

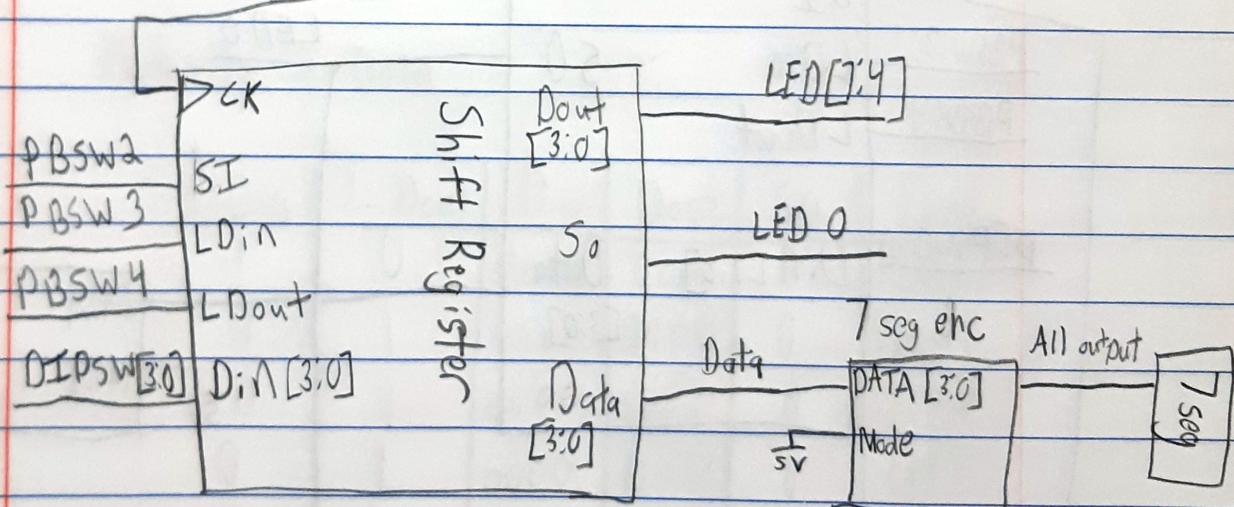
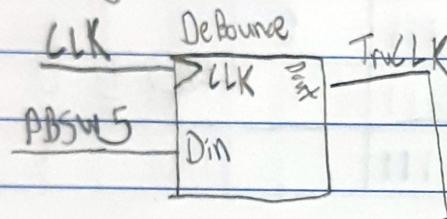
# Lab 15 Shift Register

## Overview:

The purpose of this lab is to implement a shift register onto an FPGA. Eventually, this will be made into a serial data transfer block, similar to USB.

## Activity 1 - Implement Register

### Diagram:



### Observations:

Remember to invert buttons and LEDs

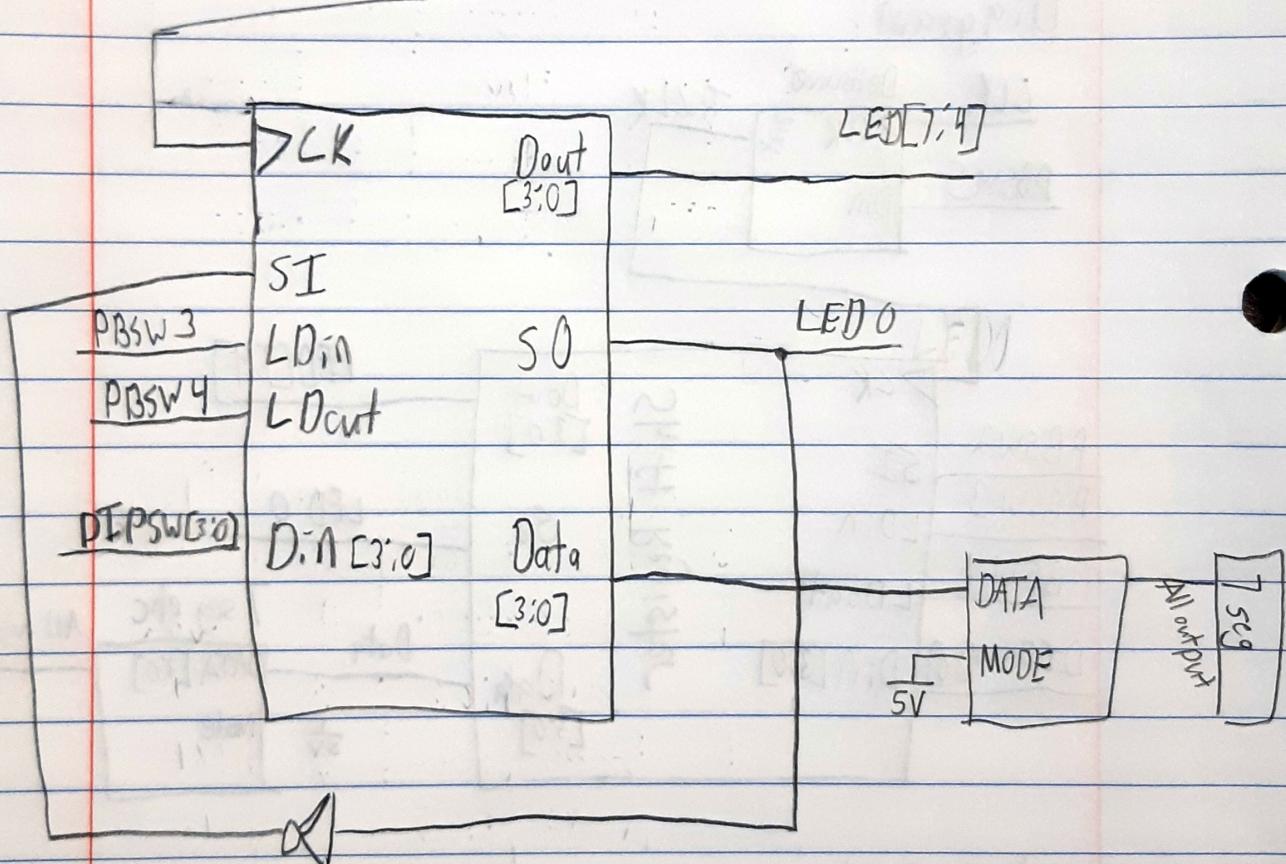
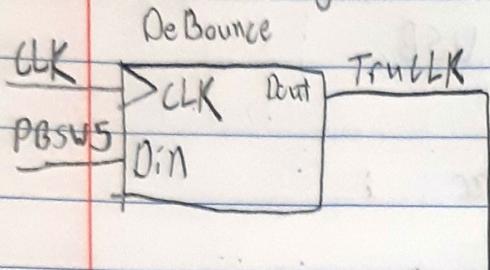
DOUT SHOWS DATA BEFORE SHIFT

Works as expected!

# Lab 15 Shift Register

## Activity 2 - Johnson Counter

### Block Diagram



Observations:

Implementation was easy and painless, no issues

# Lab 15 Shift Register

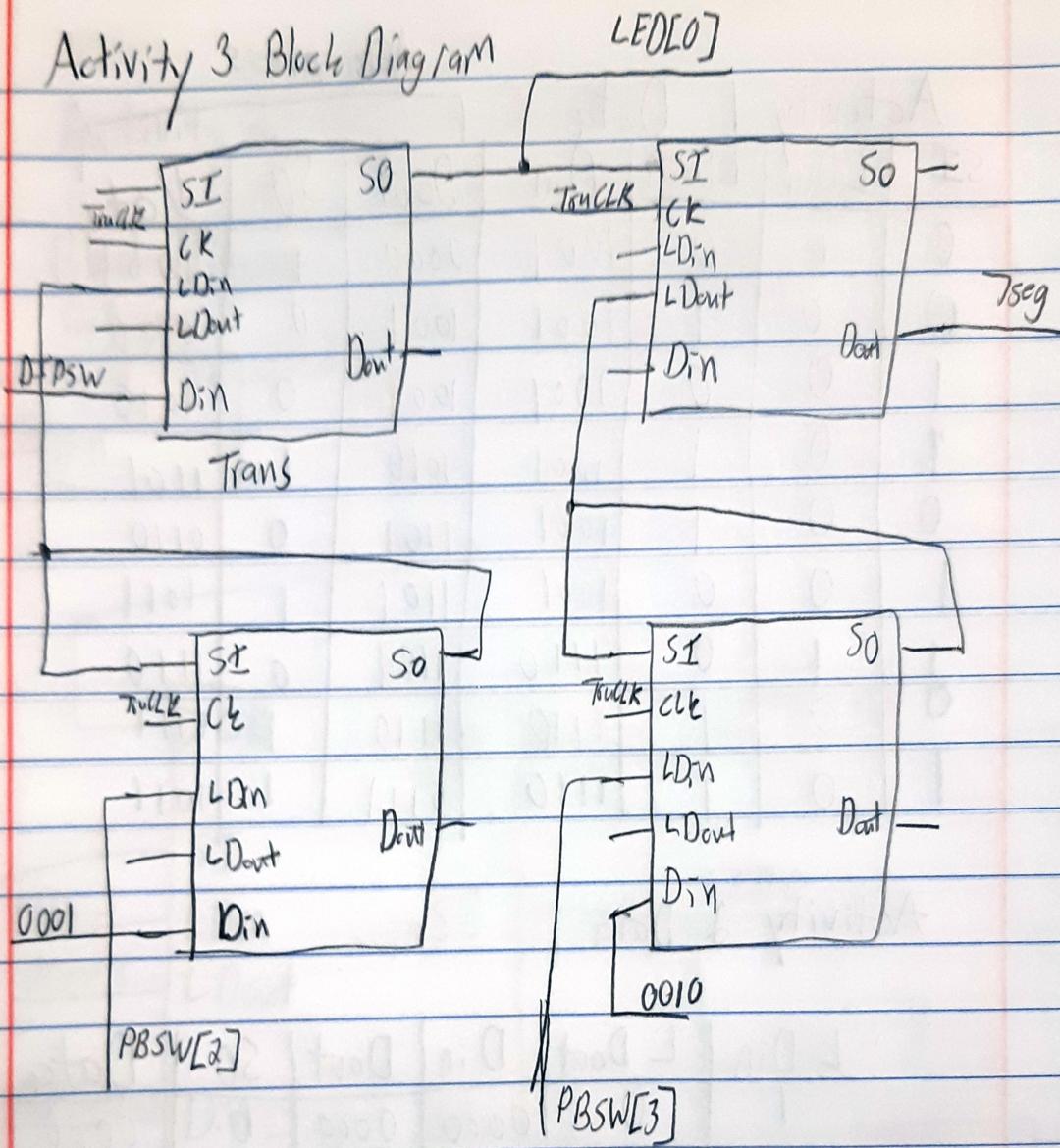
SI	Activity		Data		After press		Data <sub>q</sub>
	L D <sub>in</sub>	L D <sub>out</sub>	D <sub>in</sub>	D <sub>out</sub>	S0		
0	1	0	1001	0000	1	1001	
0	0	1	1001	1001	0	0100	
1	0	0	1001	1001	0	1010	
1	0	1	1001	1010	1	1101	
0	0	1	1001	1101	0	0110	
1	0	0	1001	1101	1	1011	
1	1	0	1110	1101	0	1110	
0	0	1	1110	1110	1	0111	
1	0	1	1110	0111	1	1011	

Activity & Data

L D <sub>in</sub>	L D <sub>out</sub>	D <sub>in</sub>	D <sub>out</sub>	S0	Data <sub>q</sub>
1	0	0000	0000	0	0000
0	1	0000	0000	0	1000
0	1	0000	1000	0	1100
0	1	0000	1100	0	1110
0	1	0000	1110	1	1111
0	1	0000	1111	1	0111
0	1	0000	0111	1	0011
1	0	1010	0111	0	1010
0	1	1010	1010	1	0101
0	1	1010	0101	0	1010
0	1	1010	1010	1	0101
0	1	1010	0101	0	1010

# Lab 15 Shift Register

## Activity 3 Block Diagram



Observations:

INVERT BUTTONS; not doing so constantly loads  
Din

Make sure that all clocks are equal

# Lab 15 Shift Register

Data:

DIPSW	Serial Pattern	Parallel Hex	Expected?
1100	1100	C	✓
0101	0101	5	✓
0000	0000	0	✓
1111	1111	F	✓

When no new info is fed in before next cycle,  
our parallel output remains constant as our  
serial output repeats the same pattern over and over

Conclusion: This lab represented the USB connection  
layout very well. It shows how timing is important  
for these types of connections as a mistiming  
results in incorrect data.