UNIVERSITY OF CALIFORNIA, DAVIS
Department of Electrical and Computer Engineering

EEC180A **DIGITAL SYSTEMS I** **Winter 2006**

LAB 7: SEQUENTIAL CIRCUIT DESIGN

In this lab, you will gain experience in the design and testing of sequential circuits. It is essential that you prepare thoroughly for this lab.

Hardware Required:

2 pcs.	74LS73A	Dual negative-edge-triggered J-K Flip-Flops
as needed	74LS00	Quad 2-input NAND gate
as needed	74LS10	Triple 3-input NAND gate
1 pc.	PALCE22V10	Programmable Logic Device

Preparation (Pre-lab)

Do the *complete* paper design of the state machine specified below implemented using J-K flip flops. Derive a state diagram and a state transition table. Encode the states, draw appropriate K-maps and derive the J-K flip-flop input equations. Determine the output equation based on the state bits.

State Machine Specifications

Design a Moore sequential network which monitors a serial input X and produces an output of Z=1 when a specific sequence, described below, is detected. Otherwise, Z=0. A block diagram of the circuit is shown in Figure 1.

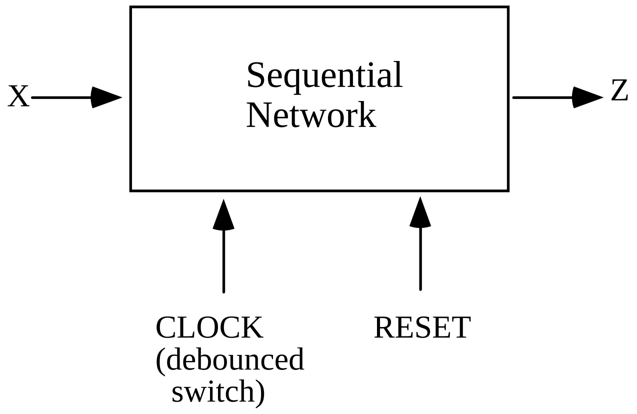

Figure 1: Block diagram

The sequence detector that you will design should meet the following specifications:

a) An output of Z=1 must be produced whenever the last string of four input bits started with a 1 and contained **either two or four** 1's total.

b) After receiving a 1 to start the string, the sequence detector must analyze the next three bits of the string before it looks for the next string. This must occur regardless of whether an output of Z=1 is produced.

An example input / output sequence is shown below, where the sequence proceeds from left to right. The start of a string of four inputs to be analyzed is shown by an up arrow beneath the input. Note that the output will be delayed slightly from the input by propagation delays.

```
INPUT:     0 1 0 1 0 1 1 1 1 1 1 1 0 1 0 0 1 0 0 0 1 1 0 0
             ^       ^       ^       ^             ^
OUTPUT:     0 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0 1
```

Part I: Using Gates and JK Flip-flops

Build your circuit using a minimal number of JK flip-flops, 2- and 3- input NAND gates. Don't forget to build the debounce circuit for the switch input. Demonstrate the circuit to your TA.

Part II: Using a Programmable Logic Device

Your entire sequential network can be implemented in a PALCE22V10. In a previous lab, you used a PALCE22V10 to implement a combinational network. Here, you will use the registered outputs for implementing your state machine. An example file which illustrates a simple state machine design is shown in Figure 2. Notice that your clock input must be on pin 1. Your state bits must be defined as **registered** outputs. As mentioned in lab 5, you should specify the device as PALCE22V10. Follow the example shown in Figure 2 for implementing a global reset input.

The PALCE22V10 has internal D-type flip-flops. Therefore, you will need to use the state transition table and next-state K-maps to derive the D input equations.

Using the PALASM software, enter your design equations for the state machine. Program your PALCE22V10 and verify its operation in circuit. Demonstrate your circuit to the TA for verification.

Lab Report

Each student must submit a lab report. Your lab report should include the minimized state diagram, the state encoding you selected, the state transition table, the K-maps and the final equations. Also include the final schematic of your circuit and your PALASM file.

Grading

Prelab	20 points
Lab Checkoff	60 points
Lab Report	20 points

```
;---------------------------------- PALASM EXAMPLE FILE --
PALASM Design Description

;--------------------------------- Declaration Segment ------------
TITLE    EXAMPLE SEQUENCE DETECTOR
PATTERN
REVISION
AUTHOR   LH
COMPANY  UCD
DATE     12/19/94

CHIP  MOORE  PALCE22V10

; Design specifications: Moore sequential network which investigates
; an input sequence X and produces an output of Z=1 for any input
; sequence ending in 0110 or 101.

;--------------------------------- PIN Declarations ---------------
PIN  1          CLOCK                              ; INPUT
PIN  2          RESET                              ; INPUT
PIN  3          X                                  ; INPUT
PIN  12         GND                                ; INPUT
PIN  20         Q0              REGISTERED         ; OUTPUT
PIN  21         Q1              REGISTERED         ; OUTPUT
PIN  22         Q2              REGISTERED         ; OUTPUT
PIN  23         Z               COMBINATORIAL      ; OUTPUT
PIN  24         VCC                                ; INPUT
NODE 1          GLOBAL                             ; INPUT

;---------------------------------- Boolean Equation Segment ------
EQUATIONS

GLOBAL.RSTF = RESET   ; When reset input is high, registered outputs reset

Q0 = Q2*/X + Q2*Q1*/Q0 + /Q2*Q0*X + /Q1*/X

Q1 = /Q1*X + Q2*Q0*X + /Q2*/Q0*X + Q2*/Q1*Q0

Q2 = Q2*/Q0*X + Q1*Q0*X + /Q2*Q1*/X + Q2*/Q1*Q0*/X

Z = Q2*Q1

;---------------------------------- Simulation Segment ------------
SIMULATION

trace_on CLOCK X Z Q0 Q1 Q2 RESET
setf RESET X
clockf CLOCK
setf /RESET
clockf CLOCK
```

```
setf /X
clockf CLOCK
setf X
clockf CLOCK
check Z
clockf CLOCK
setf /X
clockf CLOCK
check Z
setf X
clockf CLOCK
check Z
trace_off

;-------------------------------------------------------------------
```

Figure 2: Example PALASM input file