```
-- memory_1
 -- 64x8 RAM block
library IEEE;
use ieee.std_logic_1164.all;
use ieee.std_logic_unsigned.all;
use ieee.std_logic_arith.all;
library synplify;
use synplify.attributes.all;

entity memory_1 is port

(clock              : in std_logic;
--resetn             : in std_logic;
RAM_ADDRESS         : in std_logic_vector(4 downto 0);
RAM_DATA_D          : in std_logic_vector(7 downto 0);
RAM_WRITE_ENABLE    : in std_logic;
RAM_READ_ENABLE     : in std_logic;
RAM_DATA_Q          : out std_logic_vector(7 downto 0));

end memory_1;

architecture rtl of memory_1 is
attribute syn_radhardlevel of rtl : architecture is "tmr";

type MEM is array (0 to 31) of std_logic_vector(7 downto 0);
signal RAM_32x8 : MEM;

begin

 --   begin
    process (clock)
        begin
                if clock'event AND clock = '1' then
                    if RAM_WRITE_ENABLE = '1' then
                        RAM_32x8(conv_integer(RAM_ADDRESS)) <= RAM_DATA_D;
                    end if;
                    if RAM_READ_ENABLE = '1' then
                        RAM_DATA_Q <= RAM_32x8(conv_integer(RAM_ADDRESS));
                   --     RAM_32x8(conv_integer(RAM_ADDRESS)) <= "00000000";
                   -- else
                       -- RAM_DATA_Q <= RAM_DATA_Q;
                    end if;
                end if;

     end process;
 --    RAM_DATA_Q <= RAM_64x8(conv_integer(RAM_ADDRESS));
 end rtl;
```