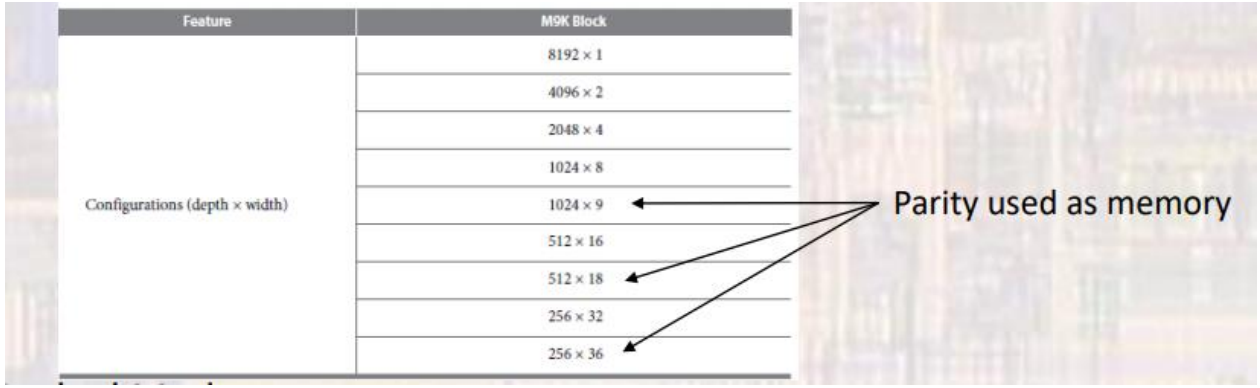
ELE 3510 HW11 Name_____

1A – How many onboard memory blocks would be required to support a 1K word, 16b/word SRAM memory? 10 pts

1KW x 16b/W = 16Kb → 2 M9K blocks

1B – What is the largest size 32bit word memory that can fit in a single on chip memory block? 10 pts

M9K has extra 1K bits attached to the word width, not the # of words → 8Kb → 1KB → 256W in x32 configuration



ELE 3510 HW11 Name_____

2 - Implement a dual clock 8KB single port SRAM in by-32 configuration using the M… …ts

C… …ode showing

P… …owing the nu…

```
----------------------------------------
--
-- sram_8KB_in_by32_mega.vhdl
--
-- created 4/25/17
-- tj
--
-- rev 0
----------------------------------------
--
-- 8KB in x32 dq RAM from mega library
--
----------------------------------------
--
-- Inputs: clk, addr, data_in, we_b
-- Outputs: data_out
--
----------------------------------------
library ieee;
use ieee.std_logic_1164.all;
use ieee.numeric_std.all;

entity sram_8KB_in_by32_mega is

   port(
         i_clk_in:   in std_logic;
         i_clk_out:  in std_logic;
         i_we_b:     in std_logic;
         i_addr:     in std_logic_vector(10 downto 0);
         i_data_in:  in std_logic_vector(31 downto 0);
         o_data_out: out std_logic_vector(31 downto 0)
   );
end;
```

```
architecture behavioral of sram_8KB_in_by32_mega is
--
-- invert we
--
   signal we_sig: std_logic;

   component SRAM_8KB_in_by32
   PORT(
      address     : IN STD_LOGIC_VECTOR (10 DOWNTO 0);
      data        : IN STD_LOGIC_VECTOR (31 DOWNTO 0);
      inclock     : IN STD_LOGIC  := '1';
      outclock    : IN STD_LOGIC ;
      wren        : IN STD_LOGIC ;
      q           : OUT STD_LOGIC_VECTOR (31 DOWNTO 0)
   );
   end component;

   begin
      we_sig <= not i_we_b;

      SRAM_8KB_in_by32_inst : SRAM_8KB_in_by32
         PORT MAP (
            address     => i_addr,
            data        => i_data_in,
            inclock     => i_clk_in,
            outclock    => i_clk_out,
            wren        => we_sig,
            q           => o_data_out
         );

end behavioral;
```