

```
1  -----
2  -- Company:
3  -- Engineer:
4  --
5  -- Create Date:    13:56:46 11/29/2018
6  -- Design Name:
7  -- Module Name:    BikeLockDebug - Behavioral
8  -- Project Name:
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 library IEEE;
21 use IEEE.STD_LOGIC_1164.ALL;
22
23 -- Uncomment the following library declaration if using
24 -- arithmetic functions with Signed or Unsigned values
25 --use IEEE.NUMERIC_STD.ALL;
26
27 -- Uncomment the following library declaration if instantiating
28 -- any Xilinx primitives in this code.
29 --library UNISIM;
30 --use UNISIM.VComponents.all;
31
32 entity BikeLockDebug is
33     Port( clk,H,reset : in STD_LOGIC; -- Both H and reset are active high
34           X : in STD_LOGIC_VECTOR (4 DOWNTO 0);
35           L,A,R_state,L_state,A_state : out STD_LOGIC
36           );
37 end BikeLockDebug;
38
39 architecture Behavioral of BikeLockDebug is
40     Type state_type is (s0,s1,s2,s3,s4,s5,s6,s7,s8,s9,s10,s11,s12,s13,s14,s15,s16,s17,
41 s18,s19,
42 s20,s21,s22,s23,s24,s25,s26,s27,s28,s29,s30,s31,s32,s33,s34,
43 s35,s36,s37,s38,s39,
44 s40,s41,s42,s43,s44,s45,s46,s47,s48,s49,s50,s51,s52,s53,s54,
45 s55,s56,s57,s58,s59);
46
47     Signal p_state,n_state : state_type;
48
49 begin
50     Process(clk,reset)
51     begin
52         if(reset='1') then
53             p_state <= s0;
54         elsif(clk' event and clk='1') then
55             p_state <= n_state;
56         end if;
57     end process;
58 end process;
```

```
55
56     Process (X,H,p_state)
57     begin
58         case h is
59             when '1' =>
60                 case p_state is
61
62                     --This is where the Unarmed or Unlocked state begins
63                     when s0 =>
64                         case X is
65                             when "00000" => n_state <= s0; L<='0'; A<='0';
66                             when "00001" => n_state <= s0; L<='0'; A<='0';
67                             when "00010" => n_state <= s0; L<='0'; A<='0';
68                             when "00100" => n_state <= s1; L<='0'; A<='0';
69                             when "01000" => n_state <= s0; L<='0'; A<='0';
70                             when "10000" => n_state <= s0; L<='0'; A<='0';
71                             when others =>
72                                 end case;
73                         when s1 =>
74                             case X is
75                                 when "00000" => n_state <= s2; L<='0'; A<='0';
76                                 when "00100" => n_state <= s1; L<='0'; A<='0';
77                                 when others =>
78                                     end case;
79                         when s2 =>
80                             case X is
81                                 when "00000" => n_state <= s2; L<='0'; A<='0';
82                                 when "00001" => n_state <= s3; L<='0'; A<='0';
83                                 when "00010" => n_state <= s59; L<='0'; A<='0';
84                                 when "00100" => n_state <= s59; L<='0'; A<='0';
85                                 when "01000" => n_state <= s59; L<='0'; A<='0';
86                                 when "10000" => n_state <= s59; L<='0'; A<='0';
87                                 when others =>
88                                     end case;
89                         when s3 =>
90                             case X is
91                                 when "00000" => n_state <= s4; L<='0'; A<='0';
92                                 when "00001" => n_state <= s3; L<='0'; A<='0';
93                                 when others =>
94                                     end case;
95                         when s4 =>
96                             case X is
97                                 when "00000" => n_state <= s4; L<='0'; A<='0';
98                                 when "00001" => n_state <= s59; L<='0'; A<='0';
99                                 when "00010" => n_state <= s59; L<='0'; A<='0';
100                                when "00100" => n_state <= s59; L<='0'; A<='0';
101                                when "01000" => n_state <= s5; L<='0'; A<='0';
102                                when "10000" => n_state <= s59; L<='0'; A<='0';
103                                when others =>
104                                    end case;
105                         when s5 =>
106                             case X is
107                                 when "00000" => n_state <= s6; L<='0'; A<='0';
108                                 when "01000" => n_state <= s5; L<='0'; A<='0';
109                                 when others =>
110                                     end case;
111                         when s6 =>
```

```
112         case X is
113             when "00000" => n_state <= s6; L<='0'; A<='0';
114             when "00001" => n_state <= s59; L<='0'; A<='0';
115             when "00010" => n_state <= s59; L<='0'; A<='0';
116             when "00100" => n_state <= s59; L<='0'; A<='0';
117             when "01000" => n_state <= s7; L<='0'; A<='0';
118             when "10000" => n_state <= s59; L<='0'; A<='0';
119             when others =>
120         end case;
121     when s7 =>
122         case X is
123             when "00000" => n_state <= s8; L<='0'; A<='0';
124             when "01000" => n_state <= s7; L<='0'; A<='0';
125             when others =>
126         end case;
127     when s8 =>
128         case X is
129             when "00000" => n_state <= s8; L<='0'; A<='0';
130             when "00001" => n_state <= s59; L<='0'; A<='0';
131             when "00010" => n_state <= s9; L<='0'; A<='0';
132             when "00100" => n_state <= s59; L<='0'; A<='0';
133             when "01000" => n_state <= s59; L<='0'; A<='0';
134             when "10000" => n_state <= s59; L<='0'; A<='0';
135             when others =>
136         end case;
137     when s9 =>
138         case X is
139             when "00000" => n_state <= s10; L<='0'; A<='0';
140             when "00010" => n_state <= s9; L<='0'; A<='0';
141             when others =>
142         end case;
143     when s10 =>
144         case X is
145             when "00000" => n_state <= s10; L<='0'; A<='0';
146             when "00001" => n_state <= s59; L<='0'; A<='0';
147             when "00010" => n_state <= s59; L<='0'; A<='0';
148             when "00100" => n_state <= s59; L<='0'; A<='0';
149             when "01000" => n_state <= s59; L<='0'; A<='0';
150             when "10000" => n_state <= s11; L<='0'; A<='0';
151             when others =>
152         end case;
153     when s11 =>
154         case X is
155             when "00000" => n_state <= s12; L<='0'; A<='0';
156             when "10000" => n_state <= s11; L<='0'; A<='0';
157             when others =>
158         end case;
159
160     --This is where the Armed or Locked state begins
161     --This is the 'correct input' path for the Armed or Locked State
162     when s12 =>
163         case X is
164             when "00000" => n_state <= s12; L<='1'; A<='0';
165             when "00001" => n_state <= s24; L<='1'; A<='0';
166             when "00010" => n_state <= s24; L<='1'; A<='0';
167             when "00100" => n_state <= s13; L<='1'; A<='0';
168             when "01000" => n_state <= s24; L<='1'; A<='0';
```

```
169         when "10000" => n_state <= s12; L<='1'; A<='0';
170         when others =>
171         end case;
172     when s13 =>
173         case X is
174             when "00000" => n_state <= s14; L<='1'; A<='0';
175             when "00100" => n_state <= s13; L<='1'; A<='0';
176             when others =>
177             end case;
178     when s14 =>
179         case X is
180             when "00000" => n_state <= s14; L<='1'; A<='0';
181             when "00001" => n_state <= s15; L<='1'; A<='0';
182             when "00010" => n_state <= s26; L<='1'; A<='0';
183             when "00100" => n_state <= s37; L<='1'; A<='0';
184             when "01000" => n_state <= s26; L<='1'; A<='0';
185             when "10000" => n_state <= s36; L<='1'; A<='0';
186             when others =>
187             end case;
188     when s15 =>
189         case X is
190             when "00000" => n_state <= s16; L<='1'; A<='0';
191             when "00001" => n_state <= s15; L<='1'; A<='0';
192             when others =>
193             end case;
194     when s16 =>
195         case X is
196             when "00000" => n_state <= s16; L<='1'; A<='0';
197             when "00001" => n_state <= s28; L<='1'; A<='0';
198             when "00010" => n_state <= s28; L<='1'; A<='0';
199             when "00100" => n_state <= s39; L<='1'; A<='0';
200             when "01000" => n_state <= s17; L<='1'; A<='0';
201             when "10000" => n_state <= s36; L<='1'; A<='0';
202             when others =>
203             end case;
204     when s17 =>
205         case X is
206             when "00000" => n_state <= s18; L<='1'; A<='0';
207             when "01000" => n_state <= s17; L<='1'; A<='0';
208             when others =>
209             end case;
210     when s18 =>
211         case X is
212             when "00000" => n_state <= s18; L<='1'; A<='0';
213             when "00001" => n_state <= s30; L<='1'; A<='0';
214             when "00010" => n_state <= s30; L<='1'; A<='0';
215             when "00100" => n_state <= s41; L<='1'; A<='0';
216             when "01000" => n_state <= s19; L<='1'; A<='0';
217             when "10000" => n_state <= s36; L<='1'; A<='0';
218             when others =>
219             end case;
220     when s19 =>
221         case X is
222             when "00000" => n_state <= s20; L<='1'; A<='0';
223             when "01000" => n_state <= s19; L<='1'; A<='0';
224             when others =>
225             end case;
```

```
226         when s20 =>
227             case X is
228                 when "00000" => n_state <= s20; L<='1'; A<='0';
229                 when "00001" => n_state <= s32; L<='1'; A<='0';
230                 when "00010" => n_state <= s21; L<='1'; A<='0';
231                 when "00100" => n_state <= s43; L<='1'; A<='0';
232                 when "01000" => n_state <= s32; L<='1'; A<='0';
233                 when "10000" => n_state <= s36; L<='1'; A<='0';
234                 when others =>
235             end case;
236         when s21 =>
237             case X is
238                 when "00000" => n_state <= s22; L<='1'; A<='0';
239                 when "00010" => n_state <= s21; L<='1'; A<='0';
240                 when others =>
241             end case;
242         when s22 =>
243             case X is
244                 when "00000" => n_state <= s22; L<='1'; A<='0';
245                 when "00001" => n_state <= s34; L<='1'; A<='0';
246                 when "00010" => n_state <= s34; L<='1'; A<='0';
247                 when "00100" => n_state <= s45; L<='1'; A<='0';
248                 when "01000" => n_state <= s34; L<='1'; A<='0';
249                 when "10000" => n_state <= s23; L<='1'; A<='0';
250                 when others =>
251             end case;
252         when s23 =>
253             case X is
254                 when "00000" => n_state <= s0; L<='0'; A<='0';
255                 when "10000" => n_state <= s23; L<='1'; A<='0';
256                 when others =>
257             end case;
258
259         --These is the 'failed input' path for the Armed state
260         when s24 =>
261             case X is
262                 when "00000" => n_state <= s25; L<='1'; A<='0';
263                 when "00001" => n_state <= s24; L<='1'; A<='0';
264                 when "00010" => n_state <= s24; L<='1'; A<='0';
265                 when "00100" => n_state <= s24; L<='1'; A<='0';
266                 when "01000" => n_state <= s24; L<='1'; A<='0';
267                 when "10000" => n_state <= s24; L<='1'; A<='0';
268                 when others =>
269             end case;
270         when s25 =>
271             case X is
272                 when "00000" => n_state <= s25; L<='1'; A<='0';
273                 when "00001" => n_state <= s26; L<='1'; A<='0';
274                 when "00010" => n_state <= s26; L<='1'; A<='0';
275                 when "00100" => n_state <= s37; L<='1'; A<='0';
276                 when "01000" => n_state <= s26; L<='1'; A<='0';
277                 when "10000" => n_state <= s36; L<='1'; A<='0';
278                 when others =>
279             end case;
280         when s26 =>
281             case X is
282                 when "00000" => n_state <= s27; L<='1'; A<='0';
```

```
283         when "00001" => n_state <= s26; L<='1'; A<='0';
284         when "00010" => n_state <= s26; L<='1'; A<='0';
285         when "01000" => n_state <= s26; L<='1'; A<='0';
286         when others =>
287             end case;
288     when s27 =>
289         case X is
290             when "00000" => n_state <= s27; L<='1'; A<='0';
291             when "00001" => n_state <= s28; L<='1'; A<='0';
292             when "00010" => n_state <= s28; L<='1'; A<='0';
293             when "00100" => n_state <= s39; L<='1'; A<='0';
294             when "01000" => n_state <= s28; L<='1'; A<='0';
295             when "10000" => n_state <= s36; L<='1'; A<='0';
296             when others =>
297                 end case;
298     when s28 =>
299         case X is
300             when "00000" => n_state <= s29; L<='1'; A<='0';
301             when "00001" => n_state <= s28; L<='1'; A<='0';
302             when "00010" => n_state <= s28; L<='1'; A<='0';
303             when "01000" => n_state <= s28; L<='1'; A<='0';
304             when others =>
305                 end case;
306     when s29 =>
307         case X is
308             when "00000" => n_state <= s29; L<='1'; A<='0';
309             when "00001" => n_state <= s30; L<='1'; A<='0';
310             when "00010" => n_state <= s30; L<='1'; A<='0';
311             when "00100" => n_state <= s41; L<='1'; A<='0';
312             when "01000" => n_state <= s30; L<='1'; A<='0';
313             when "10000" => n_state <= s36; L<='1'; A<='0';
314             when others =>
315                 end case;
316     when s30 =>
317         case X is
318             when "00000" => n_state <= s31; L<='1'; A<='0';
319             when "00001" => n_state <= s30; L<='1'; A<='0';
320             when "00010" => n_state <= s30; L<='1'; A<='0';
321             when "01000" => n_state <= s30; L<='1'; A<='0';
322             when others =>
323                 end case;
324     when s31 =>
325         case X is
326             when "00000" => n_state <= s31; L<='1'; A<='0';
327             when "00001" => n_state <= s32; L<='1'; A<='0';
328             when "00010" => n_state <= s32; L<='1'; A<='0';
329             when "00100" => n_state <= s43; L<='1'; A<='0';
330             when "01000" => n_state <= s32; L<='1'; A<='0';
331             when "10000" => n_state <= s36; L<='1'; A<='0';
332             when others =>
333                 end case;
334     when s32 =>
335         case X is
336             when "00000" => n_state <= s33; L<='1'; A<='0';
337             when "00001" => n_state <= s32; L<='1'; A<='0';
338             when "00010" => n_state <= s32; L<='1'; A<='0';
339             when "01000" => n_state <= s32; L<='1'; A<='0';
```

```
340         when others =>
341     end case;
342 when s33 =>
343     case X is
344     when "00000" => n_state <= s33; L<='1'; A<='0';
345     when "00001" => n_state <= s34; L<='1'; A<='0';
346     when "00010" => n_state <= s34; L<='1'; A<='0';
347     when "00100" => n_state <= s45; L<='1'; A<='0';
348     when "01000" => n_state <= s34; L<='1'; A<='0';
349     when "10000" => n_state <= s34; L<='1'; A<='0';
350     when others =>
351     end case;
352 when s34 =>
353     case X is
354     when "00000" => n_state <= s35; L<='1'; A<='1';
355     when "00001" => n_state <= s34; L<='1'; A<='0';
356     when "00010" => n_state <= s34; L<='1'; A<='0';
357     when "01000" => n_state <= s34; L<='1'; A<='0';
358     when "10000" => n_state <= s34; L<='1'; A<='0';
359     when others =>
360     end case;
361
362     --This state is the beginning of the Alarm states
363     when s35 =>
364         case X is
365         when "00000" => n_state <= s35; L<='1'; A<='1';
366         when "00001" => n_state <= s35; L<='1'; A<='1';
367         when "00010" => n_state <= s35; L<='1'; A<='1';
368         when "00100" => n_state <= s47; L<='1'; A<='1';
369         when "01000" => n_state <= s35; L<='1'; A<='1';
370         when "10000" => n_state <= s35; L<='1'; A<='1';
371         when others =>
372         end case;
373
374     --This is the Armed X4 reset state
375     when s36 =>
376         case X is
377         when "00000" => n_state <= s12; L<='1'; A<='0';
378         when "00100" => n_state <= s36; L<='1'; A<='0';
379         when "10000" => n_state <= s36; L<='1'; A<='0';
380         when others =>
381         end case;
382
383     --These are the states for Armed X2 resets
384     when s37 =>
385         case X is
386         when "00000" => n_state <= s38; L<='1'; A<='0';
387         when "00100" => n_state <= s37; L<='1'; A<='0';
388         when others =>
389         end case;
390     when s38 =>
391         case X is
392         when "00000" => n_state <= s38; L<='1'; A<='0';
393         when "00001" => n_state <= s28; L<='1'; A<='0';
394         when "00010" => n_state <= s28; L<='1'; A<='0';
395         when "00100" => n_state <= s36; L<='1'; A<='0';
396         when "01000" => n_state <= s28; L<='1'; A<='0';
```

```
397         when "10000" => n_state <= s36; L<='1'; A<='0';
398         when others =>
399         end case;
400     when s39 =>
401         case X is
402         when "00000" => n_state <= s40; L<='1'; A<='0';
403         when "00100" => n_state <= s39; L<='1'; A<='0';
404         when others =>
405         end case;
406     when s40 =>
407         case X is
408         when "00000" => n_state <= s40; L<='1'; A<='0';
409         when "00001" => n_state <= s30; L<='1'; A<='0';
410         when "00010" => n_state <= s30; L<='1'; A<='0';
411         when "00100" => n_state <= s36; L<='1'; A<='0';
412         when "01000" => n_state <= s30; L<='1'; A<='0';
413         when "10000" => n_state <= s36; L<='1'; A<='0';
414         when others =>
415         end case;
416     when s41 =>
417         case X is
418         when "00000" => n_state <= s42; L<='1'; A<='0';
419         when "00100" => n_state <= s41; L<='1'; A<='0';
420         when others =>
421         end case;
422     when s42 =>
423         case X is
424         when "00000" => n_state <= s42; L<='1'; A<='0';
425         when "00001" => n_state <= s32; L<='1'; A<='0';
426         when "00010" => n_state <= s32; L<='1'; A<='0';
427         when "00100" => n_state <= s36; L<='1'; A<='0';
428         when "01000" => n_state <= s32; L<='1'; A<='0';
429         when "10000" => n_state <= s36; L<='1'; A<='0';
430         when others =>
431         end case;
432     when s43 =>
433         case X is
434         when "00000" => n_state <= s44; L<='1'; A<='0';
435         when "00100" => n_state <= s43; L<='1'; A<='0';
436         when others =>
437         end case;
438     when s44 =>
439         case X is
440         when "00000" => n_state <= s44; L<='1'; A<='0';
441         when "00001" => n_state <= s34; L<='1'; A<='0';
442         when "00010" => n_state <= s34; L<='1'; A<='0';
443         when "00100" => n_state <= s36; L<='1'; A<='0';
444         when "01000" => n_state <= s34; L<='1'; A<='0';
445         when "10000" => n_state <= s34; L<='1'; A<='0';
446         when others =>
447         end case;
448     when s45 =>
449         case X is
450         when "00000" => n_state <= s46; L<='1'; A<='0';
451         when "00100" => n_state <= s45; L<='1'; A<='0';
452         when others =>
453         end case;
```



```
454         when s46 =>
455             case X is
456                 when "00000" => n_state <= s46; L<='1'; A<='0';
457                 when "00001" => n_state <= s34; L<='1'; A<='0';
458                 when "00010" => n_state <= s34; L<='1'; A<='0';
459                 when "00100" => n_state <= s36; L<='1'; A<='0';
460                 when "01000" => n_state <= s34; L<='1'; A<='0';
461                 when "10000" => n_state <= s34; L<='1'; A<='0';
462                 when others =>
463                     end case;
464
465         --These are the Alarm unlocking states
466         when s47 =>
467             case X is
468                 when "00000" => n_state <= s48; L<='1'; A<='1';
469                 when "00100" => n_state <= s47; L<='1'; A<='1';
470                 when others =>
471                     end case;
472         when s48 =>
473             case X is
474                 when "00000" => n_state <= s48; L<='1'; A<='1';
475                 when "00001" => n_state <= s49; L<='1'; A<='1';
476                 when "00010" => n_state <= s58; L<='1'; A<='1';
477                 when "00100" => n_state <= s58; L<='1'; A<='1';
478                 when "01000" => n_state <= s58; L<='1'; A<='1';
479                 when "10000" => n_state <= s58; L<='1'; A<='1';
480                 when others =>
481                     end case;
482         when s49 =>
483             case X is
484                 when "00000" => n_state <= s50; L<='1'; A<='1';
485                 when "00001" => n_state <= s49; L<='1'; A<='1';
486                 when others =>
487                     end case;
488         when s50 =>
489             case X is
490                 when "00000" => n_state <= s50; L<='1'; A<='1';
491                 when "00001" => n_state <= s58; L<='1'; A<='1';
492                 when "00010" => n_state <= s58; L<='1'; A<='1';
493                 when "00100" => n_state <= s58; L<='1'; A<='1';
494                 when "01000" => n_state <= s51; L<='1'; A<='1';
495                 when "10000" => n_state <= s58; L<='1'; A<='1';
496                 when others =>
497                     end case;
498         when s51 =>
499             case X is
500                 when "00000" => n_state <= s52; L<='1'; A<='1';
501                 when "01000" => n_state <= s51; L<='1'; A<='1';
502                 when others =>
503                     end case;
504         when s52 =>
505             case X is
506                 when "00000" => n_state <= s52; L<='1'; A<='1';
507                 when "00001" => n_state <= s58; L<='1'; A<='1';
508                 when "00010" => n_state <= s58; L<='1'; A<='1';
509                 when "00100" => n_state <= s58; L<='1'; A<='1';
510                 when "01000" => n_state <= s53; L<='1'; A<='1';
```

```
511         when "10000" => n_state <= s58; L<='1'; A<='1';
512         when others =>
513         end case;
514     when s53 =>
515         case X is
516         when "00000" => n_state <= s54; L<='1'; A<='1';
517         when "01000" => n_state <= s53; L<='1'; A<='1';
518         when others =>
519         end case;
520     when s54 =>
521         case X is
522         when "00000" => n_state <= s54; L<='1'; A<='1';
523         when "00001" => n_state <= s58; L<='1'; A<='1';
524         when "00010" => n_state <= s55; L<='1'; A<='1';
525         when "00100" => n_state <= s58; L<='1'; A<='1';
526         when "01000" => n_state <= s58; L<='1'; A<='1';
527         when "10000" => n_state <= s58; L<='1'; A<='1';
528         when others =>
529         end case;
530     when s55 =>
531         case X is
532         when "00000" => n_state <= s56; L<='1'; A<='1';
533         when "00010" => n_state <= s55; L<='1'; A<='1';
534         when others =>
535         end case;
536     when s56 =>
537         case X is
538         when "00000" => n_state <= s56; L<='1'; A<='1';
539         when "00001" => n_state <= s58; L<='1'; A<='1';
540         when "00010" => n_state <= s58; L<='1'; A<='1';
541         when "00100" => n_state <= s58; L<='1'; A<='1';
542         when "01000" => n_state <= s58; L<='1'; A<='1';
543         when "10000" => n_state <= s57; L<='1'; A<='1';
544         when others =>
545         end case;
546     when s57 =>
547         case X is
548         when "00000" => n_state <= s0; L<='0'; A<='0';
549         when "10000" => n_state <= s57; L<='1'; A<='1';
550         when others =>
551         end case;
552
553     --This is the Alarm reset state
554     when s58 =>
555         case X is
556         when "00000" => n_state <= s35; L<='1'; A<='1';
557         when "00001" => n_state <= s58; L<='1'; A<='1';
558         when "00010" => n_state <= s58; L<='1'; A<='1';
559         when "00100" => n_state <= s58; L<='1'; A<='1';
560         when "01000" => n_state <= s58; L<='1'; A<='1';
561         when "10000" => n_state <= s58; L<='1'; A<='1';
562         when others =>
563         end case;
564
565     --This is the Unarmed reset state and will lead to s0
566     when s59 =>
567         case X is
```

```
568         when "00000" => n_state <= s0; L<='0'; A<='0';
569         when "00001" => n_state <= s59; L<='0'; A<='0';
570         when "00010" => n_state <= s59; L<='0'; A<='0';
571         when "00100" => n_state <= s59; L<='0'; A<='0';
572         when "01000" => n_state <= s59; L<='0'; A<='0';
573         when "10000" => n_state <= s59; L<='0'; A<='0';
574         when others =>
575             end case;
576         when others =>
577             end case; --End of H = '1'
578
579     --This is when the lock is not latched
580     when '0' =>
581         case p_state is
582
583             -- Unlatched Unarmed states
584             when s0 => n_state <= s0; L<='0'; A<='0';
585             when s1 => n_state <= s0; L<='0'; A<='0';
586             when s2 => n_state <= s0; L<='0'; A<='0';
587             when s3 => n_state <= s0; L<='0'; A<='0';
588             when s4 => n_state <= s0; L<='0'; A<='0';
589             when s5 => n_state <= s0; L<='0'; A<='0';
590             when s6 => n_state <= s0; L<='0'; A<='0';
591             when s7 => n_state <= s0; L<='0'; A<='0';
592             when s8 => n_state <= s0; L<='0'; A<='0';
593             when s9 => n_state <= s0; L<='0'; A<='0';
594             when s10 => n_state <= s0; L<='0'; A<='0';
595             when s11 => n_state <= s0; L<='0'; A<='0';
596
597             --These are the unlatched Armed states
598             when s12 => n_state <= s35; L<='1'; A<='1';
599             when s13 => n_state <= s35; L<='1'; A<='1';
600             when s14 => n_state <= s35; L<='1'; A<='1';
601             when s15 => n_state <= s35; L<='1'; A<='1';
602             when s16 => n_state <= s35; L<='1'; A<='1';
603             when s17 => n_state <= s35; L<='1'; A<='1';
604             when s18 => n_state <= s35; L<='1'; A<='1';
605             when s19 => n_state <= s35; L<='1'; A<='1';
606             when s20 => n_state <= s35; L<='1'; A<='1';
607             when s21 => n_state <= s35; L<='1'; A<='1';
608             when s22 => n_state <= s35; L<='1'; A<='1';
609             when s23 => n_state <= s35; L<='1'; A<='1';
610             when s24 => n_state <= s35; L<='1'; A<='1';
611             when s25 => n_state <= s35; L<='1'; A<='1';
612             when s26 => n_state <= s35; L<='1'; A<='1';
613             when s27 => n_state <= s35; L<='1'; A<='1';
614             when s28 => n_state <= s35; L<='1'; A<='1';
615             when s29 => n_state <= s35; L<='1'; A<='1';
616             when s30 => n_state <= s35; L<='1'; A<='1';
617             when s31 => n_state <= s35; L<='1'; A<='1';
618             when s32 => n_state <= s35; L<='1'; A<='1';
619             when s33 => n_state <= s35; L<='1'; A<='1';
620             when s34 => n_state <= s35; L<='1'; A<='1';
621
622             --S35 is the beginning of the Alarm state
623             when s35 =>
624                 case X is
```

```
625         when "00000" => n_state <= s35; L<='1'; A<='1';
626         when "00001" => n_state <= s35; L<='1'; A<='1';
627         when "00010" => n_state <= s35; L<='1'; A<='1';
628         when "00100" => n_state <= s47; L<='1'; A<='1';
629         when "01000" => n_state <= s35; L<='1'; A<='1';
630         when "10000" => n_state <= s35; L<='1'; A<='1';
631         when others =>
632             end case;
633
634     --Unlatched Armed states continued
635     when s36 => n_state <= s35; L<='1'; A<='1';
636     when s37 => n_state <= s35; L<='1'; A<='1';
637     when s38 => n_state <= s35; L<='1'; A<='1';
638     when s39 => n_state <= s35; L<='1'; A<='1';
639     when s40 => n_state <= s35; L<='1'; A<='1';
640     when s41 => n_state <= s35; L<='1'; A<='1';
641     when s42 => n_state <= s35; L<='1'; A<='1';
642     when s43 => n_state <= s35; L<='1'; A<='1';
643     when s44 => n_state <= s35; L<='1'; A<='1';
644     when s45 => n_state <= s35; L<='1'; A<='1';
645     when s46 => n_state <= s35; L<='1'; A<='1';
646
647     --Unlatched Alarm States
648     when s47 =>
649         case X is
650             when "00000" => n_state <= s48; L<='1'; A<='1';
651             when "00100" => n_state <= s47; L<='1'; A<='1';
652             when others =>
653                 end case;
654         when s48 =>
655             case X is
656                 when "00000" => n_state <= s48; L<='1'; A<='1';
657                 when "00001" => n_state <= s49; L<='1'; A<='1';
658                 when "00010" => n_state <= s58; L<='1'; A<='1';
659                 when "00100" => n_state <= s58; L<='1'; A<='1';
660                 when "01000" => n_state <= s58; L<='1'; A<='1';
661                 when "10000" => n_state <= s58; L<='1'; A<='1';
662                 when others =>
663                     end case;
664             when s49 =>
665                 case X is
666                     when "00000" => n_state <= s50; L<='1'; A<='1';
667                     when "00001" => n_state <= s49; L<='1'; A<='1';
668                     when others =>
669                         end case;
670                 when s50 =>
671                     case X is
672                         when "00000" => n_state <= s50; L<='1'; A<='1';
673                         when "00001" => n_state <= s58; L<='1'; A<='1';
674                         when "00010" => n_state <= s58; L<='1'; A<='1';
675                         when "00100" => n_state <= s58; L<='1'; A<='1';
676                         when "01000" => n_state <= s51; L<='1'; A<='1';
677                         when "10000" => n_state <= s58; L<='1'; A<='1';
678                         when others =>
679                             end case;
680                 when s51 =>
681                     case X is
```

```
682         when "00000" => n_state <= s52; L<='1'; A<='1';
683         when "01000" => n_state <= s51; L<='1'; A<='1';
684         when others =>
685             end case;
686     when s52 =>
687         case X is
688             when "00000" => n_state <= s52; L<='1'; A<='1';
689             when "00001" => n_state <= s58; L<='1'; A<='1';
690             when "00010" => n_state <= s58; L<='1'; A<='1';
691             when "00100" => n_state <= s58; L<='1'; A<='1';
692             when "01000" => n_state <= s53; L<='1'; A<='1';
693             when "10000" => n_state <= s58; L<='1'; A<='1';
694             when others =>
695                 end case;
696     when s53 =>
697         case X is
698             when "00000" => n_state <= s54; L<='1'; A<='1';
699             when "01000" => n_state <= s53; L<='1'; A<='1';
700             when others =>
701                 end case;
702     when s54 =>
703         case X is
704             when "00000" => n_state <= s54; L<='1'; A<='1';
705             when "00001" => n_state <= s58; L<='1'; A<='1';
706             when "00010" => n_state <= s55; L<='1'; A<='1';
707             when "00100" => n_state <= s58; L<='1'; A<='1';
708             when "01000" => n_state <= s58; L<='1'; A<='1';
709             when "10000" => n_state <= s58; L<='1'; A<='1';
710             when others =>
711                 end case;
712     when s55 =>
713         case X is
714             when "00000" => n_state <= s56; L<='1'; A<='1';
715             when "00010" => n_state <= s55; L<='1'; A<='1';
716             when others =>
717                 end case;
718     when s56 =>
719         case X is
720             when "00000" => n_state <= s56; L<='1'; A<='1';
721             when "00001" => n_state <= s58; L<='1'; A<='1';
722             when "00010" => n_state <= s58; L<='1'; A<='1';
723             when "00100" => n_state <= s58; L<='1'; A<='1';
724             when "01000" => n_state <= s58; L<='1'; A<='1';
725             when "10000" => n_state <= s57; L<='1'; A<='1';
726             when others =>
727                 end case;
728     when s57 =>
729         case X is
730             when "00000" => n_state <= s0; L<='0'; A<='0';
731             when "10000" => n_state <= s57; L<='1'; A<='1';
732             when others =>
733                 end case;
734
735     --Unlatched Alarm Reset State
736     when s58 =>
737         case X is
738             when "00000" => n_state <= s35; L<='1'; A<='1';
```

```
739         when "00001" => n_state <= s58; L<='1'; A<='1';
740         when "00010" => n_state <= s58; L<='1'; A<='1';
741         when "00100" => n_state <= s58; L<='1'; A<='1';
742         when "01000" => n_state <= s58; L<='1'; A<='1';
743         when "10000" => n_state <= s58; L<='1'; A<='1';
744         when others =>
745             end case;
746
747         --Unlatched Unarmed Reset state
748         when s59 => n_state <= s0; L<='0'; A<='0';
749         when others =>
750             end case;--End of H = '0'
751
752         when others =>
753             end case; --End of H cases
754
755     end process;
756
757     Process(p_state)
758     begin
759         if (p_state = s0) then
760             r_state<= '1'; l_state<= '0'; a_state <= '0';
761         elsif(p_state = s12) then
762             r_state<= '0'; l_state<= '1'; a_state <= '0';
763         elsif(p_state = s35) then
764             r_state<= '0'; l_state<= '0'; a_state <= '1';
765         Else
766             r_state<= '0'; l_state<= '0'; a_state <= '0';
767         end if;
768     end process;
769
770
771 end Behavioral;
```