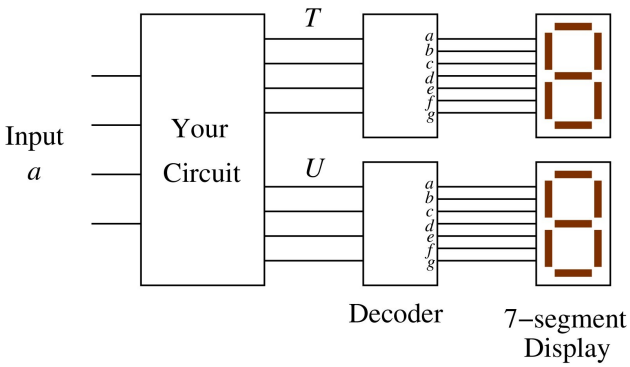
CS29002 SWITCHING LABORATORY
CSE Department, IIT Kharagpur
Spring Semester 2015–16
Module A: Boolean logic and combinational logic circuits
Assignment 2
Date: 25–January–2016

In this assignment, you convert a 4-bit binary number to BCD and display the two decimal digits in 7-segment displays. Follow two methods. The first method uses your indigenous combinational design which you realize using logic gates. The second method makes use of an adder and a comparator (but no gates).

Let a be the input number expressed in 4-bit binary format. This should be converted to two BCD digits T (tens digit) and U (units digit). Each of T and U is represented in the standard 4-bit BCD encoding. If the four bits in one BCD digit are supplied to the inputs of a decoder chip, the chip outputs the desired signals to drive a 7-segment display.



Method 1

Write the truth tables for the eight bits in T and U as functions of the four bits of a. Minimize these functions using Karnaugh maps. Realize the minimized functions using logic gates. Make a 2-level implementation. Choose one from the standard 2-level implementations: AND-OR, OR-AND, NAND, and NOR.

Method 2

Compare a with 9. If $a \leq 9$, then $T = 0$ and $U = a$. If $a > 9$, then you have $T = 1$ and $U = a + 6$ (with the carry ignored). Use a 4-bit comparator to decide which of the two cases occurs. The output of the comparator directly gives T. In the second case, a 4-bit adder is needed to generate U. In order to make the circuit uniform, use the adder always. If $T = 0$, add 0 to a, else add 6 to a.

Notes

1. You can avoid the addition of 0 by using a multiplexer. In this experiment, you do not need to use a MUX.

2. There is a hardware-friendly add3-and-shift algorithm for converting integers from binary to the BCD representation. Our Method 2 is essentially the same as this algorithm for 4-bit inputs.