

```

-- Random access memory

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity ram_array is
  generic (N: integer := 6; M: integer := 32);
  port (clk, we: in STD_LOGIC;
        adr: in STD_LOGIC_VECTOR(N-1 downto 0);
        din: in STD_LOGIC_VECTOR(M-1 downto 0);
        dout: out STD_LOGIC_VECTOR(M-1 downto 0));
end ram_array;

architecture synth of ram_array is
  type mem_array is array ((2**N-1) downto 0)
    of STD_LOGIC_VECTOR(M-1 downto 0);
  signal mem: mem_array;
begin

  process (clk) begin
    if RISING_EDGE(clk) then
      if we = '1' then
        mem (CONV_INTEGER(adr)) <= din;
      end if;
    end if;
  end process;

  dout <= mem(CONV_INTEGER(adr));
end synth;

```