# Lab 6 - Sequential Logic Circuits (II)

CDA 3201L-003
Jason Keene and Jacob Manfre
Submitted Mar 22, 2015

## Purpose and Objectives

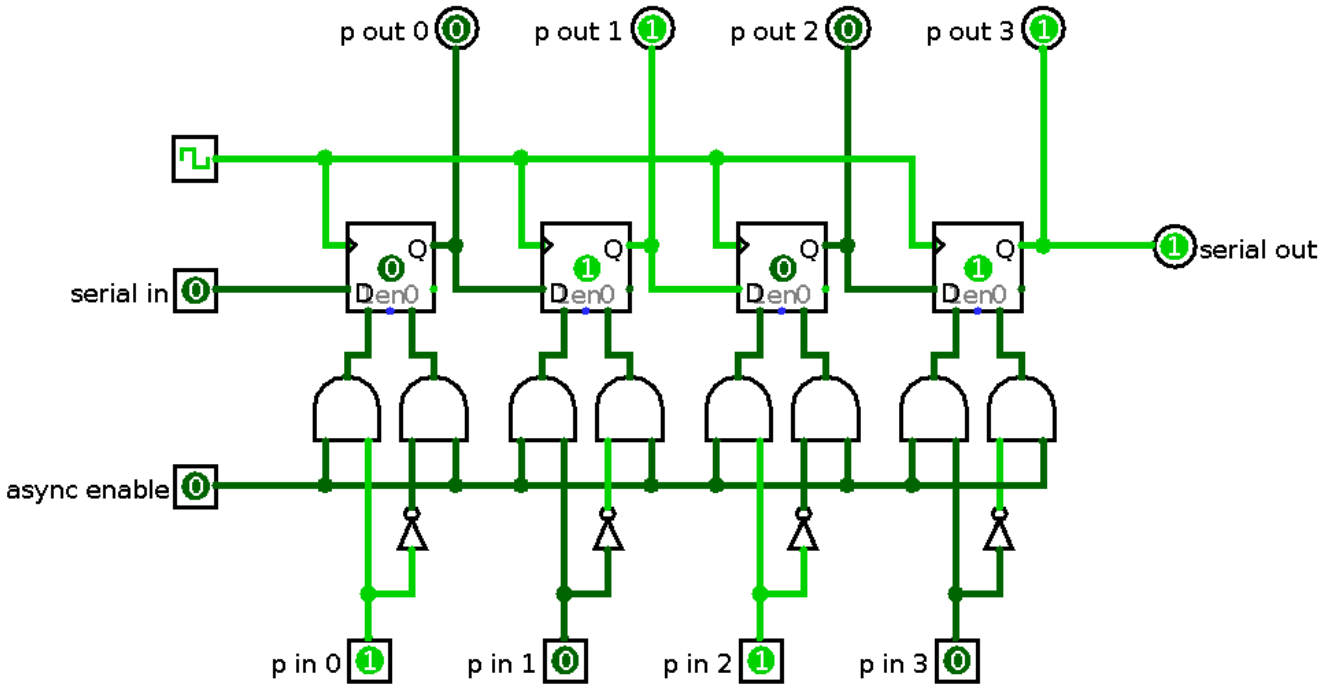
In this lab we set out to implement a 4 bit left shirt register with the knowledge of D and JK flip flops. We attempt to demonstrate left shifting using asynchronous parallel load but also understand how this can be done using synchronous parallel load. In Part B we attempt to demonstrate how to both left shift and right shift 4 bits using a single shift register.

## Component List

- Breadboard
- Wiring
- 5v power supply
- TTL 0.5 Htz Function Generator
- 2 x Dual Positive Edge Triggerd D Flip-Flops (74LS74)
- 1 x Shift Register (74LS194)
- 1 x Hex Inverter (74LS04)
- 2 x Quad 2 input NAND Gate (74LS__)
- 9 x LED
- 8 x Resistor (470 Ohms 5%)

## Design

In our design we ANDed the asynchronous imput with each of the 4 bit inputs and their negations and used each bit and its negation as our presets for our D-flip flops. Our serial input was used for the most significant bit of the D-flip flops while the Q outputs were subsequently used for the following D-flip flops from most significant bit to least. Each Q output for the D-flip flops were used for our output.




## Test Vectors and Verification

We made a truth table for all possible inputs for our left shift register and our outputs demonstrated on the breadboard matched the outputs generated through the truth table.

| W | X | Y | Z | Q(1) | Q(2) | Q(3) | Q(4) |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 |
| 1 | 0 | 1 | 0 | 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 1 | 0 | 1 | 1 | 1 |
| 1 | 1 | 0 | 0 | 1 | 0 | 0 | 1 |
| 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 |
| 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

## Discussion and Conclusion

We learned in this lab how to use asynchronous inputs to enable the input of 4 bits through D-flip flops to create a left bit register. This has a practical application in computing by creating a way to multiply binary numbers through left shifting. For every multiple of 2 being multiplied to a bit, the

output bits will shift by 1 to the left.

Part B helped us understand how we might then take advantage of this understanding to implement right shifting through a shift register to demonstrate the capacity to also represent division in binary algebra. For each multiple of 2 being divided, the output bits will shift to the right by 1. Both left and right bit shifting are very powerful tools that add to the arsenal of basic operations that can be preformed using binary logic.

## Questions and Answers

1. How would you modify this design to provide synchronous parallel load instead of asynchronous parallel load?

The clock signal along with an edge detector would need to be added to all the AND gates so that only when the clock edge and enable were high would it do a parallel load.

1. How would you modify this design to allow a synchronous right-shift along with synchronous parallel load?

For synchronous right shift the serial in would need to be wired to the D input of the right most flip flop and then it's Q output wired to the second to the right flip flop, etc.