Electronic Homework 3
Transition Counter

Dr. Tom Conte, Sally Medling – NC State University
Dr. Dave Robinson – University of Delaware
(updated by Sally Medling, Dr. Eric Rotenberg, and Dr. Dave Bradley)

April 9, 2001

1 Introduction

In this lab, you will construct a *transition counter* circuit. It has one input and one output (there is also a clock). A *transition* is defined as either the input changing from 1 to 0, or the input changing from 0 to 1. When four input changes have occured, the circuit outputs a one. The output stays high for one cycle.

Two other conditions must be specified – how does the machine behave initially and how does it restart counting after detecting four input changes.

- An asynchronous reset puts the machine in an initial state. When there have been no prior inputs, the machine should assume the last input value was a 0 (even though there wasn't a previous input). This means an initial input of 1 is counted as the first transition (a change of 0-to-1).
- After four transitions have been counted, counting immediately starts over.

Figure 1 shows two examples, the first has five input changes and the second has twelve input changes. Note that, because this is a Moore machine, there appears to be a 1 cycle delay outputting a 1 because the output depends on state only, and not on the current input. The number of state variables required will depend on the number of states in your machine.

2 Materials

Design your circuit early. You won't know how many and what parts you need until you design the circuit. As before, parts distribution is during a fixed number of days/hours. If you go the last day without a designed circuit, you'll have to guess what parts are needed and later risk having to get additional parts externally.

Here is an approximate guideline for the parts you need.

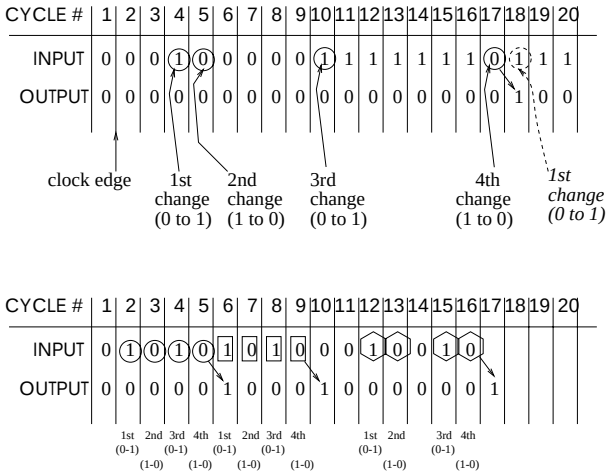

Figure 1: Example input sequences and corresponding timing diagrams.

1. If your design is efficient, you probably already have enough 2-input NAND gates, 3-input NAND gates, inverters, and other gates to implement the combinational logic for your state machine.

2. You'll need D Flip-Flops (D-FF). Each 7474 chip has 2 D-FFs. You shouldn't need more than 2 7474 chips (up to 4 state variables).

3. You'll need to build a clock generator using (1) a pushbutton switch, (2) a 555 timer chip, (3) assorted resistors (1k, 10k), and (4) assorted capacitors (0.01u and 2 10u).

   The pushbutton switch allows you to create a rising clock edge. Mechanical switches are very noisy. When you push the button, the clock edge is not very clean and often appears as multiple clock edges (causing several state transitions when you only want one). The 555 chip is used to create a cleaner "one-shot" (single pulse) signal. The pushbutton is applied to the 555 chip and the output of the 555 chip is the clean clock.

   The clock generator circuit is described in Section 4.

3 Circuit Description

Design a **Moore** machine that implements a transition counter. Your circuit will have one output, which is asserted when four transistions have occured. Examples were given in Figure 1.

The symbolic state table for this circuit has been started for you and can be seen in Table 1.

	input		
state	0	1	output
S0	S0	S1	0
S1	S2	S1	0
S2			
	state*		

Table 1: Symbolic State Table

4 Clock Generator Circuit

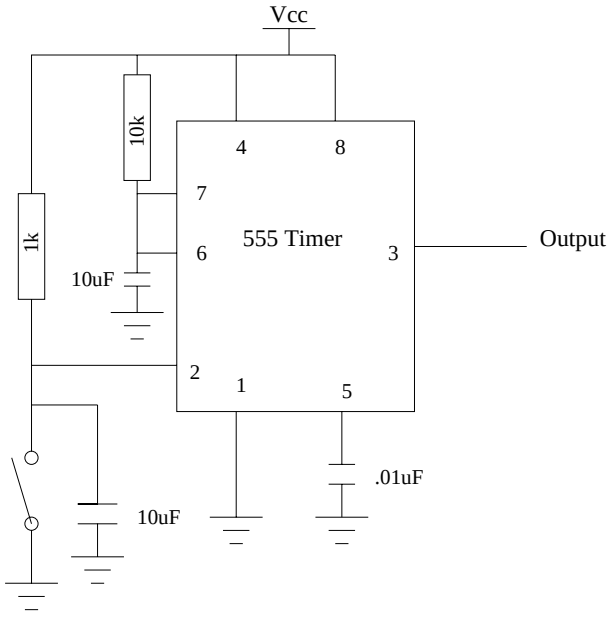

Figure 2: Clock generator circuit. Close the pushbutton switch to create a positive clock edge; the output of the 555 timer is the clock edge supplied to the D Flip-Flops.

5 What to do / Written report contents

The written report is going to be graded very strictly in this e-homework, because you must show your full design to the TAs for them to debug and to grade. Your written report should include the following. Points will be deducted for any missing items.

State Diagram Create a state diagram for the transition counter. Make sure that this is a **Moore** machine.

Symbolic State Table Complete the state table that was started in Table 1.

State Transition Table Do state assignment and then convert the symbolic state table to the state transition table.

K-maps Derive the minimized next-state (Q*) and output equations using K-maps.

Schematic Draw a schematic diagram (logic gates and D-FFs) and implement your state machine on the breadboard. Make sure that you create a fully specified schematic with pin numbers and chip numbers clearly labeled (helps you hook-up the circuit and helps TAs debug).

6 Circuit Help

- The clock input to the D Flip-Flops will come from the clock generator circuit (pushbutton switch + 555 timer).
- You should connect the *Preset* and *Clear* inputs of the D flip-flops to DIP switches. These will be used to set your circuit to state S0 at the beginning of operation. In other words, you must set your circuit to a known state at powerup (hooking up your batteries).
- The TTL datasheet for the 7474 D Flip-Flop is posted on the e-homework web page.