

1) adder32.v

Test:

```
initial begin
a=15; b=20; carry_in=0;
#`DELAY;
a=10; b=8; carry_in=1;
#`DELAY;
a=300; b=15; carry_in=1;
#`DELAY;
a=4294900000; b=67296; carry_in=1;
#`DELAY;
end
```

Result:

```

vsum 37> step -current
# time = 0, a = 15, b= 20, carry_in=0, sum = 35, carry_out=0
# time = 20, a = 10, b= 8, carry_in=1, sum = 19, carry_out=0
# time = 40, a = 300, b= 15, carry_in=1, sum = 316, carry_out=0
# time = 60, a =4294900000, b= 67296, carry_in=1, sum = 1, carry_out=1
VSM 38:

```

2) xor32.v

Test:

```
initial begin
a=5; b=3;
#`DELAY;
a=10; b=8;
#`DELAY;
a=300; b=15;
#`DELAY;
a=429400000; b=67296;
#`DELAY;
end
```

Result:

```
v$VM /> step -current  
# time = 0, a =00000000000000000000000000000101, b=0000000000000000000000000000011,  
# result =0000000000000000000000000000000110  
# time = 20, a =000000000000000000000000000001010, b=000000000000000000000000000001000,  
# result =00000000000000000000000000000000010  
# time = 40, a =00000000000000000000000000000100101100, b=000000000000000000000000000001111,  
# result =000000000000000000000000000000000100100011  
# time = 60, a =00011001100110000001111111000000, b=00000000000000010000011011100000,  
# result =0001100110011001000110010010000
```

3) sub32.v

Test:

```
initial begin
a=21; b=20;
#`DELAY;
a=10; b=8;
#`DELAY;
a=300; b=15;
#`DELAY;
a=45; b=4294967260;
#`DELAY;
end
```

Result:

```
VSIM 40> step -current
# time = 0, a =      21, b=      20, sum =      1, carry_out=1
# time = 20, a =      10, b=       8, sum =      2, carry_out=1
# time = 40, a =     300, b=      15, sum =    285, carry_out=1
# time = 60, a =      45, b=4294967260, sum =     81, carry_out=0
VSIM 41>
```

If carry_out == 1 that means no overflow, otherwise overflow.

4) mult32.v

Test:

```
initial
begin
#20;
a=32'd18; b=32'd5;
#5400;
a=32'd45; b=32'd12;
#5400;
$stop;
end
```

Result:

```
VSIM 45> step -current
# time = 0,a =      x, b =      x, result =      x
# time = 20,a =     18, b =      5, result =      x
# time = 5320,a =     18, b =      5, result =     90
# time = 5420,a =     45, b =     12, result =     90
# time = 10640,a =     45, b =     12, result =    540
# Break in Module mult32 test at C:/Users/BAIRAN SOLMAZ/Desktop/1801
```

Takes too much time to calculate,so that why delay is 5400

[illegible]

7)and32.v

Test:

```
initial begin
a=5; b=3;
#`DELAY;
a=10; b=8;
#`DELAY;
a=300; b=15;
#`DELAY;
a=429400000; b=67296;
#`DELAY;
end
```

Result:

```
VSIM 52> step -current
# time = 0, a = 5, b= 3, result = 1
# time = 20, a = 10, b= 8, result = 8
# time = 40, a = 300, b= 15, result = 12
# time = 60, a = 429400000, b= 67296, result = 1728
```

8)or32.v

Test:

```
initial begin
a=5; b=3;
#`DELAY;
a=10; b=8;
#`DELAY;
a=300; b=15;
#`DELAY;
a=429400000; b=67296;
#`DELAY;
end
```

Result:

```
VSIM 54> step -current
# time = 0, a = 5, b= 3, result = 7
# time = 20, a = 10, b= 8, result = 10
# time = 40, a = 300, b= 15, result = 303
# time = 60, a = 429400000, b= 67296, result = 429465568
```

9) mux32.v

Test:

```

initial begin
    a = 8'd15; b = 8'd0; c = 8'd0; d = 8'd0; e = 8'd0; f = 8'd0; g = 8'd0; h = 8'd0; op = 3'b000;
    #`DELAY;
    a = 8'd0; b = 8'd15; c = 8'd0; d = 8'd0; e = 8'd0; f = 8'd0; g = 8'd0; h = 8'd0; op = 3'b001;
    #`DELAY;
    a = 8'd0; b = 8'd0; c = 8'd15; d = 8'd0; e = 8'd0; f = 8'd0; g = 8'd0; h = 8'd0; op = 3'b010;
    #`DELAY;
    a = 8'd0; b = 8'd0; c = 8'd0; d = 8'd15; e = 8'd0; f = 8'd0; g = 8'd0; h = 8'd0; op = 3'b011;
    #`DELAY;
    a = 8'd0; b = 8'd0; c = 8'd0; d = 8'd0; e = 8'd15; f = 8'd0; g = 8'd0; h = 8'd0; op = 3'b100;
    #`DELAY;
    a = 8'd0; b = 8'd0; c = 8'd0; d = 8'd0; e = 8'd0; f = 8'd15; g = 8'd0; h = 8'd0; op = 3'b101;
    #`DELAY;
    a = 8'd0; b = 8'd0; c = 8'd0; d = 8'd0; e = 8'd0; f = 8'd0; g = 8'd15; h = 8'd0; op = 3'b110;
    #`DELAY;
    a = 8'd0; b = 8'd0; c = 8'd0; d = 8'd0; e = 8'd0; f = 8'd0; g = 8'd0; h = 8'd15; op = 3'b111;
end

```

Result:

```

VSI67> step -current
# time = 0, a = 15, b = 0, c = 0, d = 0, e = 0, f = 0, g = 0, h = 0, op=0, sum= 15
# time = 20, a = 0, b = 15, c = 0, d = 0, e = 0, f = 0, g = 0, h = 0, op=1, sum= 15
# time = 40, a = 0, b = 0, c = 15, d = 0, e = 0, f = 0, g = 0, h = 0, op=2, sum= 15
# time = 60, a = 0, b = 0, c = 0, d = 15, e = 0, f = 0, g = 0, h = 0, op=3, sum= 15
# time = 80, a = 0, b = 0, c = 0, d = 0, e = 15, f = 0, g = 0, h = 0, op=4, sum= 15
# time = 100, a = 0, b = 0, c = 0, d = 0, e = 0, f = 15, g = 0, h = 0, op=5, sum= 15
# time = 120, a = 0, b = 0, c = 0, d = 0, e = 0, f = 0, g = 15, h = 0, op=6, sum= 15
# time = 140, a = 0, b = 0, c = 0, d = 0, e = 0, f = 0, g = 0, h = 15, op=7, sum= 15

```

10) mult32.v / control.v

Test:

```

initial
begin
    #`DELAY rst=1;
    #`DELAY rst=0; in=0;
    #`DELAY in=1;
    #`DELAY endi=1;
    #`DELAY $stop;
end

```

Result:

```

VSI67> step -current
# time: 0, rst=0 in=x , endi:0, write=x, endo=x present:xx , next:xx
# time:20, rst=1 in=x , endi:0, write=0, endo=0 present:00 , next:xx
# time:40, rst=0 in=0 , endi:0, write=0, endo=0 present:00 , next:10
# time:60, rst=0 in=1 , endi:0, write=1, endo=0 present:01 , next:10
# time:80, rst=0 in=1 , endi:1, write=1, endo=1 present:01 , next:10

```

11) mult32.v/ datapath.v

Test:

```

10
11 initial
12     begin
13         #`DELAY mult=32'd12;inP={32'd0,32'd8};
14         #`DELAY write=0;
15         #`DELAY inP=outP;
16         #`DELAY write=0;
17         #`DELAY inP=outP;
18         #`DELAY write=0;
19         #`DELAY inP=outP;
20         #`DELAY write=1;
21         #`DELAY inP=outP;
22         #`DELAY write=0;
23         #`DELAY inP=outP;
24         #`DELAY inP=outP;
25         #`DELAY inP=outP;
26         #`DELAY inP=outP;
27         #`DELAY inP=outP;
28         #`DELAY inP=outP;
29         #`DELAY inP=outP;
30         #`DELAY inP=outP;
31         #`DELAY inP=outP;
32
33         #`DELAY inP=outP;
34         #`DELAY inP=outP;
35         #`DELAY inP=outP;
36         #`DELAY inP=outP;
37         #`DELAY inP=outP;
38         #`DELAY inP=outP;
39         #`DELAY inP=outP;
40         #`DELAY inP=outP;
41         #`DELAY inP=outP;
42         #`DELAY inP=outP;
43         #`DELAY inP=outP;
44         #`DELAY inP=outP;
45         #`DELAY inP=outP;
46         #`DELAY inP=outP;
47         #`DELAY inP=outP;
48         #`DELAY inP=outP;
49         #`DELAY inP=outP;
50
51     end

```

Result:

```

# time: 0, mult=      x ,Pl= x ,Result= x,write=x
# time:20, mult=     12 ,Pl= 0 ,Result= 4,write=x
# time:40, mult=     12 ,Pl= 0 ,Result= 4,write=0
# time:60, mult=     12 ,Pl= 0 ,Result= 2,write=0
# time:100, mult=    12 ,Pl= 0 ,Result= 1,write=0
# time:140, mult=    12 ,Pl= 0 ,Result= 0,write=0
# time:160, mult=    12 ,Pl= 0 ,Result= 0,write=1
# time:180, mult=    12 ,Pl= 6 ,Result= 0,write=1
# time:200, mult=    12 ,Pl= 6 ,Result= 0,write=0
# time:220, mult=    12 ,Pl= 3 ,Result=2147483648,write=0
# time:240, mult=    12 ,Pl= 1 ,Result=3221225472,write=0
# time:260, mult=    12 ,Pl= 0 ,Result=1610612736,write=0
# time:280, mult=    12 ,Pl= 0 ,Result=805306368,write=0
# time:300, mult=    12 ,Pl= 0 ,Result=402653184,write=0
# time:320, mult=    12 ,Pl= 0 ,Result=201326592,write=0
# time:340, mult=    12 ,Pl= 0 ,Result=100663296,write=0
# time:360, mult=    12 ,Pl= 0 ,Result=50331648,write=0
# time:380, mult=    12 ,Pl= 0 ,Result=25165824,write=0
# time:400, mult=    12 ,Pl= 0 ,Result=12582912,write=0
# time:420, mult=    12 ,Pl= 0 ,Result=6291456,write=0
# time:440, mult=    12 ,Pl= 0 ,Result=3145728,write=0
# time:460, mult=    12 ,Pl= 0 ,Result=1572864,write=0
# time:480, mult=    12 ,Pl= 0 ,Result=786432,write=0
# time:500, mult=    12 ,Pl= 0 ,Result=393216,write=0
# time:520, mult=    12 ,Pl= 0 ,Result=196608,write=0
# time:540, mult=    12 ,Pl= 0 ,Result=98304,write=0
# time:560, mult=    12 ,Pl= 0 ,Result=49152,write=0
# time:580, mult=    12 ,Pl= 0 ,Result=24576,write=0
# time:600, mult=    12 ,Pl= 0 ,Result=12288,write=0
# time:620, mult=    12 ,Pl= 0 ,Result=6144,write=0
# time:640, mult=    12 ,Pl= 0 ,Result=3072,write=0
# time:660, mult=    12 ,Pl= 0 ,Result=1536,write=0
# time:680, mult=    12 ,Pl= 0 ,Result=768,write=0
# time:700, mult=    12 ,Pl= 0 ,Result=384,write=0
# time:720, mult=    12 ,Pl= 0 ,Result=192,write=0
# time:740, mult=    12 ,Pl= 0 ,Result=96,write=0

```

Truth table and boolean expressions for control.v

S1	S0	Product	end		N1	N0	Write	Exit	
0	0	0	0		1	0	0	0	
0	0	0	1		1	0	0	1	
0	0	1	0		0	1	0	0	
0	0	1	1		0	1	0	1	
0	1	0	0		1	0	1	0	
0	1	0	1		1	0	1	1	
0	1	1	0		1	0	1	0	
0	1	1	1		1	0	1	1	
1	0	0	0		0	0	0	0	
1	0	0	1		1	1	0	1	
1	0	1	0		0	0	0	0	
1	0	1	1		1	1	0	1	
1	1	0	0		0	0	0	1	
1	1	0	1		0	0	0	1	
1	1	1	0		0	0	0	1	
1	1	1	1		0	0	0	1	
					N1= S1'Product' + S1'S0 + S1S0'Exit				
					N0= s1's0'product + s1s0'exit				
					Write = s1's0				
					Exit= s1s0 + end				

Start : 00

Add: 01

End: 11