```
1  ----------------------------------------------------------------------------------
2  -- Company: HSRW
3  -- Engineer: Andy Stamm
4  --
5  -- Create Date:    08:49:12 12/17/2015
6  -- Design Name:
7  -- Module Name:    Saving the switch state
8  -- Project Name:  Lab 3.3 & 3.4
9  -- Target Devices:
10 -- Tool versions:
11 -- Description:
12 --
13 -- Dependencies:
14 --
15 -- Revision:
16 -- Revision 0.01 - File Created
17 -- Additional Comments:
18 --
19 ----------------------------------------------------------------------------------
20 -- Uncomment the following library declaration if using
21 -- arithmetic functions with Signed or Unsigned values
22 --use IEEE.NUMERIC_STD.ALL;
23
24 -- Uncomment the following library declaration if instantiating
25 -- any Xilinx primitives in this code.
26 --library UNISIM;
27 --use UNISIM.VComponents.all;
28
29 library IEEE;
30 use IEEE.STD_LOGIC_1164.ALL;
31 use IEEE.STD_LOGIC_ARITH.ALL;
32 use IEEE.STD_LOGIC_UNSIGNED.ALL;
33
34 entity sr_latch is
35    Port( --clk: in std_logic;
36       SW0, SW1, SW2, SW3, SW4, SW5, SW6: in std_logic;
37       BTN0, BTN3: in std_logic;
38       LED0, LED1, LED2, LED3, LED4, LED5, LED6: out std_logic);
39    end entity sr_latch;
40
41 architecture Behavioral of sr_latch is
42    signal a: std_logic_vector (6 downto 0);
43       begin
44       process
45          begin
46          if BTN0='1' then
47             a(0)<=SW0;
48             a(1)<=SW1;
49             a(2)<=SW2;
50             a(3)<=SW3;
51             a(4)<=SW4;
52             a(5)<=SW5;
53             a(6)<=SW6;
54          end if;
55
56          if BTN3='1' then
57             LED0 <= a(0);
```

```
58                LED1 <= a(1);
59                LED2 <= a(2);
60                LED3 <= a(3);
61                LED4 <= a(4);
62                LED5 <= a(5);
63                LED6 <= a(6);
64             end if;
65
66             if BTN0='0' and BTN3='0' then
67                LED0 <= SW0;
68                LED1 <= SW1;
69                LED2 <= SW2;
70                LED3 <= SW3;
71                LED4 <= SW4;
72                LED5 <= SW5;
73                LED6 <= SW6;
74             end if;
75          end process;
76
77    end Behavioral;
78
```