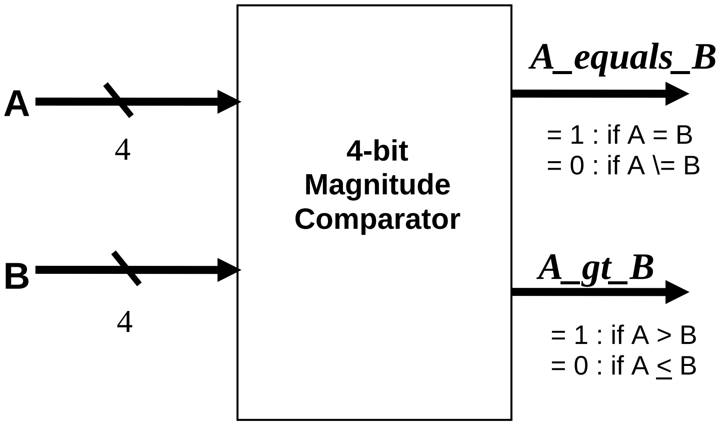
CPE 129 **Homework 15BAssignment** W. Pilkington

1. Use VHDL and Structural Design techniques to create a 4-bit Magnitude Comparator to compare two positive binary numbers (A, B), each having 4 bits (A= $A_3A_2A_1A_0$ and B= $B_3B_2B_1B_0$). The Comparator should output two signals: *A_equals_B* (A=B) and *A_gt_B* (A>B) to describe the relative magnitudes of the two numbers.



Since you have already designed a 2-bit version of this device, try to reuse one of your designs (from Assignment #13D) to create this larger version of the Comparator. You should be able to accomplish this using 2 copies of your previously-designed 2-bit Comparators and some additional logic as shown in the following diagram. (Other approaches may work as well, but try it this way to get practice in Structural Modeling.)

Some suggested steps for your solution:

1) **Create an algorithm for how to determine when the 4-bit numbers are equal, based on comparing the numbers 2-bits at a time using your already-designed comparator (as shown in the following figure).**

   A = B if (A(3 downto 2) = B(3 downto 2)) AND (A(1 downto 0) = B(1 downto 0))

2) **Create an algorithm for how to determine when the 4-bit value for A is greater than the 4-bit value of B, based on the 2-bit comparator results.**

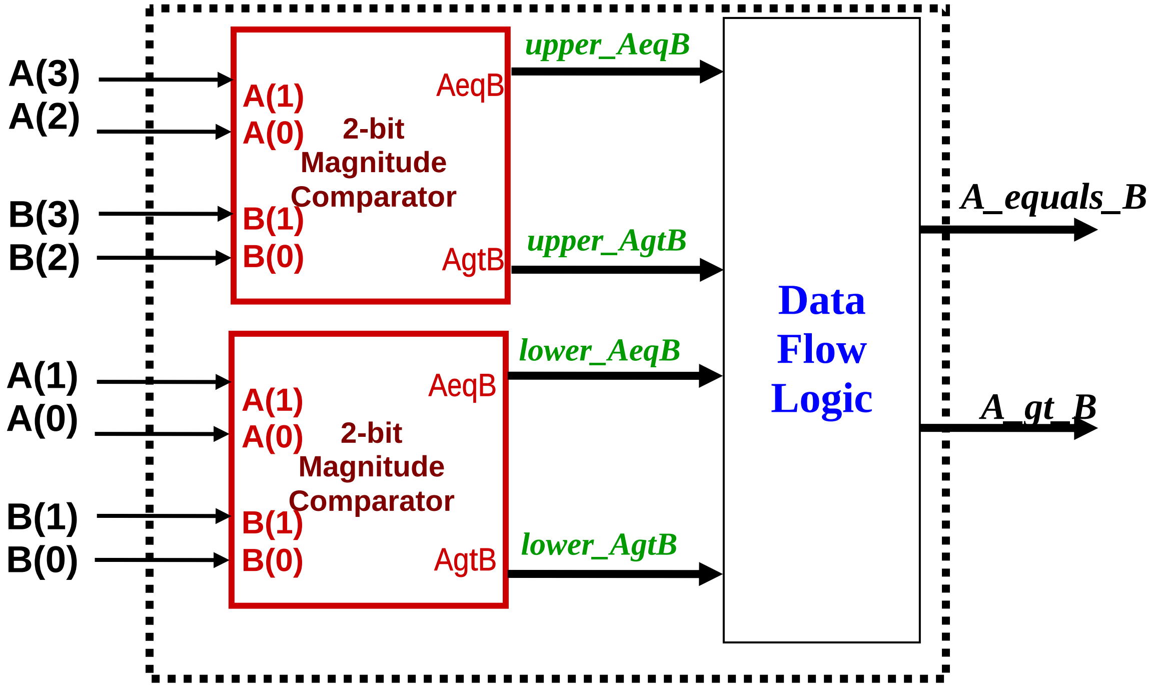
   A > B if (A(3 downto 2) > B(3 downto 2))
   OR
   [(A(3 downto 2) = B(3 downto 2)) AND (A(1 downto 0) > B(1 downto 0))]

3) Choose your favorite version of VHDL description for the 2-bit Comparator design, that you developed in the previous assignment.

```
-- Company: California Polytechnic State University
-- Engineer: Dr. Waynonius Pilkingtonium
-- Create Date:    23:50:34 03/01/2008
-- Design Name:
-- Module Name:    Comparator2Bit - DataFlow
-- Project Name:              Two-Bit Comparator
-- Target Devices: Nexys-2  Spartan 3E-500 FG320-4
-- Tool versions: ISE 9.2i
-- Description: Two-Bit Comparator using DataFlow Architecture
--------------------------------------------------------------------------------
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity Comparator2Bit is
   Port ( A : in  STD_LOGIC_VECTOR(1 downto 0);
            B : in  STD_LOGIC_VECTOR(1 downto 0);
         AeqB : out  STD_LOGIC;          -- True if A = B
         AgtB : out  STD_LOGIC);          -- True if A > B
end Comparator2Bit;

architecture DataFlow of Comparator2Bit is

begin

-- Hopefully, you arrived at these Boolean expressions for the 2-bit comparator:

AeqB <= ( A(1) xnor B(1) )  and  ( A(0) xnor B(0) );

AgtB <= ( A(1) and not B(1) )   or   ( A(0) and not B(1) and not B(0) )
                                          or   ( A(1) and A(0) and not B(0) );

end DataFlow
```

4) Create a VHDL description of the 4-Bit Comparator using the 2-bit Comparator as a “component” in a structural design. Use a Data Flow (concurrent) or a Behavioral Description (process) to implement the additional circuitry needed to generate the 4-bit Comparator outputs from the two sets of 2-bit comparator outputs.

a. Begin by drawing a Block Diagram of your structural design and showing and naming any intermediate signals required.



Basic Block Diagram of a 4-Bit Magnitude Comparator Using 2-Bit Magnitude Comparators

b) Create a VHDL description of the 4-Bit Comparator using the 2-bit Comparator as a “component” in a structural design.

```
--------------------------------------------------------------------------------
-- Company: Calitechnic Unifornia State Polyveristy
-- Engineer: Dr. Waynonius Pilkingtonium
-- Create Date:    23:50:34 03/01/2008
-- Design Name:
-- Module Name:    Comparator4Bit - Structural
-- Project Name:                  4-Bit Comparator
-- Target Devices: Nexys-2  Spartan 3E-500 FG320-4
-- Tool versions: ISE 9.2i
-- Description: 4-Bit Comparator using Structural + DataFlow Arch.
--------------------------------------------------------------------------------
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity Comparator4Bit is
   Port ( A : in  STD_LOGIC_VECTOR (3 downto 0);
          B : in  STD_LOGIC_VECTOR (3 downto 0);
          A_equals_B : out  STD_LOGIC;
          A_gt_B : out  STD_LOGIC);
end Comparator4Bit;
```

```
architecture Structural of Comparator4Bit is

component Comparator2Bit is
        Port ( A : in  STD_LOGIC_VECTOR(1 downto 0);
               B : in  STD_LOGIC_VECTOR(1 downto 0);
             AeqB : out  STD_LOGIC;           -- True if A = B
             AgtB : out  STD_LOGIC);          -- True if A > B
end component;

signal upper_AeqB, upper_AgtB, lower_AeqB, lower_AgtB : std_logic;

begin
-- Instantiate the two 2-bit Comparator modules
upper_comp: Comparator2Bit
                port map (A(3 downto 2), B(3 downto 2), upper_AeqB, upper_AgtB);

lower_comp: Comparator2Bit
                port map (A(1 downto 0), B(1 downto 0), lower_AeqB, lower_AgtB);

-- Logic to use 2-bit comparisons to make 4-bit compare results

A_equals_B <= upper_AeqB and lower_AeqB;

A_gt_B <= upper_AgtB  or (upper_AeqB and lower_AgtB);

end Structural;
```