**5. Design Verilog HDL to implement Decimal adder.**

**Verilog Code**

//module declaration with inputs and outputs

**module** decimaladder(a,b,carry\_in,sum,carry);

//declare the inputs and outputs of the module with their sizes.

**input** [3:0] a,b;

**input** carry\_in;

**output** [3:0] sum;

**output** carry;

//Internal variables

**reg** [4:0] sum\_temp;

**reg** [3:0] sum;

**reg** carry;

//always block for doing the addition

**always** @(a,b,carry\_in)

**begin**

sum\_temp = a+b+carry\_in; //add all the inputs

**if**(sum\_temp > 9)

**begin**

sum\_temp = sum\_temp+6; //add 6, if result is more than 9.

carry = 1; //set the carry output

sum = sum\_temp[3:0];

**end**

**else**

**begin**

carry = 0;

sum = sum\_temp[3:0];

**end**

**end**

**endmodule**

**Code for testbench**

**module** testbenchdadder;

// Inputs

**reg** [3:0] a;

**reg** [3:0] b;

**reg** carry\_in;

// Outputs

**wire** [3:0] sum;

**wire** carry;

// Instantiate the Unit Under Test (UUT)

**decimaladder** uut (.a(a),.b(b),.carry\_in(carry\_in),.sum(sum), .carry(carry));

**initial begin**

**$dumpfile**("dump.vcd");

**$dumpvars(1);**

**end**

**initial begin**

// Apply Inputs

a = 0; b = 0; carry\_in = 0;

#100 a = 6; b = 9; carry\_in = 0;

#100 a = 3; b = 3; carry\_in = 1;

#100 a = 4; b = 5; carry\_in = 0;

#100 a = 8; b = 2; carry\_in = 0;

#100 a = 9; b = 9; carry\_in = 1;

#100;

**$finish;**

**end**

**endmodule**