

```
module multi_4bits_pipelining(mul_a, mul_b, clk, rst_n, mul_out);
```

```
    input [3:0] mul_a, mul_b;  
    input      clk;  
    input      rst_n;  
    output [7:0] mul_out;
```

```
    reg [7:0] mul_out;
```

```
    reg [7:0] stored0;  
    reg [7:0] stored1;  
    reg [7:0] stored2;  
    reg [7:0] stored3;
```

```
    reg [7:0] add01;  
    reg [7:0] add23;
```

```
    always @(posedge clk or negedge rst_n) begin
```

```
        if(!rst_n) begin  
            mul_out <= 0;  
            stored0 <= 0;  
            stored1 <= 0;  
            stored2 <= 0;  
            stored3 <= 0;  
            add01 <= 0;  
            add23 <= 0;
```

```
        end
```

```
        else begin
```

```
            stored0 <= mul_b[0]? {4'b0, mul_a} : 8'b0;  
            stored1 <= mul_b[1]? {3'b0, mul_a, 1'b0} : 8'b0;  
            stored2 <= mul_b[2]? {2'b0, mul_a, 2'b0} : 8'b0;  
            stored3 <= mul_b[3]? {1'b0, mul_a, 3'b0} : 8'b0;
```

```
            add01 <= stored1 + stored0;  
            add23 <= stored3 + stored2;
```

```
            mul_out <= add01 + add23;
```

```
        end
```

```
    end
```

```
endmodule
```