VALPARAISO UNIVERSITY
ELECTRICAL AND COMPUTER ENGINEERING DEPARTMENT

ECE 221 **Design Project #2 - BCD to Excess-5 Converter** **FALL 2003**

Introduction: In class we have been talking about combinational logic minimization. It is important to minimize the amount of logic in a digital circuit because it saves both time and money. In this design project you will create two different implementations of the same design, one to minimize the number of gates in your design and one to minimize the number of ICs.

SECTION I. SPECIFICATION

You are to design a combinational circuit that acts as a BCD to Excess-5 Code converter. The converter takes a 4-bit BCD number as input and outputs the corresponding 4-bit Excess-5 Code. The Excess-5 code is simply the number plus five. As an example, the if the BCD number 0 (0000) ere input into the converter, the output Excess-5 code would be 3 (0101). The converter will never have a number greater than 9 as an input so all outputs for those input values can be don't cares.

SECTION II. CONCEPTUAL DESIGN

1. Create a truth table that describes the converter operation. Then use K-maps or Espresso to determine the minimized SOP and POS form for each output (E3 through E0). Draw the circuit (on paper) that implements the form that uses the minimum the number of gates. Use only 2-input AND and OR gates along with inverters.

2. Using the chart below which shows the number of gates per integrated circuit (IC), create a second implementation which uses the LEAST number of ICs (NOT gates) for the converter. You can use any combination of gates given below in your implementation.

7400	Quad 2-input NAND (contains four NAND gates)
7402	Quad 2-input NOR (contains four NOR gates)
7404	Hex Inverter (contains six inverters)
7408	Quad 2-input AND (contains four AND gates)
7432	Quad 2-input OR (contains four OR gates)
7486	Quad 2-input XOR (contains four XOR gates)

SECTION III. MENTOR IMPLEMENTATION

1. Open Design Architect and open a new design named **converter_gates**.

2. Create your minimized GATE implementation and verify it is working correctly. For all of the parts use components in the GEN_LIB. For you input variables use **B3**, **B2**, **B1**, and **B0** and for your output variables use **E3**, **E2**, **E1**, and **E0**.

3. Verify that your design is working correctly using Quicksim. A force file has already been created for you to use and is located at

`/home/ejohnson/Public/ece221/converter.forces`

Run the simulation for 400ns. Once it is correct, print out the waveforms.

4. For the IC minimization implementation open a new sheet and name your design **converter_ic**. Using your conceptual design as a guide, create the design that minimizes the number of ICs used. For you input variables use **B3**, **B2**, **B1**, and **B0** and for your output variables use **E3_IC**, **E2_IC**, **E1_IC**, and **E0_IC**. If you need a +5 volts or ground you can use the **VCC** or **GND** components in the gen_lib.

5. Follow the same procedure in step 3 for your minimized IC implementation. Make sure that the two simulations are equivalent (converter_gates and converter_ic).

SECTION IV. WHAT TO TURN IN

1. The design project header page.
2. Your conceptual design of the BCD to Excess-5 converter including your truth table, k-maps or Espresso input and output files, final POS and SOP expressions along with the minimized gate implementation on paper.
3. A copy of design schematic showing your minimized gate implementation and the output waveforms verifying the implementation is working properly.
4. A copy of design schematic showing your minimized IC implementation and the output waveforms verifying the implementation is working properly.

ECE 221 Digital Logic Design
Design Project #2: BCD to Excess-5 Converter

October 6, 2003

Name: ____

Honor Code Pledge:

____

____

____

Signature:____

Please staple this sheet to the front of your assignment