```
1   ----------------------------------------------------------------------------------
2   -- Company:
3   -- Engineer:
4   --
5   -- Create Date:    15:11:02 09/21/2023
6   -- Design Name:
7   -- Module Name:    fifo_ent - 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  use IEEE.STD_LOGIC_ARITH.ALL;
23  use IEEE.STD_LOGIC_UNSIGNED.ALL;
24
25  -- Uncomment the following library declaration if using
26  -- arithmetic functions with Signed or Unsigned values
27  --use IEEE.NUMERIC_STD.ALL;
28
29  -- Uncomment the following library declaration if instantiating
30  -- any Xilinx primitives in this code.
31  --library UNISIM;
32  --use UNISIM.VComponents.all;
33
34  entity fifo_ent is
35      Port ( enr : in  STD_LOGIC;
36             enw : in  STD_LOGIC;
37             dataout : out  STD_LOGIC_VECTOR (3 downto 0);
38             datain : in  STD_LOGIC_VECTOR (3 downto 0);
39             empty : out  STD_LOGIC;
40             full : out  STD_LOGIC;
41             clk : in  STD_LOGIC);
42  end fifo_ent;
43
44  architecture Behavioral of fifo_ent is
45
46
47  type memory_type is array(0 to 7) of std_logic_vector(3 downto 0);
48  signal memory : memory_type:=(others => (others => '0'));
49  signal readptr , writeptr : std_logic_vector(3 downto 0):= "0000";
50
51  begin
52        process(clk)
53           begin
54                 if(clk'event and clk = '1' and enw = '1') then
55
56                    memory(conv_integer(writeptr))<= datain;
57                    writeptr <= writeptr + '1';
```

```
58                 end if;
59
60                 if(clk'event and clk ='1' and enr ='1') then
61
62                    dataout <= memory(conv_integer(readptr));
63                    readptr <= readptr + '1';
64                 end if;
65                 if(readptr = "1000") then
66                                empty<='1';
67                                readptr<="0000";
68                 else
69                                empty<='0';
70                 end if;
71                 if(writeptr = "1000") then
72                                full<='1';
73                                writeptr<="0000";
74                 else
75                                full<='0';
76                 end if;
77         end process;
78
79   end Behavioral;
80
81
```