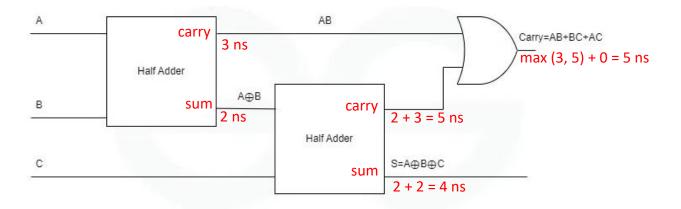
تمرین سری سوم DSD

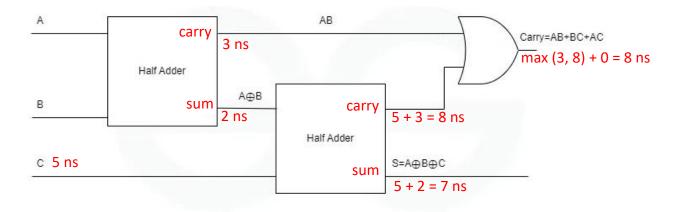
متین باقری 402105727

محاسبه تئوري تأخير جمع كننده:

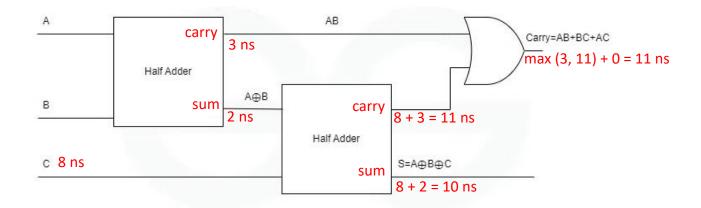
برای محاسبه تاخیر جمع کننده ۴ بیتی، ابتدا ساختار یک FA را در نظر می گیریم:



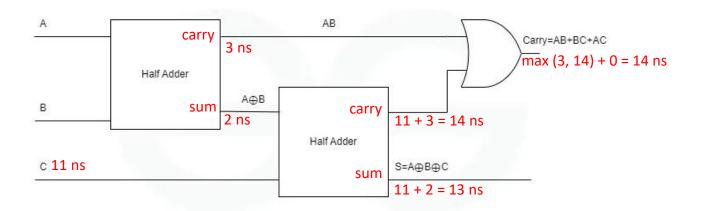
تاخیر آماده سازی هر داده در شکل بالا نوشته شده است. سپس FA دوم که FA جالا را به عنوان ورودی می گنیم:



همین روند را برای سومین FA ادامه میدهیم:



و نهایتا آخرین FA:



زمان آماده سازی هر یک از بیت های جواب نهایی به این صورت است:

Bit	Carry	S3	S2	S1	S0
Delay (ns)	14	13	10	7	4

max(14, 13, 10, 7, 4) = 14

بنابراین جواب نهایی پس از گذشت حداکثر 14ns آماده است.

گزارش و مقایسه نتایج شبیهسازی با محاسبات تئوری:

کد وریلاگ نوشته شده به شرح زیر است:

```
`timescale 1ns/1ns
module HalfAdder(input A, B, output Sum, Carry);
    xor #2 (Sum, A, B); // Sum with 2ns delay
    and #3 (Carry, A, B); // Carry with 3ns delay
endmodule
`timescale 1ns/1ns
module FullAdder(input A, B, Cin, output S, Cout);
   wire sum1, carry1, carry2;
   HalfAdder HA1(.A(A), .B(B), .Sum(sum1), .Carry(carry1));
   HalfAdder HA2(.A(sum1), .B(Cin), .Sum(S), .Carry(carry2));
    or #0 (Cout, carry1, carry2);// OR gate with Ons delay
endmodule
module adder(input cin, input [3:0] a, b, output cout, output [3:0] s);
   wire [2:0] carry;
    FullAdder FAO(.A(a[0]), .B(b[0]), .Cin(cin), .S(s[0]), .Cout(carry[0]));
    FullAdder \ FA1(.A(a[1]), \ .B(b[1]), \ .Cin(carry[0]), \ .S(s[1]), \ .Cout(carry[1]));
    FullAdder FA2(.A(a[2]), .B(b[2]), .Cin(carry[1]), .S(s[2]), .Cout(carry[2]));
    FullAdder FA3(.A(a[3]), .B(b[3]), .Cin(carry[2]), .S(s[3]), .Cout(cout));
endmodule
`timescale 1ns/1ns
module test_bench();
    reg [3:0] A, B;
    reg Cin;
   wire [3:0] Sum;
   wire Cout;
   reg [4:0] expected_result;
   integer total_tests = 0;
    integer successful_tests = 0;
    realtime max_delay = 0;
   realtime start_time, end_time;
    adder uut(.a(A), .b(B), .cin(Cin), .s(Sum), .cout(Cout));
    task check_result;
        input [3:0] a_val, b_val;
        input cin_val;
        reg [4:0] local_expected;
        realtime local_start, local_end;
        begin
            A = a_val;
            B = b_val;
            Cin = cin val;
            local_expected = A + B + Cin;
            local_start = $realtime;
            fork
                begin
                    wait({Cout, Sum} === local_expected);
```

```
local_end = $realtime;
                    disable timeout_block;
                end
                begin : timeout_block
                    #50;
                    $display("Warning: Timeout at %0tns for A=%b, B=%b, Cin=%b",
                            $time, A, B, Cin);
                    local end = $realtime;
                end
            join_any
            disable fork;
            if ((local_end - local_start) > max_delay) begin
                max_delay = local_end - local_start;
            end
            total_tests = total_tests + 1;
            if ({Cout, Sum} === local expected) begin
                successful tests = successful tests + 1;
                $display("PASS: A=%b, B=%b, Cin=%b => Sum=%b, Cout=%b (Delay: %0tns)",
                        A, B, Cin, Sum, Cout, local_end - local_start);
            end else begin
                $display("FAIL: A=%b, B=%b, Cin=%b => Got {Cout=%b, Sum=%b}, Expected
{Cout=%b, Sum=%b}",
                        A, B, Cin, Cout, Sum, local_expected[4], local_expected[3:0]);
            end
        end
    endtask
    initial begin
        #10;
        // Test all combinations
        for (integer i = 0; i < 16; i = i + 1) begin
            for (integer j = 0; j < 16; j = j + 1) begin
                        check_result(i, j, 1);
                        check_result(i, j, 0);
            end
        end
        $display("\nTest Results:");
        $display("Successful calculations: %0d / %0d", successful_tests, total_tests);
        $display("Maximum delay observed: %0.3f ns", max delay);
        $display("Theoretical worst-case delay: 14 ns");
    end
endmodule
```

خروجی کد بالا در فایل result.txt موجود است و انتهای آن به این صورت است:

```
# Test Results:
# Successful calculations: 512 / 512
# Maximum delay observed: 14.000 ns
# Theoretical worst-case delay: 14 ns
```

که نشان می دهد بیشترین تاخیر در شبیه سازی انجام شده (که تمام ۵۱۲ حالت ورودی ممکن را درنظر میگیرد) برابر ۱۴ نانوثانیه است و با محاسبات تئوری ما همخوانی دارد. همچنین در حین اجرای برنامه صحت خروجی جمع کننده بررسی می شود و همانطور که مشاهده می شود، تمام ورودی ها به خروجی صحیح منجر شده اند.