Muğla Sıtkı Koçman University

EEE-2002 – Digital Systems

DESIGN PROJECT

Ahmad Zameer Nazarı

220702706

3rd Friday 2024

111 Sequence Detector from a Serial Shift Register

Using JK Flip-Flops

Table of Contents

Introduction

My design project consists of two clocked sequential circuits. The first is a serial in serial out (SISO) shift register. Which serves as a medium to transfer serial data to the 2nd circuit. this latter portion of the circuit then examines this serial data fed to it and detects a certain string of bits. The one I chose is three consecutive 1s, i.e. 111. It will include flip flops, of course it being a sequential circuit.

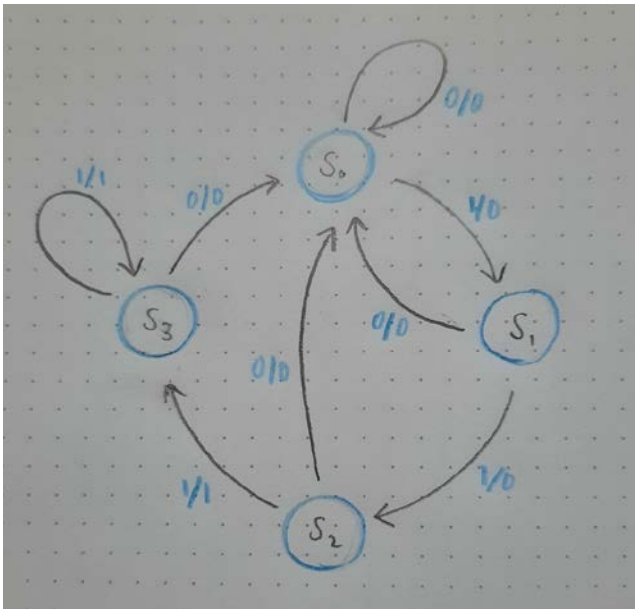
Sequence detectors or recognizers are quite important, as they are essential for tasks such as error detection, data compression etc.

Design

State Diagram

I will base my design on Mealy machine type of Finite State Machines (FSM), where the output will depend on current state and input.

It can be readily observed that the circuit has 4 states, for initially, it awaits the first 1. then when it is received it goes to the next state S_1 from S_0, Then it awaits the 2nd 1. Which when received, enables it to go to the next higher state S_2. Receiving the 3rd 1, will push it the final state S_3.



Now an important thing to consider is overlapping inputs. What should be the output when the 4th or similarly higher consecutive 1s are fed. If we assume overlapping inputs,

then next higher 1s will still be included as part of state S_3. While for nonoverlapping case, the next higher 1 will not be accounted and the system is to be reset to S_0.

We will assume the case of overlapping inputs.

State Table

With the states determined we can now proceed to state assignment. These 4 states can be sufficiently described by 2 bits. So we let $S_0 = 00, S_1 = 01, S_2 = 10$ and $S_3 = 11$.

Which also determines the number of flip flops required as two.

States	Present State		Input	Next State		Output	Flip Flop Inputs			
	Q_1	Q_0	x	Q_1^+	Q_0^+	y	J_1	K_1	J_0	K_0
S_0	0	0	0	0	0	0	0	X	0	X
	0	0	1	0	1	0	0	X	1	X
S_1	0	1	0	0	0	0	0	X	X	1
	0	1	1	1	0	0	1	X	X	1
S_2	1	0	0	0	0	0	X	1	0	X
	1	0	1	1	1	1	X	0	1	X
S_3	1	1	0	0	0	0	X	1	X	1
	1	1	1	1	1	1	X	0	X	0

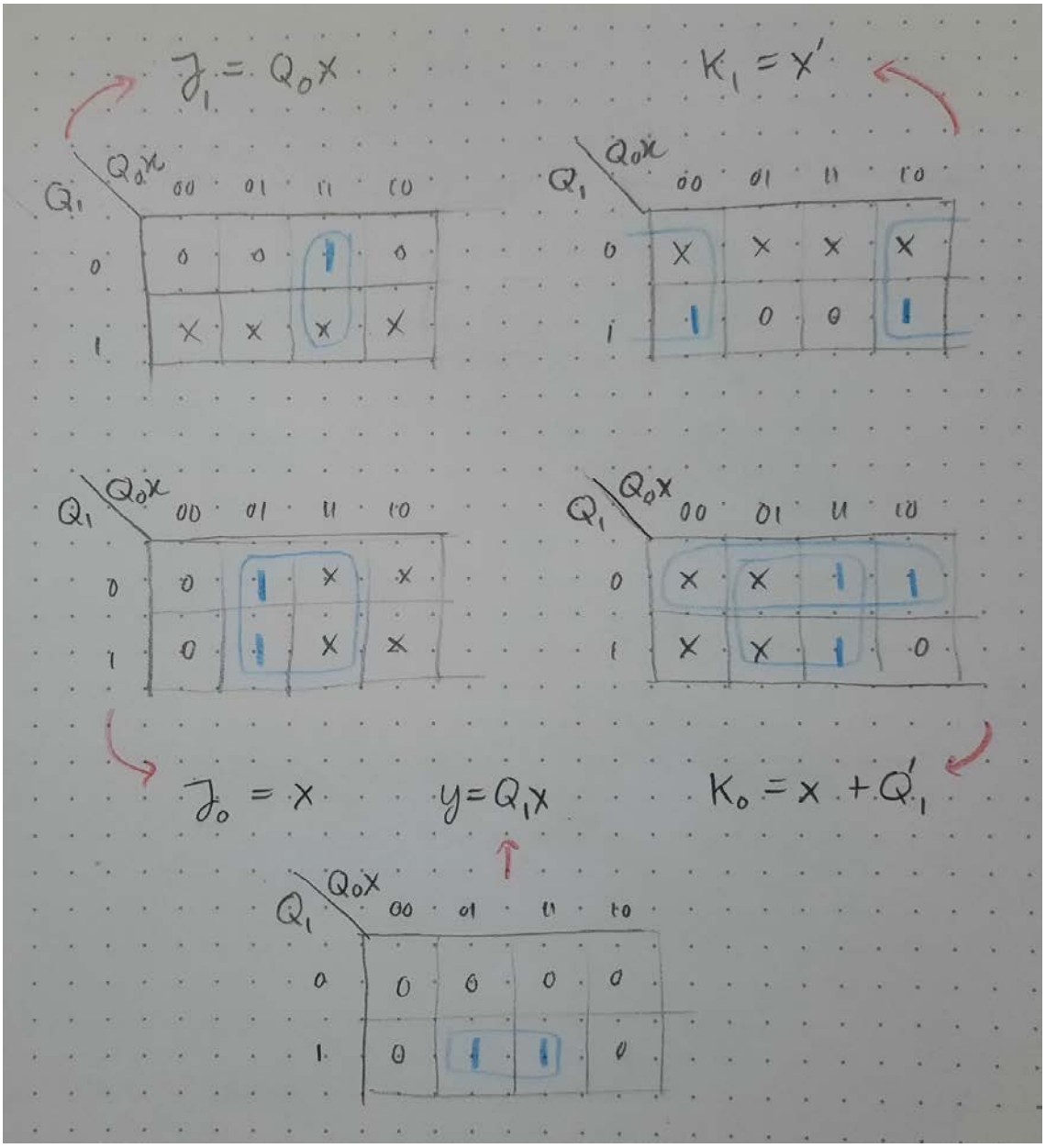
Where the flip flop inputs have been determined from the JK excitation table:

$Q(t)$	$Q(t+1)$	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

JK Flip flop truth/characteristic table is:

J	K	$Q(t+1)$
0	0	$Q(t)$
0	1	0
1	0	1
1	1	$Q'(t)$

Karnaugh Maps



Thus, we get:

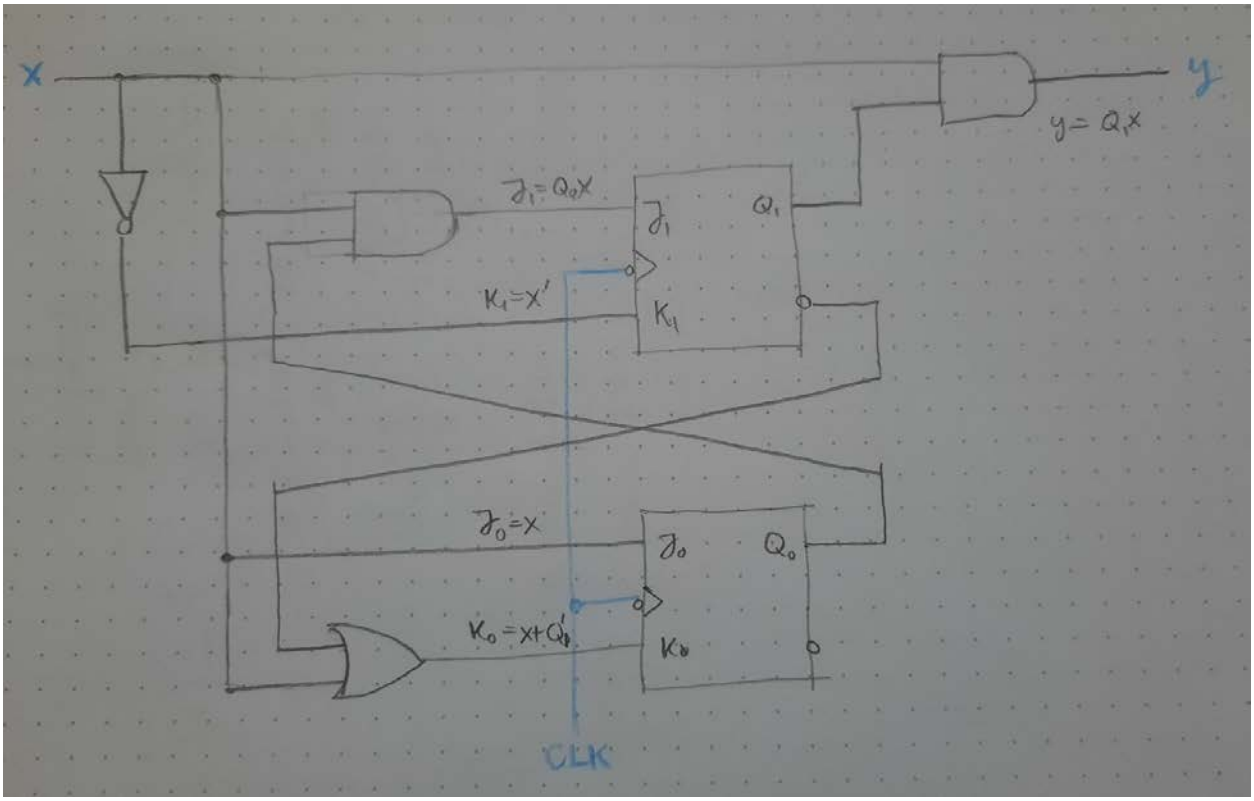
$$J_1 = Q_0 x \quad , \quad K_1 = x'$$

$$J_0 = x \quad , \quad K_0 = Q_1' + x$$

While output: $$y = Q_1 x$$

Resulting Logic Diagram

The resulting circuit is one given below where the flip flops constitute the clocked sequential portion, and the basic gates the combinational circuit portion.

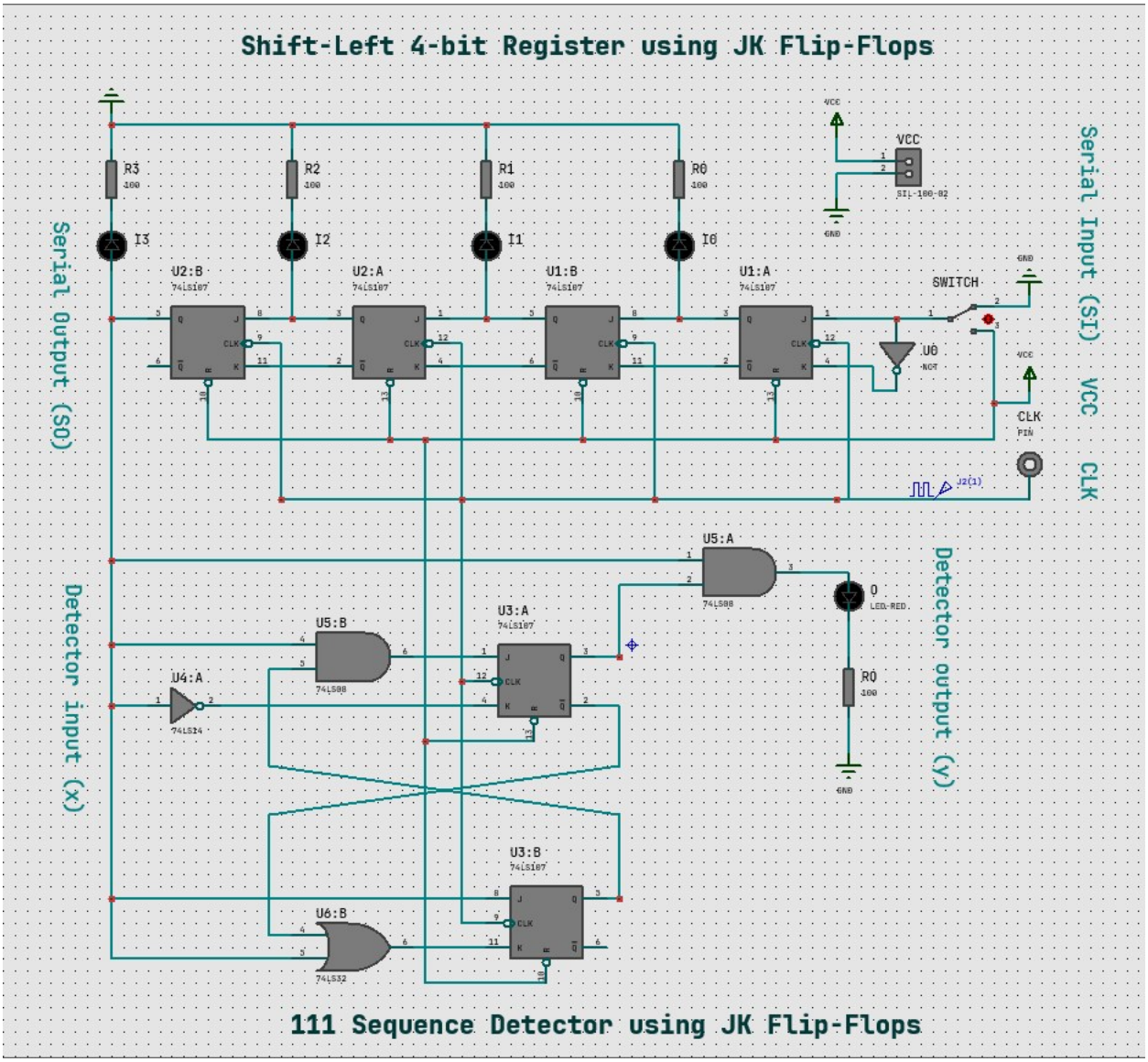


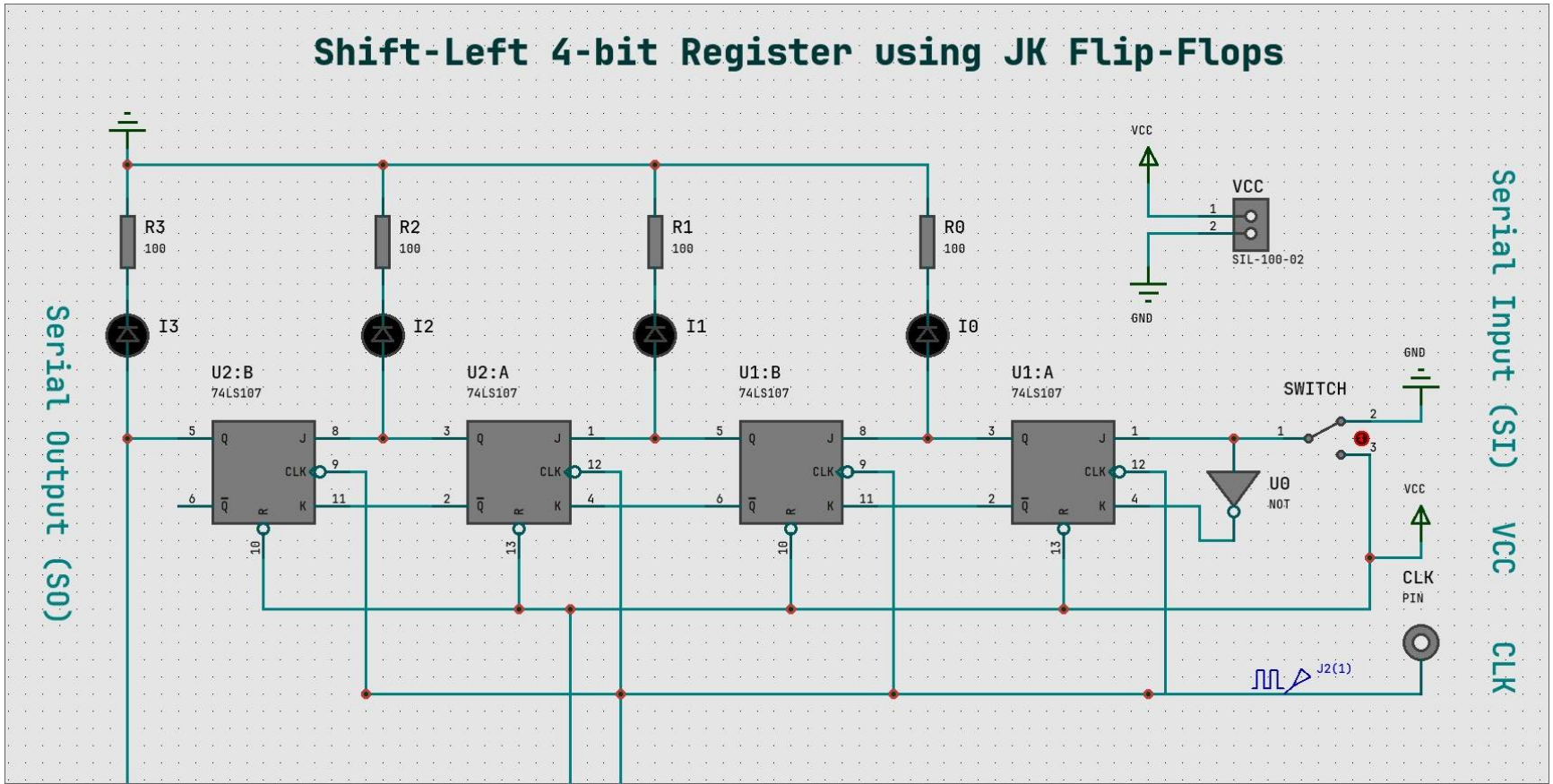
Serial Shift Register

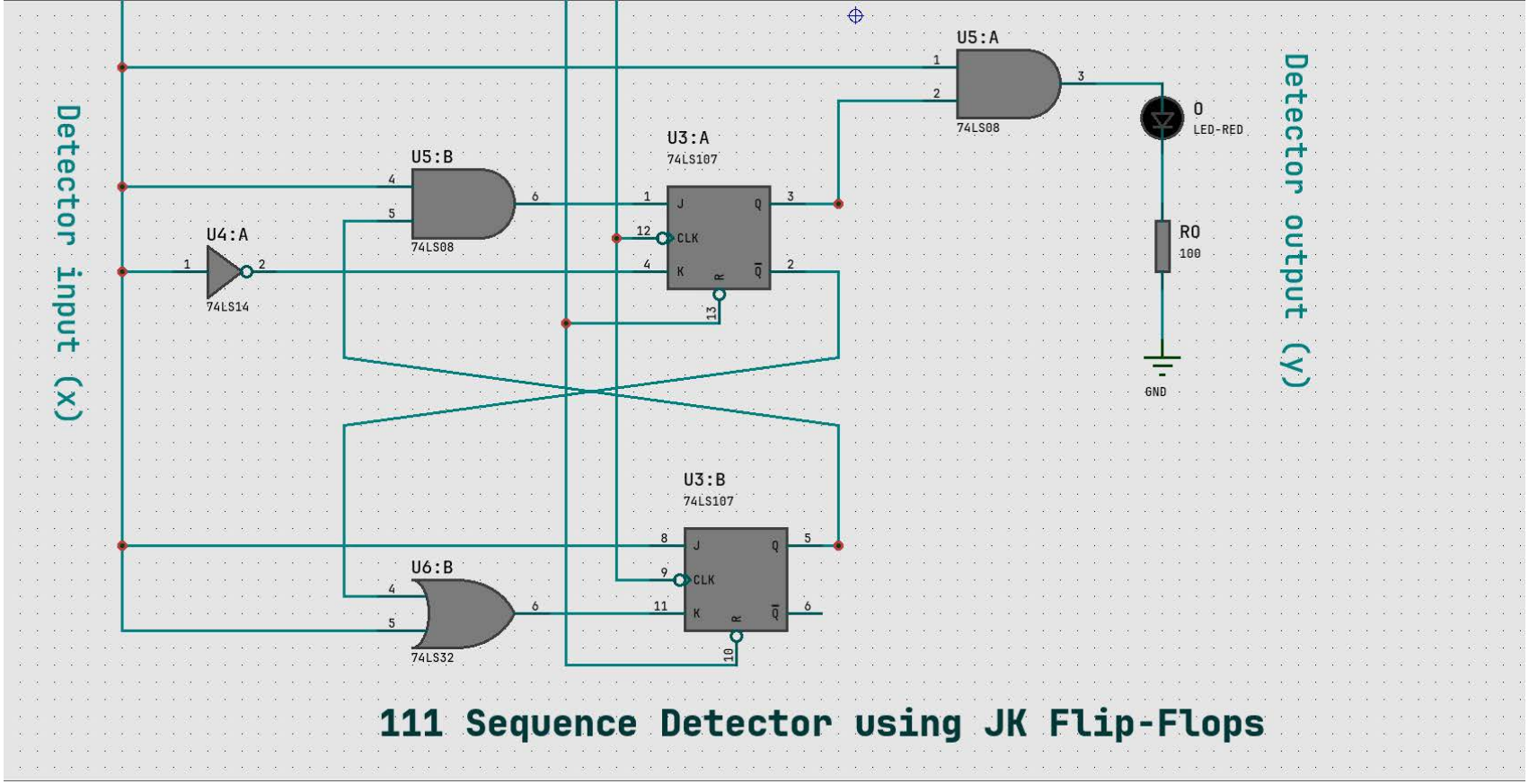
I will use JK Flip flops to implement a 4-bit shift register. 4 JK flip flops will be required. The outputs of each flip flop is connected to the J input of the subsequent flip flop and the complement output to the corresponding K input. Then if J and K of the first flip flop is given a complemented input (as we can see from the flip flop truth table), the result is either set or reset, which then propagates at each appropriate clock edge. We say that the configuration has registered the input value given to it, and is being shifted. It goes without saying that it is a clocked sequential circuit

Simulation

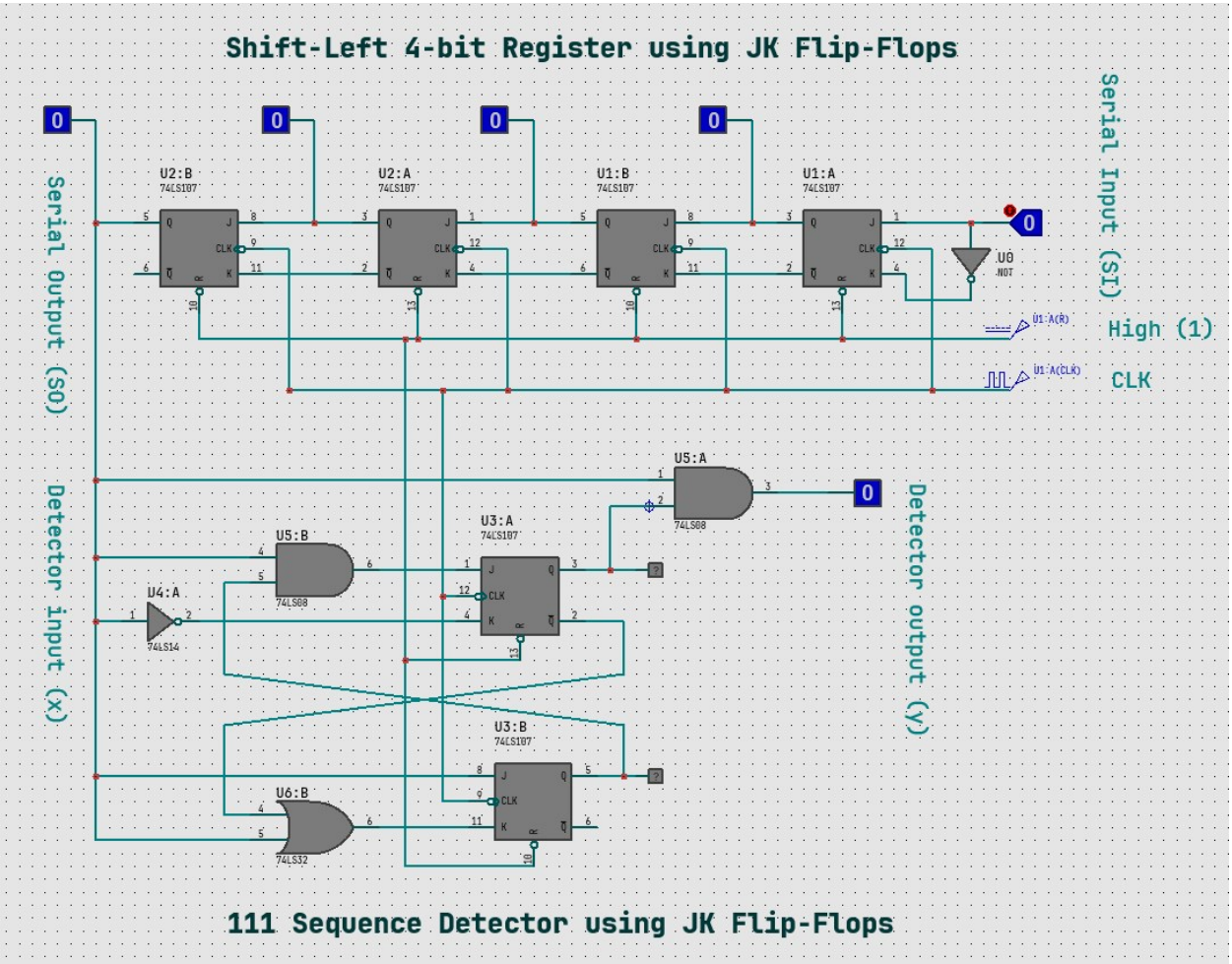
Below is the circuit schematic in the simulation software, complete with connections ready for simulation and also for PCB design.



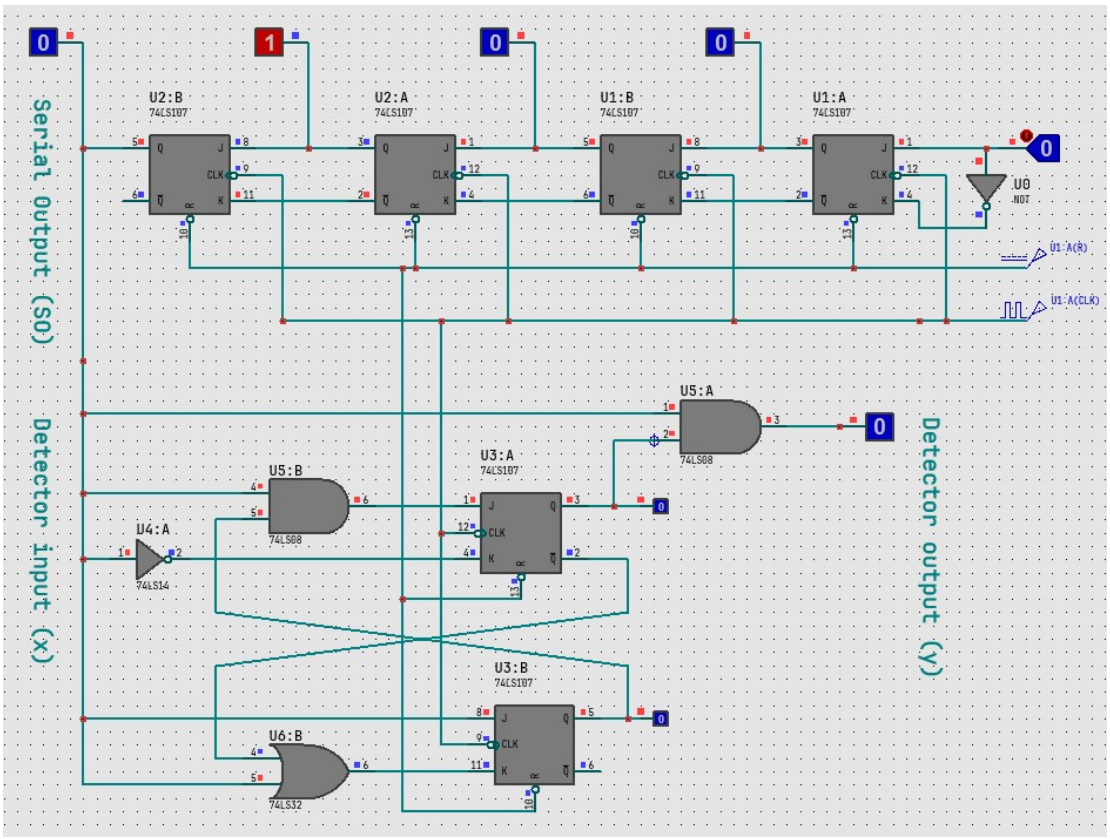

Shift-Left 4-bit Register using JK Flip-Flops
Serial Output (SO)
Serial Input (SI)
VCC
CLK
R3
100
R2
100
R1
100
R0
100
I3
I2
I1
I0
U2:B
74LS107
U2:A
74LS107
U1:B
74LS107
U1:A
74LS107
VCC
GND
SIL-100-02
SWITCH
GND
U0
NOT
VCC
CLK
PIN
J2(1)

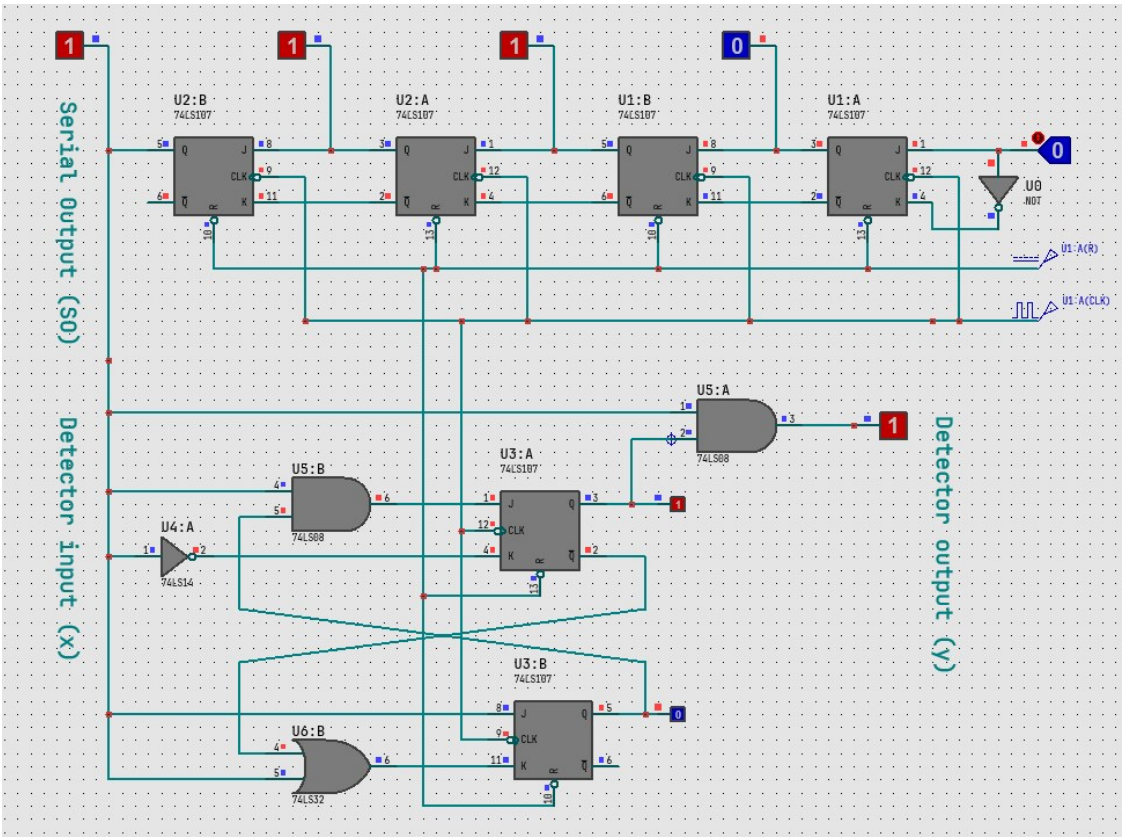

Detector input (x)
Detector output (y)
U5:A
74LS08
D
LED-RED
R0
100
GND
U3:A
74LS107
U5:B
74LS08
U4:A
74LS14
U3:B
74LS107
U6:B
74LS32
111 Sequence Detector using JK Flip-Flops

Replacing the LEDs with logic probes will be more convenient in seeing the result.

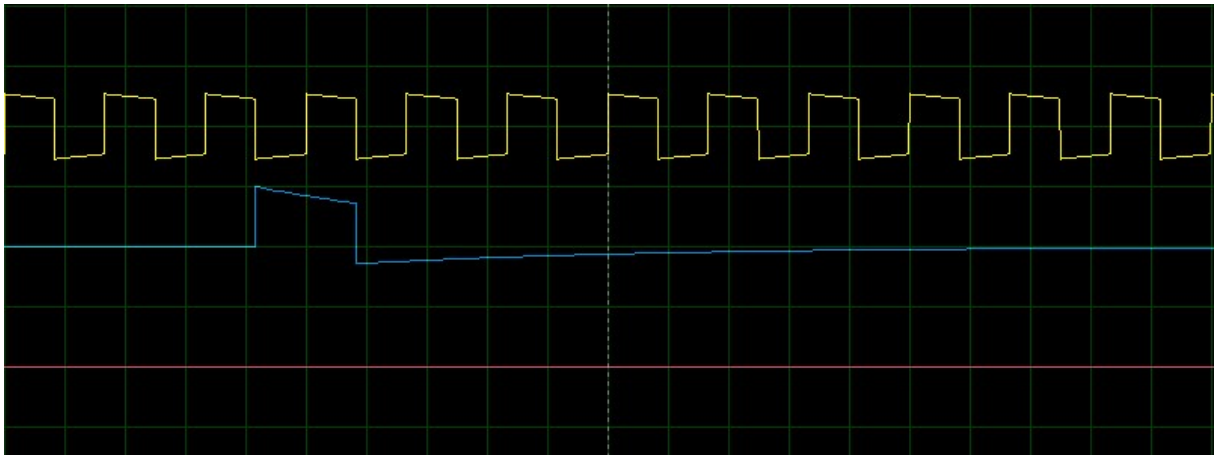


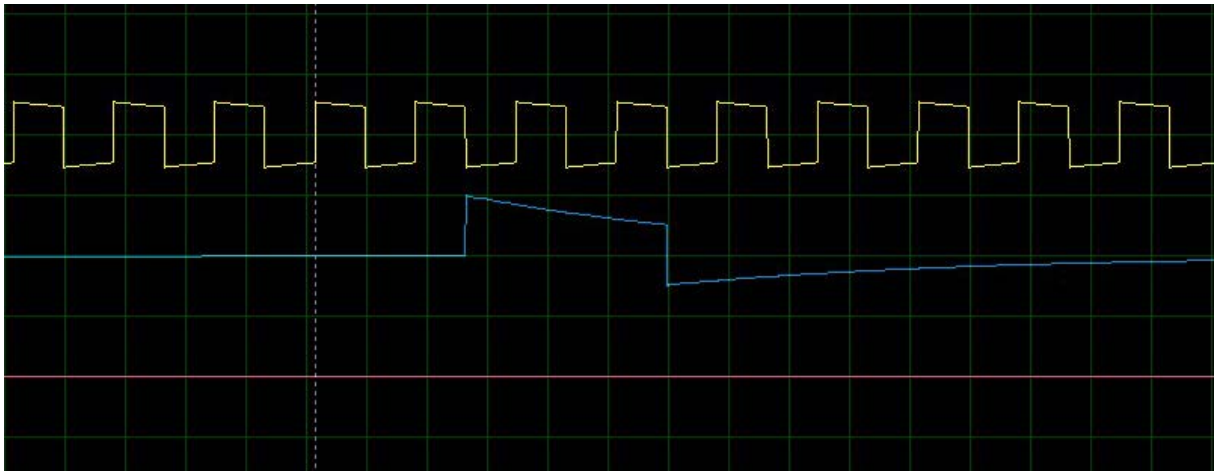
When the serial input logic toggle is kept at zero, or when smaller one or two pulses of 1 are applied, the detector output remains unactivated. But right when the complete three consecutive sequence of 1s are detected it turns on.

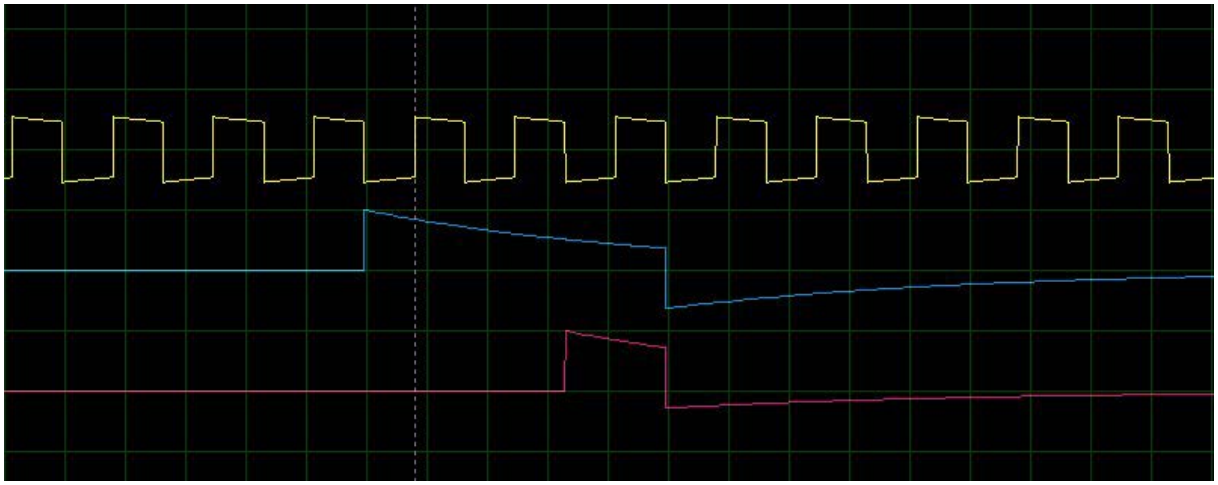

Serial Output (SO)
Detector input (x)
Detector output (y)
U2:B
74LS107
U2:A
74LS107
U1:B
74LS107
U1:A
74LS107
U0
NOT
U1:A(R)
U1:A(CLK)
U5:A
74LS08
U5:B
74LS08
U3:A
74LS107
U4:A
74LS14
U3:B
74LS107
U6:B
74LS32

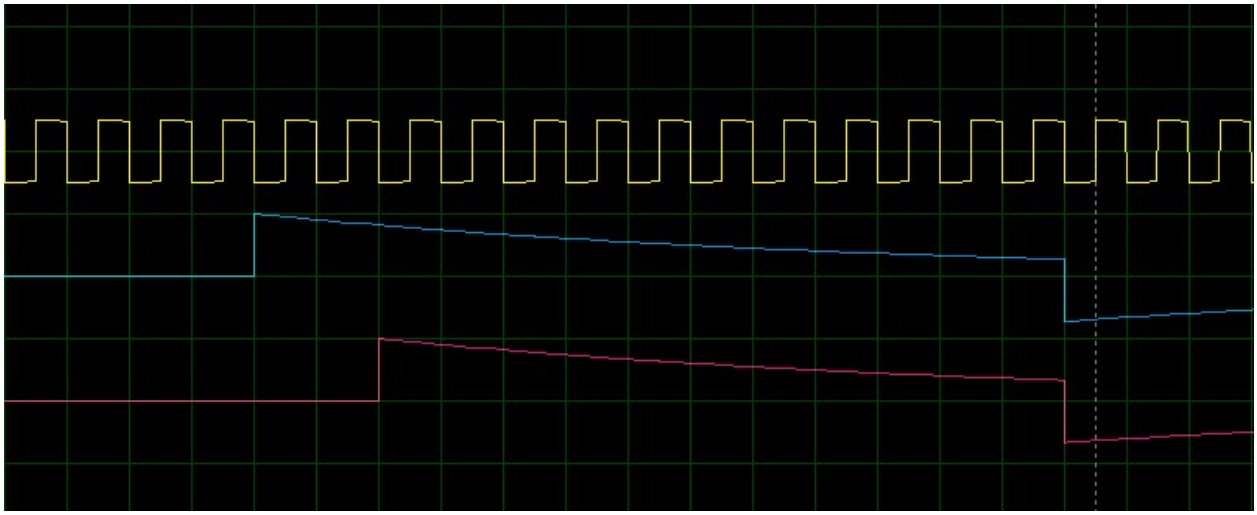

Serial Output (SO)
Detector input (x)
Detector output (y)
U2:B
74LS107
U2:A
74LS107
U1:B
74LS107
U1:A
74LS107
U0
NOT
U1:A(R)
U1:A(CLK)
U5:A
74LS08
U5:B
74LS08
U3:A
74LS107
U4:A
74LS14
U3:B
74LS107
U6:B
74LS32

The input timeline and the digital response to it can be observed on the oscilloscope, where we have **clock signals**, the **serial output**, and the **detector output** at corresponding colors.









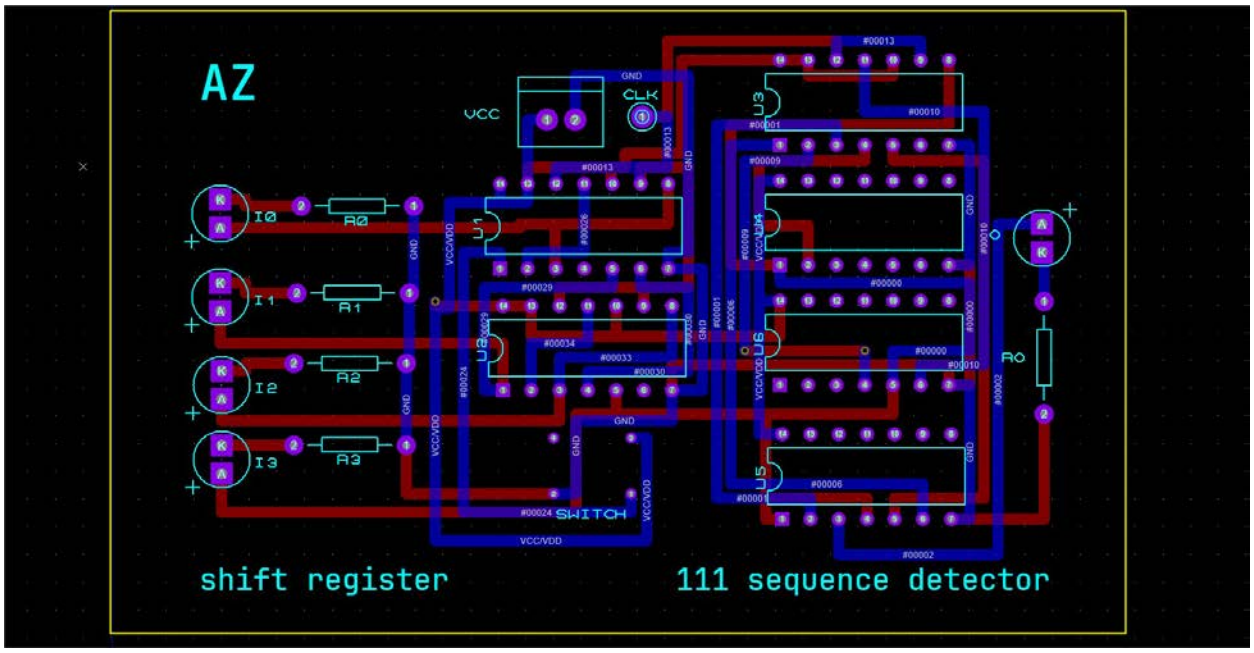
As is clear, the detector only gives an output when the serial output remains high for 3 clock cycles, i.e. passes 3 consecutive 1s.

But also notice, that the detector remains activated when the number of consecutive 1s increases. For it was designed so, that the detector output remains at state S_3, for 3 or more consecutive 1 inputs and only drops down to state S_0 when the next input is 0.

It must be noted that changes occur at the falling edge because negative edge trigger flip flops are used.

PCB

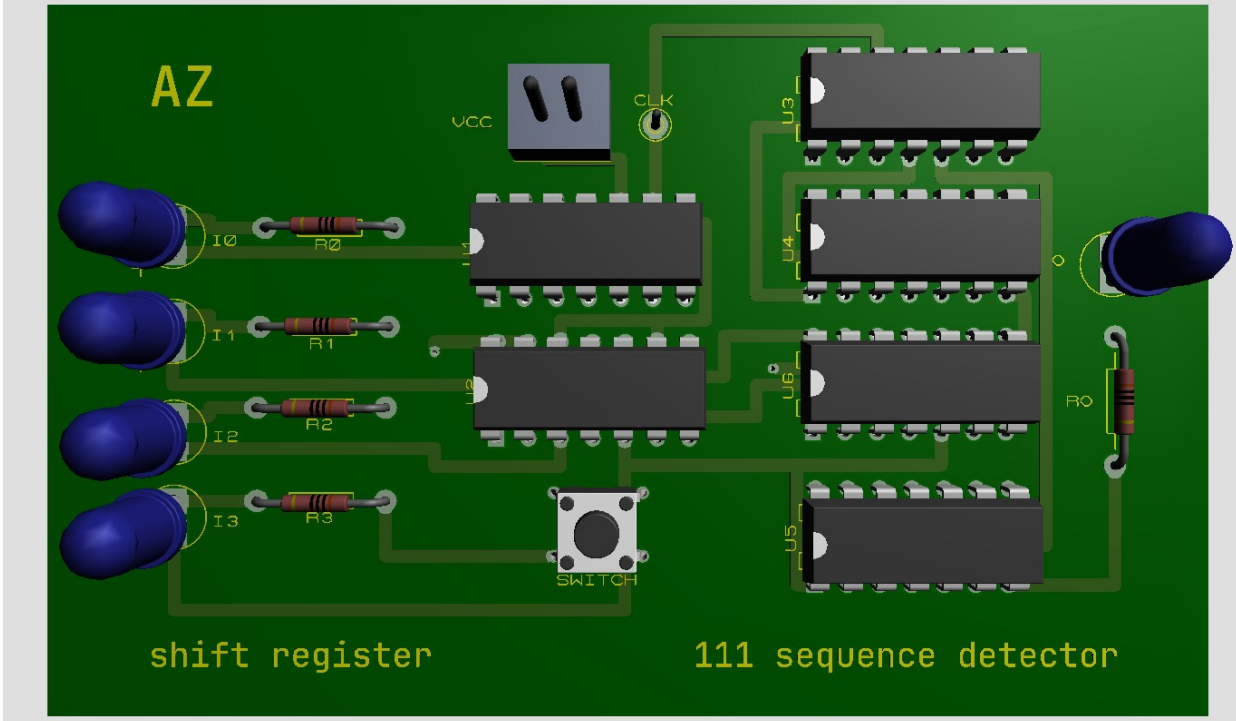
Given below is the blueprint for the PCB of the circuit. It's a one sided copper plate PCB.

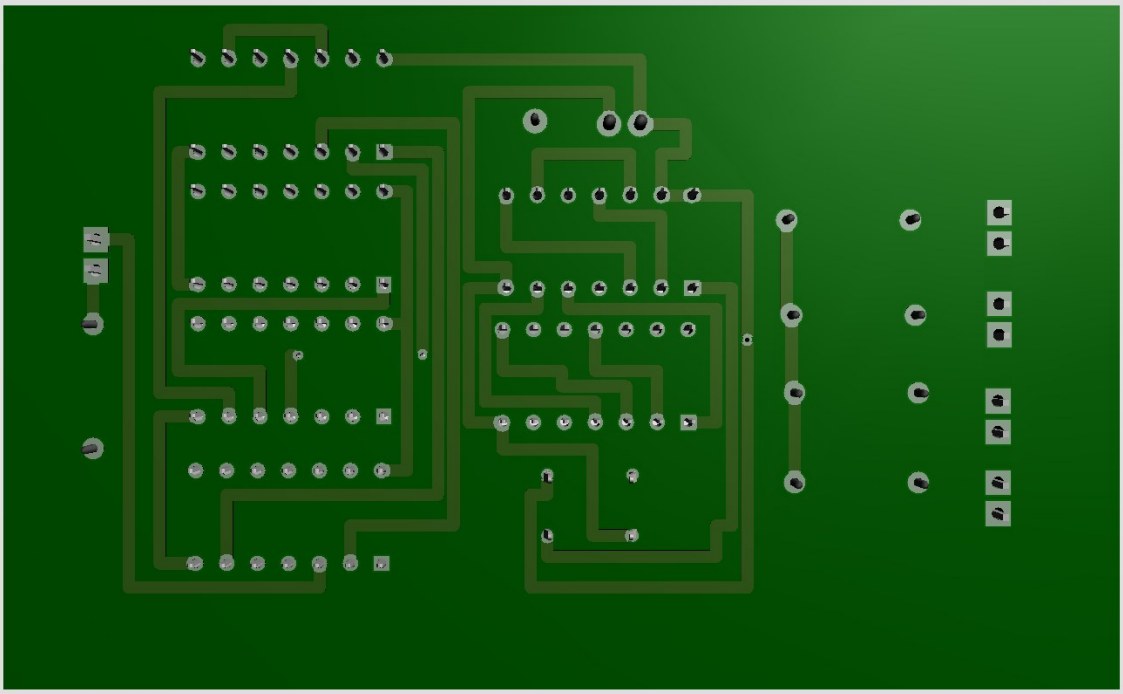


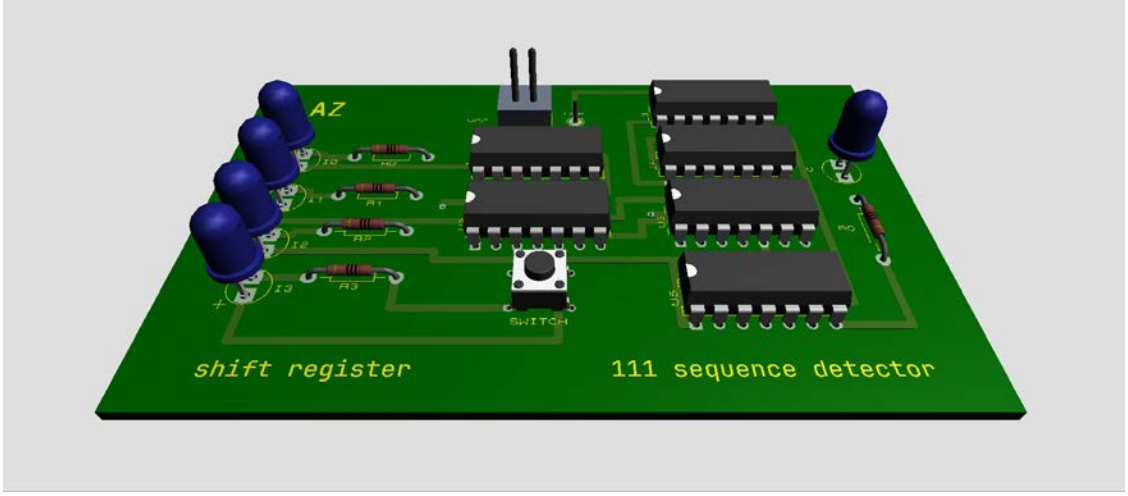
Components used:

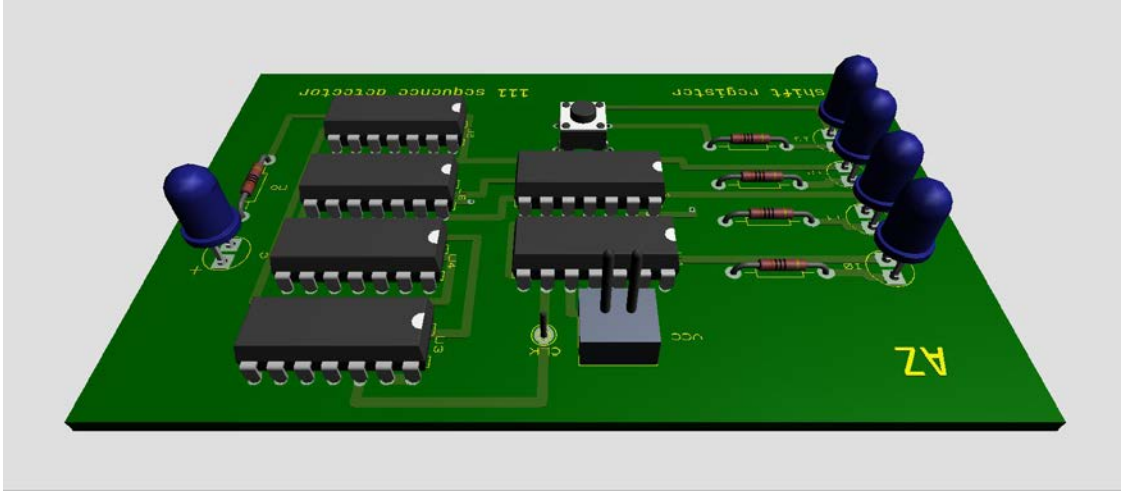
- Some gates. One 74LS08 AND Gate. One 74LS32 OR Gate and one 74LS14 NOT Gate.
- Three 74LS73 negative edge triggered JK Flip-Flop ICs. Two ICs for the serial register, and one for the detector
- Four LEDs to indicate the serial input sequence of the serial register. One LED for the detector output
- Five low 100Ω resistors connected to the LEDs to prevent them from floating in the circuit.
- One switch to input a sequence to the register then detector.

Here are the final PCB 3D visualizations in various perspectives:

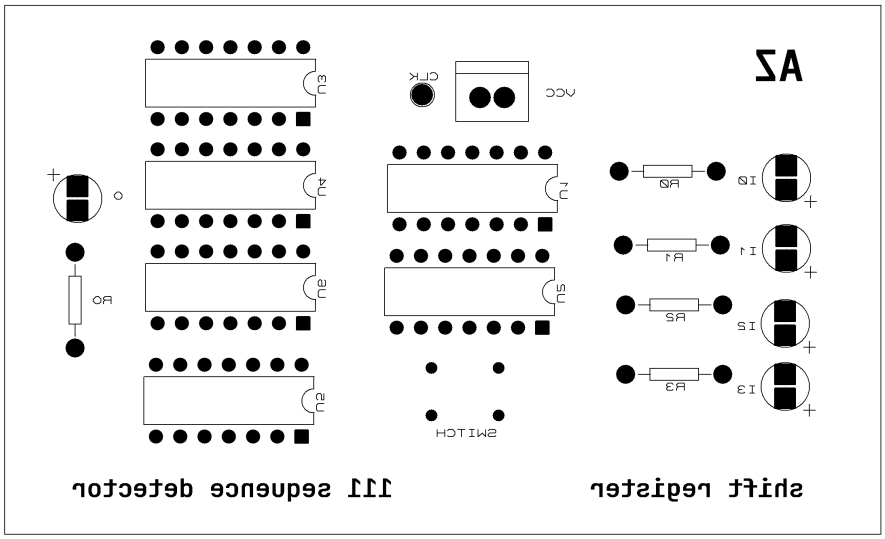

AZ
VCC
CLK
U3
U4
U6
U5
U1
U2
I0
I1
I2
I3
R0
R1
R2
R3
O
RO
SWITCH
shift register
111 sequence detector

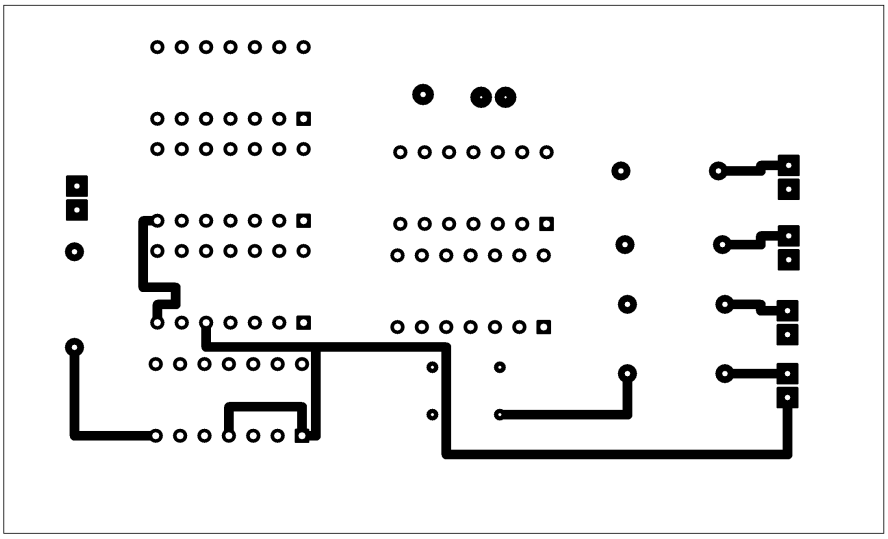


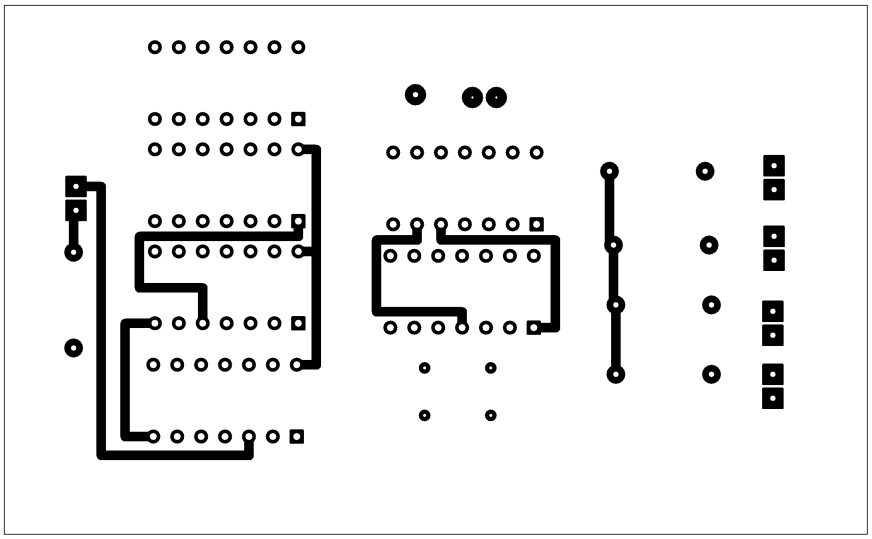




PCB Print-Outs







Top and Bottom copper respectively.

all files available

https://github.com/az-yugen/EEE-2002-4-6.-LAB