

## Data Flow Modelling

```
module full_adder(  
    input A,B,Cin,  
    output Sum,Cout  
);  
    assign Sum = (A^B^Cin);  
    assign Cout = (A&B)|(A&Cin)|(B&Cin);  
endmodule
```

## Test Bench

```
module full_adder_tb;  
    reg A,B,Cin;  
    wire Sum, Cout;  
  
    // Instantiate uut  
    full_adder uut(  
        .A(A),  
        .B(B),  
        .Cin(Cin),  
        .Sum(Sum),  
        .Cout(Cout)  
    );  
  
    initial begin  
        $display ("A B Cin | Sum Cout");  
  
        A=0;B=0;Cin=0;#10;
```

```
$display("%b %b %b | %b %b",A,B,Cin,Sum,Cout);
```

```
A=0;B=0;Cin=1;#10;
```

```
$display("%b %b %b | %b %b",A,B,Cin,Sum,Cout);
```

```
A=0;B=1;Cin=0;#10;
```

```
$display("%b %b %b | %b %b",A,B,Cin,Sum,Cout);
```

```
A=0;B=1;Cin=1;#10;
```

```
$display("%b %b %b | %b %b",A,B,Cin,Sum,Cout);
```

```
A=1;B=0;Cin=0;#10;
```

```
$display("%b %b %b | %b %b",A,B,Cin,Sum,Cout);
```

```
A=1;B=0;Cin=1;#10;
```

```
$display("%b %b %b | %b %b",A,B,Cin,Sum,Cout);
```

```
A=1;B=1;Cin=0;#10;
```

```
$display("%b %b %b | %b %b",A,B,Cin,Sum,Cout);
```

```
A=1;B=1;Cin=1;#10;
```

```
$display("%b %b %b | %b %b",A,B,Cin,Sum,Cout);
```

```
$finish;
```

```
end
```

```
endmodule
```