

```

1 //Aaron Hong (ahong02)
2 //Stephen Macris (smacris)
3 //EE469 Lab1
4
5 //This module adds two 32-bit numbers and a given carry-in value.
6
7 //Inputs: Two 32-bits A, B (inputs to be added), one 1-bit cin (carry-in value).
8 //Outputs: One 32-bit sum (sum), one 1-bit cout (carry-out value).
9 module fulladder32 (A, B, cin, sum, cout);
10
11     input logic [31:0] A;
12     input logic [31:0] B;
13     input logic cin;
14
15     output logic [31:0] sum;
16     output logic cout;
17
18     logic c0, c1, c2, c3, c4, c5, c6, c7, c8, c9, c10, c11, c12, c13, c14, c15, c16, c17, c18
19     , c19, c20, c21, c22, c23, c24, c25, c26, c27, c28, c29, c30;
20
21     //Chains together 32 instantiations of a 1-bit adder.
22     fulladder FA0(.A(A[0]), .B(B[0]), .cin(cin), .sum(sum[0]), .cout(c0));
23     fulladder FA1(.A(A[1]), .B(B[1]), .cin(c0), .sum(sum[1]), .cout(c1));
24     fulladder FA2(.A(A[2]), .B(B[2]), .cin(c1), .sum(sum[2]), .cout(c2));
25     fulladder FA3(.A(A[3]), .B(B[3]), .cin(c2), .sum(sum[3]), .cout(c3));
26     fulladder FA4(.A(A[4]), .B(B[4]), .cin(c3), .sum(sum[4]), .cout(c4));
27     fulladder FA5(.A(A[5]), .B(B[5]), .cin(c4), .sum(sum[5]), .cout(c5));
28     fulladder FA6(.A(A[6]), .B(B[6]), .cin(c5), .sum(sum[6]), .cout(c6));
29     fulladder FA7(.A(A[7]), .B(B[7]), .cin(c6), .sum(sum[7]), .cout(c7));
30     fulladder FA8(.A(A[8]), .B(B[8]), .cin(c7), .sum(sum[8]), .cout(c8));
31     fulladder FA9(.A(A[9]), .B(B[9]), .cin(c8), .sum(sum[9]), .cout(c9));
32     fulladder FA10(.A(A[10]), .B(B[10]), .cin(c9), .sum(sum[10]), .cout(c10));
33     fulladder FA11(.A(A[11]), .B(B[11]), .cin(c10), .sum(sum[11]), .cout(c11));
34     fulladder FA12(.A(A[12]), .B(B[12]), .cin(c11), .sum(sum[12]), .cout(c12));
35     fulladder FA13(.A(A[13]), .B(B[13]), .cin(c12), .sum(sum[13]), .cout(c13));
36     fulladder FA14(.A(A[14]), .B(B[14]), .cin(c13), .sum(sum[14]), .cout(c14));
37     fulladder FA15(.A(A[15]), .B(B[15]), .cin(c14), .sum(sum[15]), .cout(c15));
38     fulladder FA16(.A(A[16]), .B(B[16]), .cin(c15), .sum(sum[16]), .cout(c16));
39     fulladder FA17(.A(A[17]), .B(B[17]), .cin(c16), .sum(sum[17]), .cout(c17));
40     fulladder FA18(.A(A[18]), .B(B[18]), .cin(c17), .sum(sum[18]), .cout(c18));
41     fulladder FA19(.A(A[19]), .B(B[19]), .cin(c18), .sum(sum[19]), .cout(c19));
42     fulladder FA20(.A(A[20]), .B(B[20]), .cin(c19), .sum(sum[20]), .cout(c20));
43     fulladder FA21(.A(A[21]), .B(B[21]), .cin(c20), .sum(sum[21]), .cout(c21));
44     fulladder FA22(.A(A[22]), .B(B[22]), .cin(c21), .sum(sum[22]), .cout(c22));
45     fulladder FA23(.A(A[23]), .B(B[23]), .cin(c22), .sum(sum[23]), .cout(c23));
46     fulladder FA24(.A(A[24]), .B(B[24]), .cin(c23), .sum(sum[24]), .cout(c24));
47     fulladder FA25(.A(A[25]), .B(B[25]), .cin(c24), .sum(sum[25]), .cout(c25));
48     fulladder FA26(.A(A[26]), .B(B[26]), .cin(c25), .sum(sum[26]), .cout(c26));
49     fulladder FA27(.A(A[27]), .B(B[27]), .cin(c26), .sum(sum[27]), .cout(c27));
50     fulladder FA28(.A(A[28]), .B(B[28]), .cin(c27), .sum(sum[28]), .cout(c28));
51     fulladder FA29(.A(A[29]), .B(B[29]), .cin(c28), .sum(sum[29]), .cout(c29));
52     fulladder FA30(.A(A[30]), .B(B[30]), .cin(c29), .sum(sum[30]), .cout(c30));
53     fulladder FA31(.A(A[31]), .B(B[31]), .cin(c30), .sum(sum[31]), .cout(cout));
54
55 endmodule
56
57 //Tests fulladder32.
58 module fulladder32_testbench ();
59
60     logic [31:0] A, B, sum;
61     logic cin, cout;
62
63     fulladder32 dut (A, B, cin, sum, cout);
64
65     integer i;
66     initial begin
67
68
69         A = 32'd15; B = 32'd16; cin = 1'b0; #10;
70         A = 32'd15; B = 32'd16; cin = 1'b1; #10;
71         A = 32'd15; B = 32'd163; cin = 1'b0; #10;
72         A = 32'd15; B = 32'd216; cin = 1'b0; #10;
73         A = 32'd15; B = 32'd146; cin = 1'b0; #10;
74
75     end //initial

```

76

77 endmodule