移位相加乘法器（Debug 后）

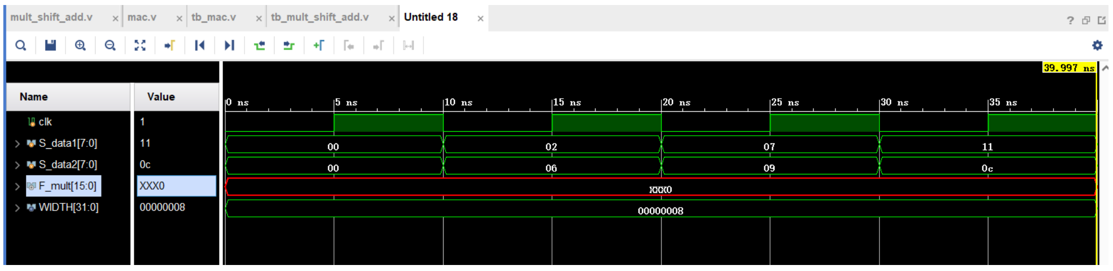
```
module mult_shift_add #(parameter WIDTH = 8) (
    input [WIDTH - 1 : 0] S_data1,
    input [WIDTH - 1 : 0] S_data2,
    output reg [2 * WIDTH - 1 : 0] F_mult
    );

    reg [2 * WIDTH - 1 : 0] temp;
    //wire [WIDTH - 1 : 0] S_data1, S_data2;
    reg [2 * WIDTH - 1 : 0] S_data2_temp;
    integer index;
    reg [2 * WIDTH - 1 : 0] result;
    always @(*)
    begin
        F_mult = 0;//需置零，否则是 XXX0
//          result = 0;
        S_data2_temp = {{WIDTH{1'b0}}, S_data2}; //Expand
        $monitor("S_data2_temp = %d", S_data2_temp);
        for (index = 0; index < WIDTH; index = index + 1)
        begin
            temp = {2 * WIDTH{S_data1[index]}} & {2*WIDTH{S_data2_temp}};
            F_mult = F_mult + (temp << index);//移位运算优先级比+-*/低，要加括号
//              $monitor("index = %d", index);
            $monitor("F_mult = %d", F_mult);
            $monitor("temp = %d", temp);
            $monitor("2 * WIDTH{S_data1[index]} %d", {2 * WIDTH{S_data1[0]}});
            $monitor("1111 & 1000 = %d", 4'b1111 & 4'b1000);
        end
//          $monitor("S_data1 = %d", S_data1);
//          $monitor("S_data2 = %d", S_data2_temp);
    end
endmodule
```

Testbench:

```
module tb_mult_shift_add();
parameter WIDTH = 8;
reg clk;
reg [WIDTH - 1 : 0] S_data1,S_data2;
wire [2 * WIDTH - 1 : 0] F_mult;
initial begin
```

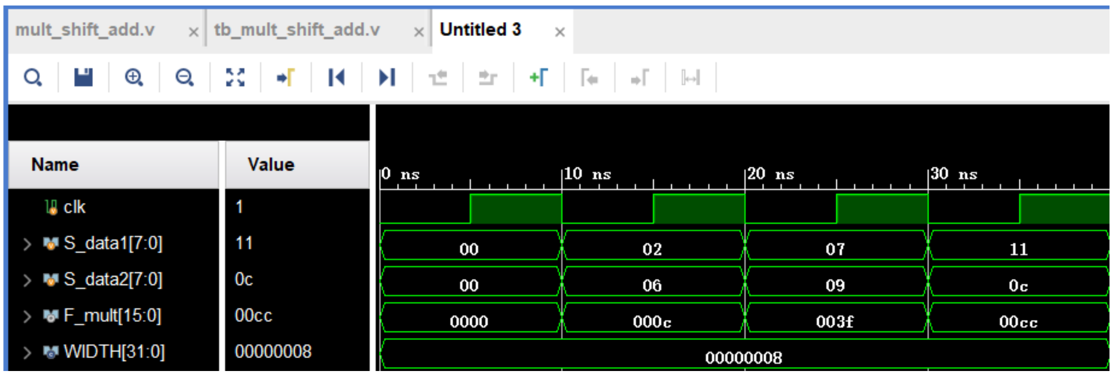
```
        clk = 0;
        S_data1 = 0;
        S_data2 = 0;
        #10
        S_data1 = 2;
        S_data2 = 6;
        #10
        S_data1 = 7;
        S_data2 = 9;
        #10
        S_data1 = 17;
        S_data2 = 12;
        #10 $stop;
//      #5 repeat (5) @(posedge clk)
//      begin
//              S_data1 <= ($random);
//              S_data2 <= ($random);
//      end
end
always #5 clk = ~clk;
mult_shift_add #(WIDTH) mult_shift_adder(S_data1, S_data2, F_mult);
endmodule
```

仿真:



Name	Value
clk	1
S_data1[7:0]	11
S_data2[7:0]	0c
F_mult[15:0]	XXX0
WIDTH[31:0]	00000008

Debug 后的仿真:



Name	Value
clk	1
S_data1[7:0]	11
S_data2[7:0]	0c
F_mult[15:0]	00cc
WIDTH[31:0]	00000008