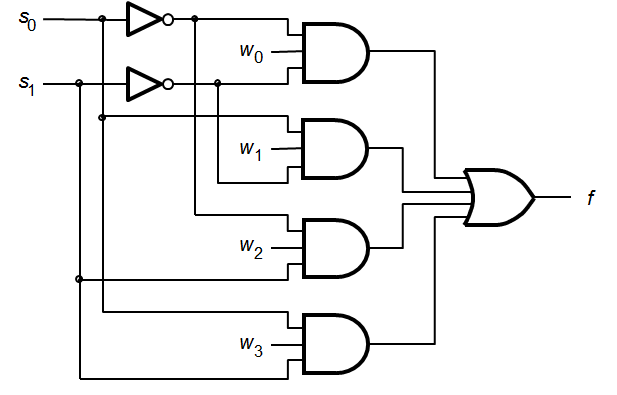
**CS375 LAB 5**

**Multiplexers and VHDL**

Use the Altera Quartus software to implement the following:

1. Create a block diagram for a 4x1 multiplexer, as shown below (from page 319 of your textbook). Create a symbol for this multiplexer.



1. Use five instances of your 4x1 multiplexer to implement a 16x1 multiplexer.
2. Use the 16x1 multiplexer to implement the function ∑m(0,3,6,8,12,13,14).
3. Use Shannon’s Expansion Theorem (section 6.1.2 in your text) to implement the same function you implemented in (3) using a 2x1 multiplexer and some additional logic.
4. Write a VHDL implementation of a HEX-to-7-segment decoder (code converter). Use the included sample VHDL (from page 348 of your textbook) as a guide. Your design should produce the digits (1-9, A, b, c, d, E, and F). Remember that the 7-segment display is active low.

Demonstrate your working designs on the Altera FPGA Starter Board. Use switches Sw3 through Sw0 to drive the HEX-to-7-segment decoder (Sw3 is the most significant bit). Use switches Sw9 to Sw6 to provide input to the multiplexer you implemented in parts 3 and 4(SW9 is MSB). The output from the multiplexers should appear on LED0 and LED1.

Store your design files on the shared google drive in your lab5 folder.

LIBRARY ieee ;

USE ieee.std\_logic\_1164.all ;

ENTITY dec2to4 IS

PORT ( w : IN STD\_LOGIC\_VECTOR(1 DOWNTO 0) ;

En : IN STD\_LOGIC ;

y : OUT STD\_LOGIC\_VECTOR(0 TO 3) ) ;

END dec2to4 ;

ARCHITECTURE Behavior OF dec2to4 IS

SIGNAL Enw : STD\_LOGIC\_VECTOR(2 DOWNTO 0) ;

BEGIN

Enw <= En & w ;

WITH Enw SELECT

y <= "1000" WHEN "100",

"0100" WHEN "101",

"0010" WHEN "110",

"0001" WHEN "111",

"0000" WHEN OTHERS ;

END Behavior ;