# DAY-27
# #100DAYSOFRTL

## PROBLEM STATEMENT:--

1. You are provided with a BCD (binary-coded decimal) one-digit adder named `bcd_fadd` that adds two BCD digits and carry-in, and produces a sum and carry-out.

```
module bcd_fadd (
    input [3:0] a,
    input [3:0] b,
    input     cin,
    output    cout,
    output [3:0] sum );
```

Instantiate 4 copies of `bcd_fadd` to create a 4-digit BCD ripple-carry adder. Your adder should add two 4-digit BCD numbers (packed into 16-bit vectors) and a carry-in to produce a 4-digit sum and carry out.

### Write your solution here

[Load a previous submission] ▾  **Load**

```verilog
1  module top_module (
2    input [15:0] a, b,
3    input cin,
4    output cout,
5    output [15:0] sum );
6    wire [3:0] stage_sum[3:0];
7    wire [3:0] stage_carry[3:0];
8
9    bcd_fadd stage0 (a[3:0], b[3:0], cin, stage_carry[0], stage_sum[0]);
10   bcd_fadd stage1 (a[7:4], b[7:4], stage_carry[0], stage_carry[1], stage_sum[1]);
11   bcd_fadd stage2 (a[11:8], b[11:8], stage_carry[1], stage_carry[2], stage_sum[2]);
12   bcd_fadd stage3 (a[15:12], b[15:12], stage_carry[2], cout, stage_sum[3]);
13
14   assign sum = {stage_sum[3], stage_sum[2], stage_sum[1], stage_sum[0]};
15
16 endmodule
17
```

**Submit**  **Submit (new window)**

Upload a source file... ≫

# Status: Success!

You have solved 61 problems.

## Timing diagrams for selected test cases

These are timing diagrams from some of the test cases we used. They may help you debug your circuit. The diagrams show inputs to the circuit, outputs from your circuit, and the expected reference outputs. The "Mismatch" trace shows which cycles your outputs don't match the reference outputs (0 = correct, 1 = incorrect).

**BCD addition**