CPE322
SIM 04 HW
Christopher Bero

texter_control

```
module texter_control (clk, sw, dash_dit, dc_error, space, tm_reset,
nxt_char, back_sp, nxt_bit, sp_load, out_char, out_space);

input clk, sw, dash_dit, dc_error, space;
output tm_reset, nxt_char, back_sp, nxt_bit, sp_load, out_char,
out_space;

reg [2:0] state, nextstate;
reg tm_reset, nxt_char, nxt_bit, back_sp, sp_load, out_char,
out_space;

initial
begin
state = 0;
nextstate = 0;
tm_reset = 0;
nxt_char = 0;
nxt_bit = 0;
back_sp = 0;
sp_load = 0;
out_char = 0;
out_space = 0;
end

always @(posedge clk)
begin
if (state == 0) begin
     if (sw) begin
          tm_reset = 1;
          nxt_char = 1;
          nextstate = 1;
          end
     else
          nextstate = 0;
end

else if (state == 1) begin
     if (sw) begin
          nextstate = 1;
     end
     else if (space) begin
     back_sp = 1;
     nextstate = 1;
     end
     else begin
          nxt_bit = 1;
          if (dash_dit) begin
               sp_load = 1;
          end
          nextstate = 2;
     end
end

else if (state == 2) begin
```

```
     tm_reset = 1;
     nextstate = 3;
end

else if (state == 3) begin
     if (sw) begin
          sp_load = 1;
          nextstate = 4;
     end
     else begin
          if (dash_dit) begin
               if (dc_error) begin
                    nextstate = 0;
               end
               else begin
                    out_char = 1;
                    nextstate = 5;
               end
          end
          else begin
               nextstate = 3;
          end
     end
end

else if (state == 4) begin
     tm_reset = 1;
     nextstate = 1;
end

else if (state == 5) begin
     if (sw) begin
          nxt_char = 1;
          tm_reset = 1;
          nextstate = 1;
     end
     else begin
          if (space) begin
               out_space = 1;
               nextstate = 0;
          end
          else begin
               nextstate = 5;
          end
     end
end
state = nextstate;
end

endmodule
```

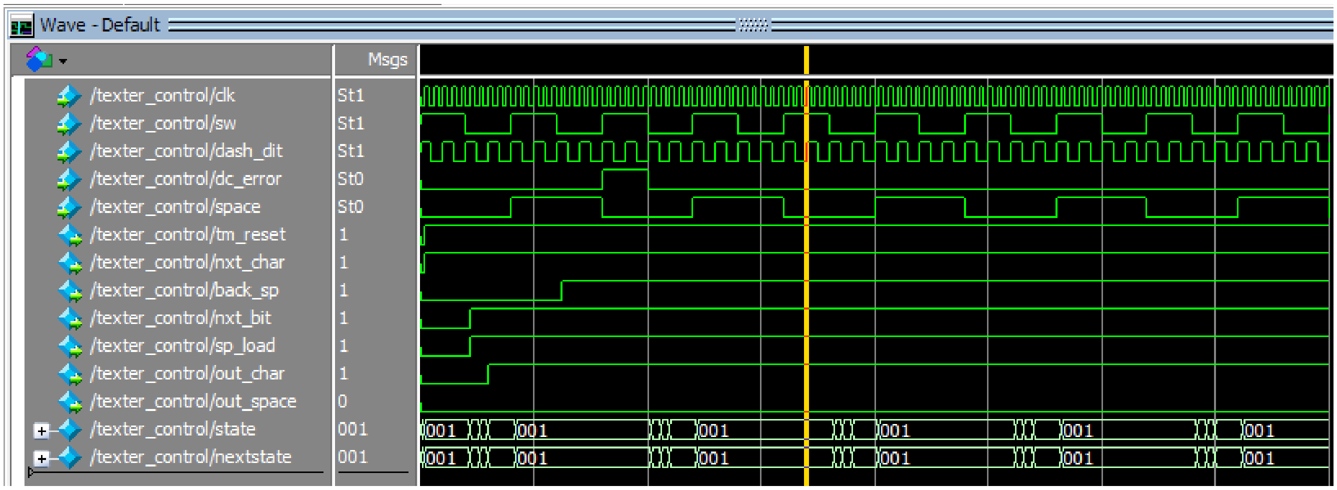
Stimulus

```
#
# Stimulus
# [0] is LSB
# SUB_ADD: 0=add (A+B), 1=subtract (A-B)
# Library: cycloneive_ver
#

#add list Bin reset clk Bout

#add wave Bin reset clk Bout

force dc_error 0 0, 1 800, 0 1000

force dash_dit 1 0, 0 50 -repeat 100

force space 0 0, 1 400 -repeat 800

force sw 1 0, 0 200 -repeat 400

force clk 0 0, 1 20 -repeat 40

run 4000
```

Output

List - Default

ps delta	/texter_control/clk	/texter_control/tm_reset	/texter_control/out_char	/texter_control/sw	/texter_control/nxt_char	/texter_control/out_space	/texter_control/dash_dit	/texter_control/back_sp	/texter_control/state	/texter_control/dc_error	/texter_control/nxt_bit	/texter_control/nextstate	/texter_control/space	/texter_control/sp_load
0 +0	St0	St1	St1	St0	St0	0	0	0	0	0	0	0	000	000
20 +0	St1	St1	St1	St0	St0	1	1	0	0	0	0	0	001	001
40 +0	St0	St1	St1	St0	St0	1	1	0	0	0	0	0	001	001
50 +0	St0	St1	St0	St0	St0	1	1	0	0	0	0	0	001	001
60 +0	St1	St1	St0	St0	St0	1	1	0	0	0	0	0	001	001
80 +0	St0	St1	St0	St0	St0	1	1	0	0	0	0	0	001	001
100 +0	St1	St1	St1	St0	St0	1	1	0	0	0	0	0	001	001
120 +0	St0	St1	St1	St0	St0	1	1	0	0	0	0	0	001	001
140 +0	St1	St1	St1	St0	St0	1	1	0	0	0	0	0	001	001
150 +0	St1	St1	St0	St0	St0	1	1	0	0	0	0	0	001	001
160 +0	St0	St1	St0	St0	St0	1	1	0	0	0	0	0	001	001
180 +0	St1	St1	St0	St0	St0	1	1	0	0	0	0	0	001	001
200 +0	St0	St0	St1	St0	St0	1	1	0	0	0	0	0	001	001
220 +0	St1	St0	St1	St0	St0	1	1	0	1	1	0	0	010	010
240 +0	St0	St0	St1	St0	St0	1	1	0	1	1	0	0	010	010
250 +0	St0	St0	St0	St0	St0	1	1	0	1	1	0	0	010	010
260 +0	St1	St0	St0	St0	St0	1	1	0	1	1	0	0	011	011
280 +0	St0	St0	St0	St0	St0	1	1	0	1	1	0	0	011	011
300 +0	St1	St0	St1	St0	St0	1	1	0	1	1	1	0	101	101
320 +0	St0	St0	St1	St0	St0	1	1	0	1	1	1	0	101	101
340 +0	St1	St0	St1	St0	St0	1	1	0	1	1	1	0	101	101
350 +0	St1	St0	St0	St0	St0	1	1	0	1	1	1	0	101	101
360 +0	St0	St0	St0	St0	St0	1	1	0	1	1	1	0	101	101
380 +0	St1	St0	St0	St0	St0	1	1	0	1	1	1	0	101	101
400 +0	St0	St1	St1	St0	St1	1	1	0	1	1	1	0	101	101
420 +0	St1	St1	St1	St0	St1	1	1	0	1	1	1	0	001	001
440 +0	St0	St1	St1	St0	St1	1	1	0	1	1	1	0	001	001
450 +0	St0	St1	St0	St0	St1	1	1	0	1	1	1	0	001	001
460 +0	St1	St1	St0	St0	St1	1	1	0	1	1	1	0	001	001
480 +0	St0	St1	St0	St0	St1	1	1	0	1	1	1	0	001	001
500 +0	St1	St1	St1	St0	St1	1	1	0	1	1	1	0	001	001
520 +0	St0	St1	St1	St0	St1	1	1	0	1	1	1	0	001	001
540 +0	St1	St1	St1	St0	St1	1	1	0	1	1	1	0	001	001
550 +0	St1	St1	St0	St0	St1	1	1	0	1	1	1	0	001	001
560 +0	St0	St1	St0	St0	St1	1	1	0	1	1	1	0	001	001
580 +0	St1	St1	St0	St0	St1	1	1	0	1	1	1	0	001	001
600 +0	St0	St0	St1	St0	St1	1	1	0	1	1	1	0	001	001
620 +0	St1	St0	St1	St0	St1	1	1	1	1	1	1	0	001	001
640 +0	St0	St0	St1	St0	St1	1	1	1	1	1	1	0	001	001
650 +0	St0	St0	St0	St0	St1	1	1	1	1	1	1	0	001	001
660 +0	St1	St0	St0	St0	St1	1	1	1	1	1	1	0	001	001
680 +0	St0	St0	St0	St0	St1	1	1	1	1	1	1	0	001	001
700 +0	St1	St0	St1	St0	St1	1	1	1	1	1	1	0	001	001
720 +0	St0	St0	St1	St0	St1	1	1	1	1	1	1	0	001	001
740 +0	St1	St0	St1	St0	St1	1	1	1	1	1	1	0	001	001
750 +0	St1	St0	St0	St0	St1	1	1	1	1	1	1	0	001	001



Flow Status	Successful - Tue Apr 28 10:19:52 2015
Quartus II 32-bit Version	13.0.1 Build 232 06/12/2013 SP 1 SJ Full Version
Revision Name	texter_control
Top-level Entity Name	texter_control
Family	Cyclone IV E
Device	EP4CE115F29C7
Timing Models	Final
Total logic elements	2 / 114,480 (< 1 %)
Total combinational functions	2 / 114,480 (< 1 %)
Dedicated logic registers	2 / 114,480 (< 1 %)
Total registers	2
Total pins	12 / 529 (2 %)
Total virtual pins	0
Total memory bits	0 / 3,981,312 (0 %)
Embedded Multiplier 9-bit elements	0 / 532 (0 %)
Total PLLs	0 / 4 (0 %)