

```

1  -----
2  -- Block code:  alarm_level_display_wo_default.vhd
3  -- History:    30.Sep.2011 - example for introduction to comb logic
4  --             05.Okt.2013 - also check example with default statements!! (dgtm)
5  -- Function:   Decodes the output for a alarm level display.
6  --             Only comb logic. Example of logic with priority.
7  -----
8
9  -- Library & Use Statements
10 LIBRARY ieee;
11 use ieee.std_logic_1164.all;
12
13 -- Entity Declaration
14 ENTITY alarm_level_display_wo_default IS
15     PORT(
16         alarm_prio1      : IN      std_logic;
17         alarm_prio2      : IN      std_logic;
18         alarm_prio3      : IN      std_logic;
19         display_red       : OUT     std_logic;
20         display_orange    : OUT     std_logic;
21         display_yellow    : OUT     std_logic;
22         display_green     : OUT     std_logic
23     );
24 END alarm_level_display_wo_default ;
25
26 -- Architecture Declaration
27 ARCHITECTURE rtl OF alarm_level_display_wo_default IS
28
29 -- Begin Architecture
30 BEGIN
31
32     -----
33     -- Process for combinational logic
34     -----
35     comb_alarm: PROCESS(alarm_prio1,alarm_prio2,alarm_prio3)
36     BEGIN
37         IF (alarm_prio1 = '1') THEN
38             display_red      <= '1';
39             display_orange    <= '0';
40             display_yellow    <= '0';
41             display_green     <= '0';
42
43         ELSIF(alarm_prio2 = '1') THEN
44             display_red      <= '0';
45             display_orange    <= '1';
46             display_yellow    <= '0';
47             display_green     <= '0';
48
49         ELSIF(alarm_prio3 = '1') THEN
50             display_red      <= '0';
51             display_orange    <= '0';
52             display_yellow    <= '1';
53             display_green     <= '0';
54
55         ELSE
56             display_red      <= '0';
57             display_orange    <= '0';
58             display_yellow    <= '0';
59             display_green     <= '1';
60         END IF;
61     END PROCESS comb_alarm;
62
63 END rtl;
64
65

```