[0],B[0]);
        and (w2,w1,Cin);
        and (w3,A[0],B[0]);
        xor      (S,w1,Cin);
        or(Cout,w2,w3);

initial $monitor("%d A=%d,B=%d,ld=%b%b %b %b %b%b",$time,A,B,ld,Cin,A[0],B[0],Cout,S);

        always @( posedge clk, posedge ld)
                begin
                        if(ld==0)
                        begin
                        A[0]<=A[1];
                        A[1]<=A[2];
                        A[2]<=A[3];
                        A[3]<=A[4];
                        A[4]<=A[5];
                        A[5]<=A[6];
                        A[6]<=A[7];
                        A[7]<=S;

                        B[0]<=B[1];
                        B[1]<=B[2];
```

```
                B[2]<=B[3];
                B[3]<=B[4];
                B[4]<=B[5];
                B[5]<=B[6];
                B[6]<=B[7];
                B[7]<=B[0];

                Cin<=Cout;
                end
                if(ld==1)
                begin A<=DA;B<=DB;Cin<=Cin1;end
        end

endmodule
```

TEST BENCH SIMULATION

```
`timescale 10ns/1ns
module testbench;

reg[7:0]A1,B1;
wire Cout;
reg Cin;
reg clk,ld1; wire ld; wire [7:0]Sum;

assign ld=ld1;

serial_adder SA(clk,A1,B1,Cin,Cout,ld,Sum);

initial
  begin
        #11 A1=8'b00001010; B1=8'b00001010;Cin =0; ld1=1;
        #1 ld1=0;
        clk=1'b0;
        forever #5 clk=~clk;
  end
initial #95 $finish;
endmodule
```

Results:

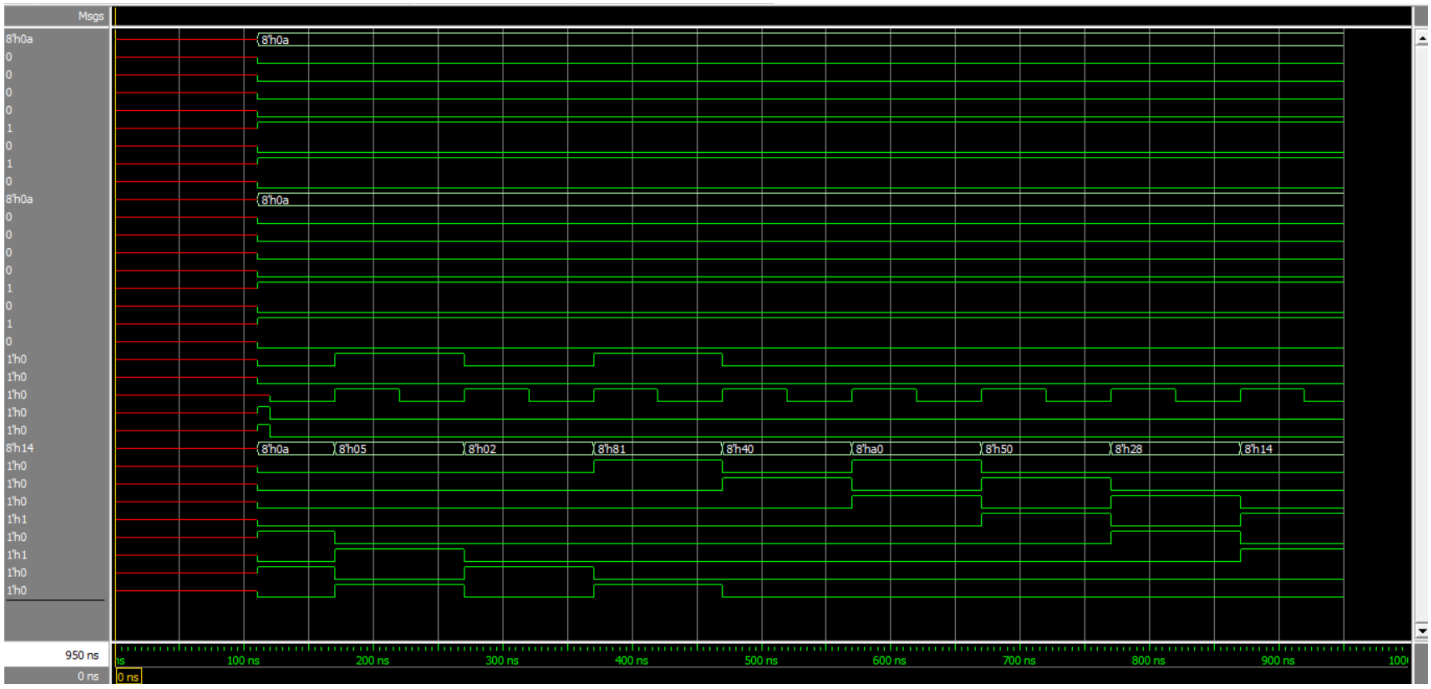


Fig.4.1 – Verilog result

Input A – 10
Input B – 10
Output – 20 (14 in hexadecimal) at time 900ns as shown in the figure.