ECE 270 Lab Verification / Evaluation Form

Experiment 8

Evaluation:

IMPORTANT! You must complete this experiment during your scheduled lab period. All work for this experiment must be demonstrated to and verified by your lab instructor *before the end* of your scheduled lab period.

STEP	DESCRIPTION	MAX	SCORE
Pre-lab 1	Sketch schematic of logic realized by the Verilog module	2	
Pre-lab 2	Build circuit on breadboard	2	
Step 1	Verify functionality of circuit and Verilog Description	2	
Step 2	Test SR latch input combinations	2	
Step 3	Create and test a D latch	2	
Step 4	Create and test a negative edge-triggered D flip-flop	2	
Step 5	Create and test a positive edge-triggered D flip-flop	2	
Step 6	Complete 74HC74 function table	2	
Step 7	Create a two-bit binary counter	2	
Step 8	Print waveforms generated by counter	2	
Step 9	Measure 74HC74 propagation delays	2	
Step 10	Thought questions	3	
	TOTAL	25	

Signature of Evaluator: _____

Academic Honesty Statement:

"In signing this statement, I hereby certify that the work on this experiment is my own and that I have not copied the work of any other student (past or present) while completing this experiment. I understand that if I fail to honor this agreement, I will receive a score of ZERO for this experiment and be subject to possible disciplinary action."

Last Name (Printed): _____ **Lab Div:** _____ **Date:** _____

E-mail: __ __ __ __ __ __ __ __ **@**purdue.edu **Signature:** _____

Introduction to Sequential Circuits

Instructional Objectives:

- To learn how to build and test simple feedback sequential circuits
- To learn the basic functionality of an edge-triggered D flip-flop
- To learn how to measure the propagation delay of an edge-triggered D flip-flop

Pre-lab Preparation:

- Read this document in its entirety
- Review the material presented in Module 3

Lecture/Demonstration:

Your lab instructor will give a brief presentation that includes the following:

- A review of the difference between a latch and a flip-flop
- A review of how to measure propagation delays

Experiment Description:

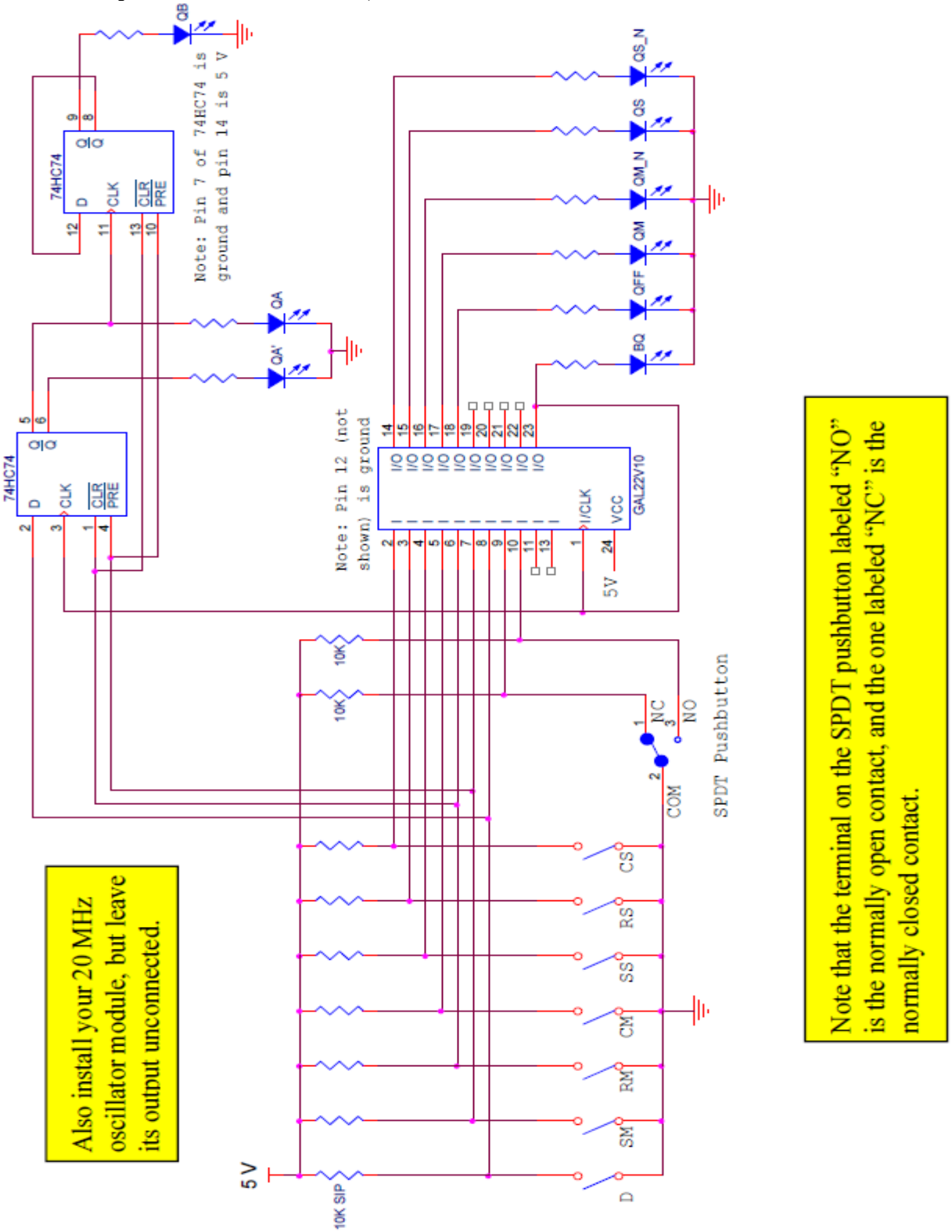
In this experiment, you will implement and test some simple sequential circuits that have been discussed in lecture. You will verify the operation of a typical edge-triggered D flip-flop and measure its timing parameters.

Pre-lab Step (1):

Examine the `Verilog Module` provided on the course website for this experiment and sketch a schematic of the logic that it realizes. HINT: This program realizes a bounceless switch, two SR latches with enables, and a D flip-flip (corresponding to `always @(posedge ---)` block).

Pre-lab Step (2):

Build the following circuit using your DIP switch, SPDT pushbutton, ATF22V10C, 74HC74, and RED (resistor) LEDs, leaving ample room around the PLD to easily remove it from the breadboard for reprogramming. Also install your 20 MHz oscillator module and connect it to power and ground (but leave its output unconnected for now).



Step (1):

Synthesize the Verilog Module provided on the course web site using ispLEVER™, targeting it for a **GAL22V10C** device, and use the Universal Programmer at your lab station to "burn" the JEDEC file produced into your ATF22V10C PLD. Test the bounceless switch, making sure the `BQ` LED illuminates when the pushbutton is pressed. Verify that `QFF` operates as a positive edge-triggered D flip-flop: `QFF` should "load" the value input on `D` when the pushbutton is pressed, and remain that value when the pushbutton is released. Also verify that the 74HC74 output labeled `QA` mirrors that of the PLD's `QFF` output (make sure the **active low** `CLR'` and `PRE'` inputs of the 74HC74 are ***negated***, i.e., ***high***).

Step (2):

Test all the possible input combinations for the two SR latches ("`QM`" and "`QS`") and verify that they function as expected. In particular, note the *next state* each SR latch goes to when the S R **C** input combination is changed from "1 1 **1**" to "1 1 **0**". Verify that `QM` and `QM_N` (as well as `QS` and `QS_N`) are *not always* complements of each other – note the input combinations that cause both outputs to be the same value, and be prepared to explain what is happening.

Checkpoint: Demonstrate circuit functionality and SR latch behavior to Lab Instructor.

Step (3):

Comment out the wire and port declarations for `CM`, `RM`, and `SM` (`loc 5,6,7`) and assign `CM=CLKIN`, `SM=D`, and `RM=!D`. Reprogram your PLD with the modified Verilog description and verify that you have created a D latch (the output of which is `QM`) that "opens" when the pushbutton is pressed (i.e., assumes the value input on `D` and "follows" it as long as the latch is open), and "closes" (retains its value) when the pushbutton is released. Also verify that `QM` and `QM_N` are always complements of each other. (`QM` is referred to as the "master" or "input" latch.)

Step (4):

Comment out the `wire and port` declarations for `CS`, `RS`, and `SS` (`loc 2,3,4`) and assign `CS=!CLKIN`, `SS=QM`, and `RS=QM_N`. Reprogram your PLD with the modified Verilog description and verify that you have created a *negative edge-triggered* flip-flop (the output of which is `QS`) that "reads" its `D` input when the pushbutton is pressed and changes state when the pushbutton is released. Also verify that `QS` and `QS_N` are always complements of each other. (`QS` is referred to as the "slave" or "output" latch.)

Checkpoint: Demonstrate negative edge-triggered flip-flop to Lab Instructor.

Step (5):

Swap the polarity of the master and slave latch enables, i.e., assign `CM=!CLKIN` and `CS=CLKIN`. Reprogram your PLD with the modified program and verify that you have created a *positive edge triggered* flip-flop that "reads" the `D` input when the pushbutton is released and changes state when the pushbutton is pressed. Also verify that `QS` and `QS_N` are always complements of each other and that the behavior of `QS` mirrors that of outputs `QFF` and `QA`.

Checkpoint: Demonstrate positive edge-triggered flip-flop to Lab Instructor.

Step (6):

Referring to the 74HC74 data sheet provided on the course web site (under References), verify that the D flip-flop designated QA (Q output on pin 5) operates according to the published function table. Be sure to try clocking the flip flop (using the pushbutton) when either the PRE′ (preset) or CLR′ (clear) input is **asserted low.** Record your observations in the table below (refer to the schematic to determine the DIP switches used to provide the PRE′ and CLR′ signals).

Inputs				*Outputs*	
PRE′ (pin 4)	CLR′ (pin 1)	CLK (pin 3)	D (pin 2)	Q (pin 5)	Q′ (pin 6)
L	H	X	X		
H	L	X	X		
L	L	X	X		
H	H	↑	H		
H	H	↑	L		
H	H	L	X		
H	H	H	X		
H	H	↓	X		

Step (7):

Disconnect pin 2 of the 74HC74 from the DIP switch and reconnect it to pin 6 of the 74HC74 (i.e., connect the QA flip-flop's Q′ output to its D input). Assert the CLR′ inputs of the two D flip flops (with the PRE′ inputs negated) and verify that both QA and QB clear to 0. Assert the PRE′ inputs of the two D flip flops (with the CLR′ inputs negated) and verify that both QA and QB set to 1. Next, negate both the PRE′ and CLR′ inputs and clock the circuit multiple times (by pressing the pushbutton); observe the sequence of values generated on QA and QB. Also, note what happens if either the PRE′ or CLR′ inputs are asserted while the circuit is being clocked. Be prepared to describe what the circuit is doing and explain why.

Checkpoint: Verify table entries and demonstrate QA and QB behavior to Lab Instructor.

Step (8):

Disconnect pin 3 of the 74HC74 from the BQ output of the ATF22V10C and reconnect it to the output of the 20 MHz oscillator module (i.e., connect the QA flip-flop's CLK input to the output of the 20 MHz oscillator). Disconnect the QA and QB LEDs (from pins 5 and 9 of the 74HC74, respectively) and connect Channel 1 of the oscilloscope to pin 5 of the 74HC74 and Channel 2 to pin 9. Observe the output waveforms produced on the oscilloscope and determine the frequency relationship between them. Print the waveforms produced and have them available for your lab instructor to check.

Step (9):

Connect Channel 1 of the oscilloscope to pin 11 of the 74HC74 and Channel 2 to pin 9. Measure the $t_{PLH(C\rightarrow Q)}$ and $t_{PHL(C\rightarrow Q)}$ of the flip-flop. Next, disconnect Channel 2 of the oscilloscope from pin 9 of the 74HC74 and reconnect it to pin 8. Measure the $t_{PLH(C\rightarrow QN)}$ and $t_{PHL(C\rightarrow QN)}$ of the flip-flop. Record your results below.

$t_{PLH(C\rightarrow Q)}$ = _____ $t_{PHL(C\rightarrow Q)}$ = _____ $t_{PLH(C\rightarrow QN)}$ = _____ $t_{PHL(C\rightarrow QN)}$ = _____

Checkpoint: Show waveform printouts and timing measurement results to Lab Instructor.

Step (10):

Place your answers to the following "thought questions" in the space provided below:

(a) In your own words, explain the difference in behavior between a latch and a flip-flop.

(b) With respect to the observed behavior of the `Q` and `Q_N` outputs, what differences (if any) are there between the S and R inputs of an SR latch and the `PRE` and `CLR` inputs of the 74HC74?

(c) How do the C→Q rise and fall propagation delays you measured for the 74HC74 compare with its *published* specifications? (Refer to the data sheets provided on the course web site under References.) Is it possible to build a latch or flip-flop with a "balanced" rise and fall propagation delay? Why or why not?

