Faculty of Engineering and Technology

Department of Electrical and Computer Engineering

ENCS 211

Digital Electronics and Computer Organization Lab

Experiment No. 5 -Sequential Logic Circuits

5.1 OBJECTIVES

- To understand the differences between combinational and sequential Logic circuits; and the applications of various memory units.
- To study the operating principles and applications of various flip-flops.
- To understand the operating principles of counters and how to construct counters with JK flip-flops.
- To study the synchronous and asynchronous counters

5.2 EQUIPMENT REQUIRED

- IT-3000 Basic Electricity Circuit Lab.
- IT-3007 J-K Flip-Flop Circuits.
- IT-3008 Flip-Flop Circuits.

5.3 LABORATORY REGULATIONS AND SAFETY RULES

The following Regulations and Safety Rules must be observed in the laboratory:

1. It is the duty of all concerned who use any electrical laboratory to take all reasonable steps to safeguard the HEALTH and SAFETY of themselves and all other users and visitors.
2. Be sure that all equipment is properly working before using them for laboratory exercises. Any defective equipment must be reported immediately to the Lab. Instructors or Lab. Technical Staff.
3. Students are allowed to use only the equipment provided in the experiment manual or equipment used for senior project laboratory.
4. Power supply terminals connected to any circuit are only energized with the presence of the Instructor or Lab. Staff.
5. Students should keep a safe distance from the circuit breakers, electric circuits or any moving parts during the experiment.
6. Avoid any part of your body to be connected to the energized circuit and ground.
7. Switch off the equipment and disconnect the power supplies from the circuit before leaving the laboratory.
8. Observe cleanliness and proper laboratory housekeeping of the equipment and other related accessories.
9. Double check your circuit connections before switching “ON” the power supply.
10. Make sure that the last connection to be made in your circuit is the power supply and first thing to be disconnected is also the power supply.
11. Equipment should not be removed, transferred to any location without permission from the laboratory staff.
12. Software installation in any computer laboratory is not allowed without the permission from the Laboratory Staff.
13. Computer games are strictly prohibited in the computer laboratory.
14. Students are not allowed to use any equipment without proper orientation and actual hands-on equipment operation.
15. Smoking and drinking in the laboratory are not permitted.

5.4 PRE LAB

1. Prepare all sections and be ready for a quiz!
2. For each used IC, search for its datasheet that explains exactly what a component does and how to use it.

5.5 INTRODUCTION

5.5.1 Sequential Circuits

Any digital circuit could be classified as either a combinational or a sequential circuit. Combinational logic circuits implement Boolean functions. Boolean functions are mappings of inputs to outputs. These circuits are functions of input only.

Sequential circuits are *two-valued* networks in which the outputs at any instant are dependent not only upon the inputs present at that instant, but also upon the past history (sequence) of inputs. The block diagram of a sequential circuit is shown in Figure 5.1. The basic logic element that provides memory in many sequential circuits is the *flip-flop*.

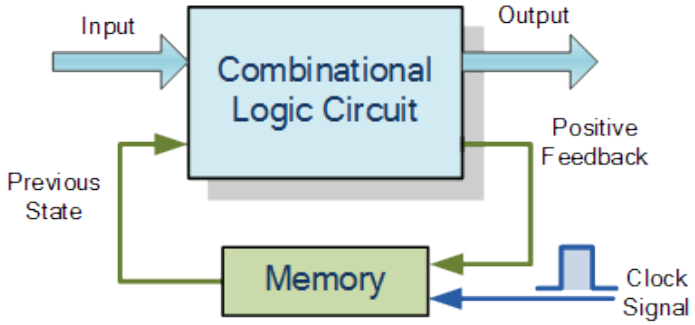

Figure 5.1: Sequential Circuit Block Diagram

5.5.2 Latches

Latches form one class of flip-flops. This class is characterized by the fact that the timing of the output changes is not controlled. Although latches are useful for storing binary information and for the design of asynchronous sequential circuits, they are not practical for use in synchronous sequential circuits.

5.5.2.1 The SR (Set-Reset) Latch

It is a circuit with two cross-coupled NOR or NAND gates.

a. SR latch with NAND gates:

The one with *NAND* gates is shown in ***Figure 5.2***. Note that this circuit is ***active low*** set/reset latch; that means the output Q goes to 1 when S (set) input is 0 and goes to 0 when R (Reset) input is 0 as shown in ***Table 5.1***. The condition that is undefined is when both inputs are equal to 0 at the same time.

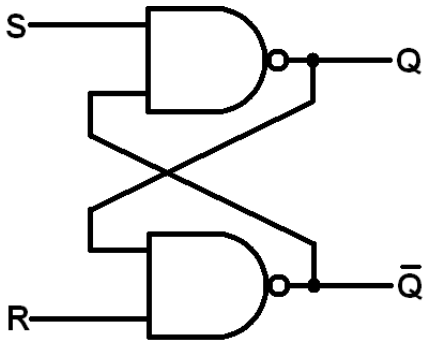

Figure 5.2: SR latch with NAND gates.

Table 5.1: SR latch with NAND gate Truth table.

INPUT		OUTPUT		State
S	R	Q	$\overline{Q}$	
1	0	0	1	SET
1	1	0	1	No Change/ Memory
0	1	1	0	RESET
1	1	1	0	No Change/ Memory
0	0	1	1	Invalid

b. SR latch with NOR gates:

Design the Logic Diagram, function table of the SR latch using NOR gates, and explain how it works. **(Pre lab)**

c. RS latch with control input:

The RS latch with control input C is shown in ***Figure 5.3***. If C=0 the output Q does not change regarding less the R and S values. If C= 1the circuit will work normally as shown in ***Table 5.2***.

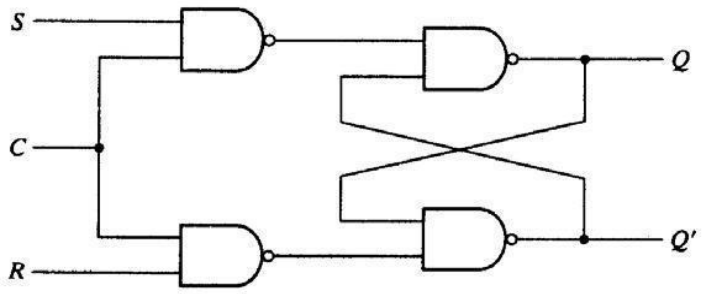

Figure 5.3: RS latch with control input.

Table 5.2: RS latch with control truth table.

INPUT			OUTPUT		State
C	S	R	Q_{n+1}	Q´	
0	X	X	Q_n	Q´	No Change/ Memory
1	0	0	Q_n	Q´	No Change/ Memory
1	0	1	0	1	RESET
1	1	0	1	0	SET
1	1	1	0	0	Indeterminate

5.5.2.2 The D Latch

The D latch was developed to eliminate the undefined condition of the indeterminate state in the RS latch. The D latch and its state table is shown in ***Figure 5.4*** and ***Table 5.3***.

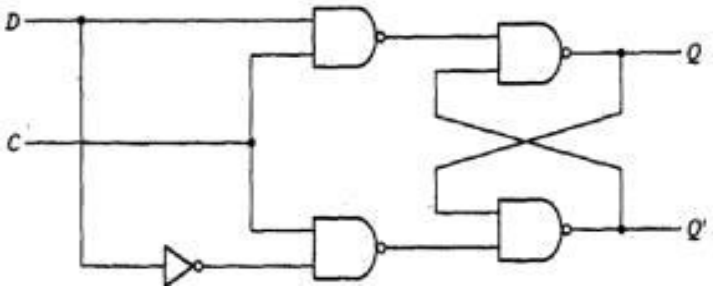

Figure 5.4: D-Latch.

Table 5.3: D latch truth table.

INPUT		OUTPUT		State
C	D	Q_{n+1}	Q´	
0	X	Q_n	Q´	No Change/ Memory
1	0	0	1	RESET
1	0	1	0	SET

5.5.3 Flip-Flops

Like latches, flip-flops are also used for storing binary information, but the difference is: The output change in the flip-flop occurs only at the clock edge while in the latch it occurs at the clock level.

A flip-flop can be implemented using two separate latches. ***Figure 5.5*** shows the D flip-flop implemented with two D latches.

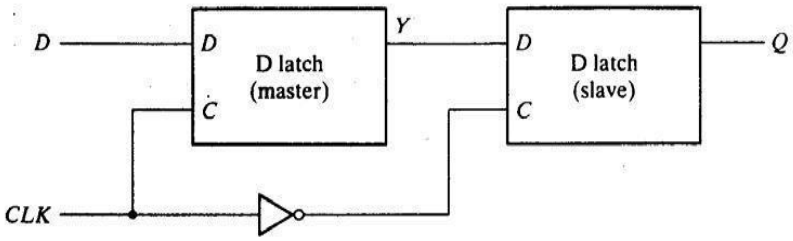

Figure 5.5: D flip flop implemented with two D latches.

There are several types of flip-flops, the common ones are D, T, and JK flip flops. ***Figure 5.6*** shows these flip flops and their function tables.

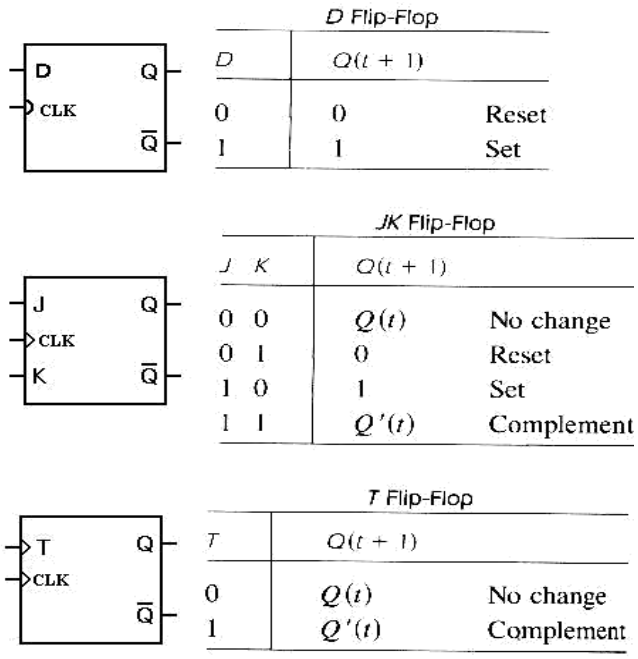

Figure 5.6: D, JK, and T flip flops.

5.5.4 Registers

Digital systems use registers to hold binary entities. The register is a collection of flip flops; N-bit register consists of N flip-flops. ***Figure 5.7*** shows simple 4-bit register implemented with D- flip flops. All the flip-flops are driven by a common clock, and all are reset simultaneously.

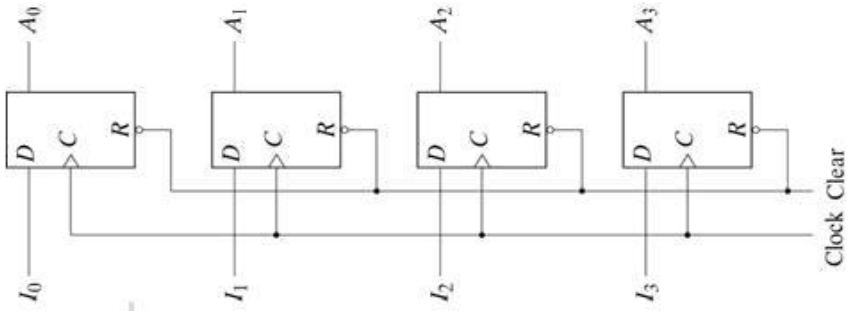

Figure 5.7: 4-bit Register.

Shift register is a group of flip-flops connected in a chain so that the output from one flip-flop becomes the input of the next flip-flop. ***Figure 5.8*** shows 4-bit shift-right register.

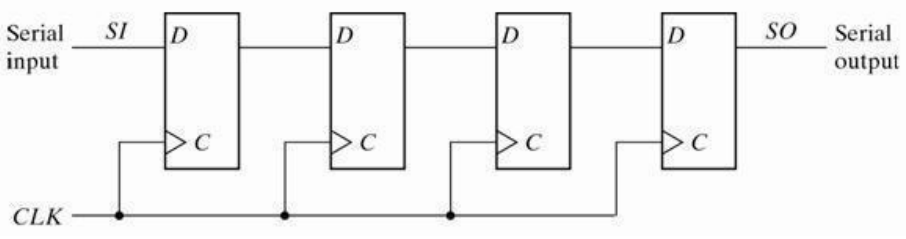

Figure 5.8: 4-bit shift- right register.

5.5.5 Counters

The counter is a special-purpose register; it is a register that goes through a prescribed sequence of states.

The counters are classified into two categories: Ripple and Synchronous counters. In ripple counters, there is no common clock; the flip-flop output transition serves as a source for triggering other flip-flops. In synchronous counters, all flip flops receive a common clock. ***Figure 5.9*** shows 3-bit ripple and synchronous counters.

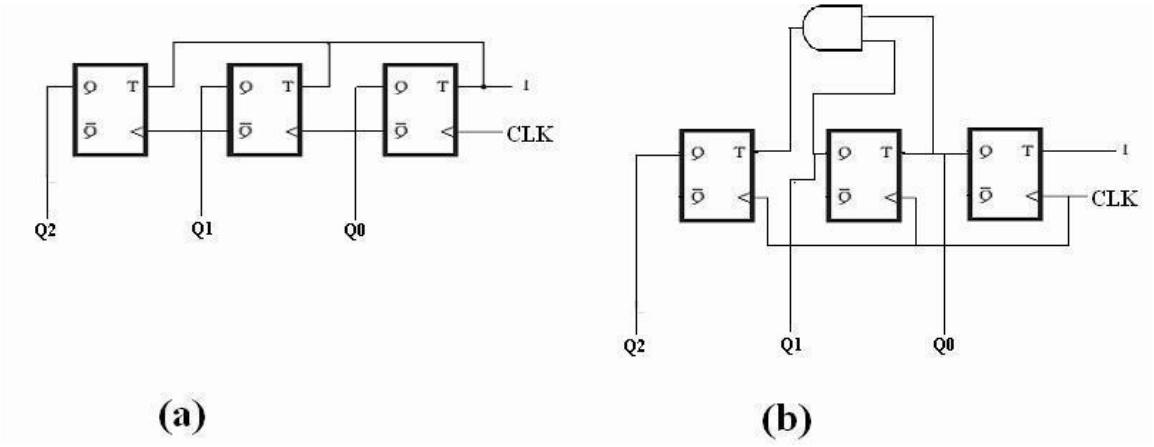

Figure 5.9: (a) 3-bit ripple counters, (b) 3-bit synchronous counter.

5.6 PROCEDURE

5.6.1 Latches and Flip flops:

A) Constructing RS latch with Basic Logic Gates

1. Use IT-3008 module to construct the circuit shown in ***Figure 5.10***.
2. Connect +5V of module IT-3008 to the +5V output of fixed power supply and GND of module IT-3008 to GND output of fixed power supply
2. Connect inputs A3, A4 to Switches SW1, SW2.
4. Connect outputs F6 and F7 to Logic Indicators L1, L2. Follow the sequences in ***Table 5.4***.

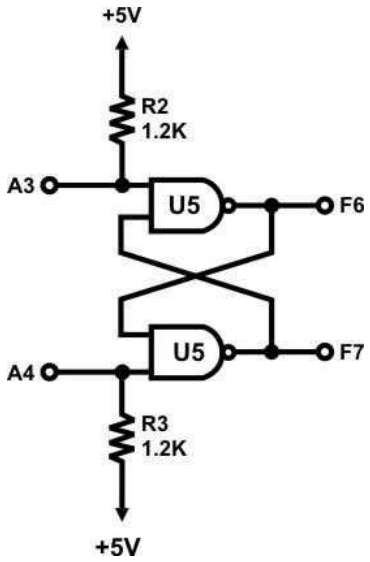

Figure 5.10: RS latch.

Table 5.4: Data for part 5.6.1 (A).

INPUT		OUTPUT	
A3	**A4**	**F6**	**F7**
0	**0**		
0	**1**		
1	**0**		
1	**1**		

B) Constructing RS latch with control input

1. Use IT-3008 module to connect the circuit shown in *Figure 5.11*.
2. Connect inputs A1, A5 to Switches SW1, SW2 and follow the input sequence in *Table 5.5*.

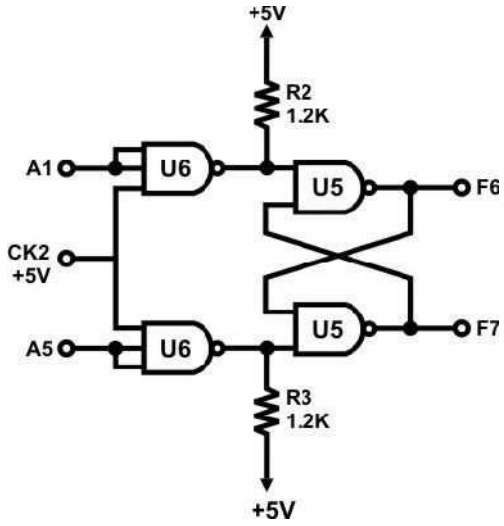

Figure 5.11: RS Latch with control input.

Table 5.5: Data for part 5.6.1 (B).

INPUT		OUTPUT	
A1	**A5**	**F6**	**F7**
0	**0**		
0	**1**		
1	**0**		
1	**1**		

C) Constructing D latch with RS latch

1. Use IT-3008 module to construct the circuit shown in ***Figure 5.12***.
2. Connect A1 to SW1; CK2 to SWA A and F6 to L1. Follow the sequences in ***Table 5.6***.

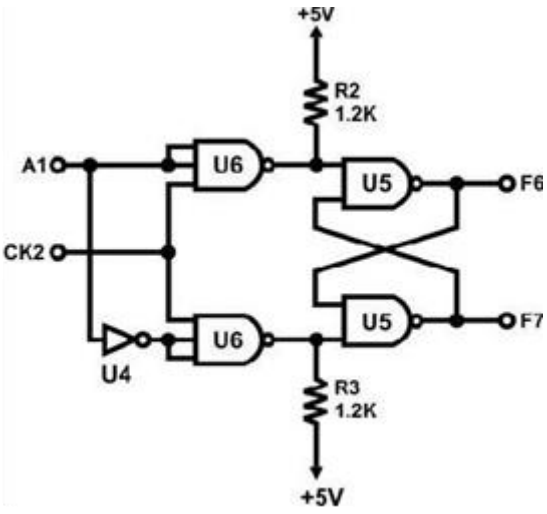

Figure 5.12: D Latch.

Table 5.6: Data for part 5.6.1 (C).

INPUT		OUTPUT
CK2	A1	F6
0	0	
0	1	
⎍	0	
⎍	1	

D) Constructing JK latch with RS latch

1. Use IT-3008 module to construct the circuit shown in ***Figure 5.13***.
2. Connect CK2 to SWB B output; A1 to SW0; A5 to SW1; F6 to L1.
3. Follow the sequences in ***Table 5.7***.

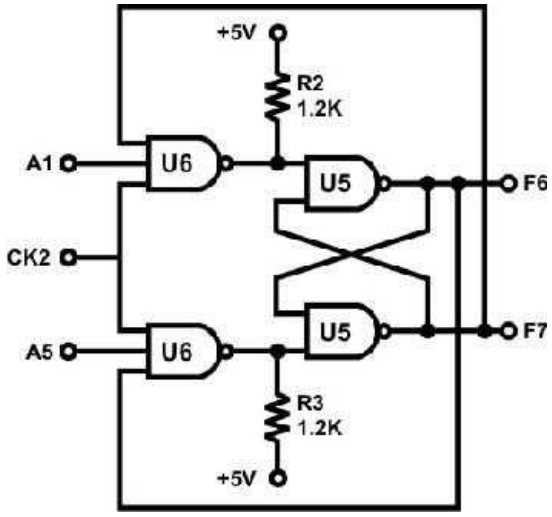

Figure 5.13: JK Latch.

Table 5.7: Data for part 5.6.1 (D).

INPUT			OUTPUT
CK2	A1	A5	F6
⎍	0	0	
⎍	0	1	
⎍	1	0	
⎍	1	1	

E) Constructing JK Flip-flop with master- slave RS latches

The master-slave flip-flop cancels all timing problems by using two SR flip-flops connected with each other. First flip-flop acts as the "Master" circuit, which triggers on the leading edge of the clock pulse while the other acts as the "Slave" circuit, which triggers on the falling edge of the clock pulse. This results in the two sections, the master section and the slave section being enabled during opposite half-cycles of the clock signal. Let us do this part by following these steps:

1. Use IT-3008 module to construct the circuit shown in ***Figure 5.14***.
2. Connect CK1 to SWA **A** output; J to SW1; K to SW0; F1, F2, F6, F7 to L3, L2, L1 and L0, respectively.
3. Follow the sequences in ***Table 5.8***.

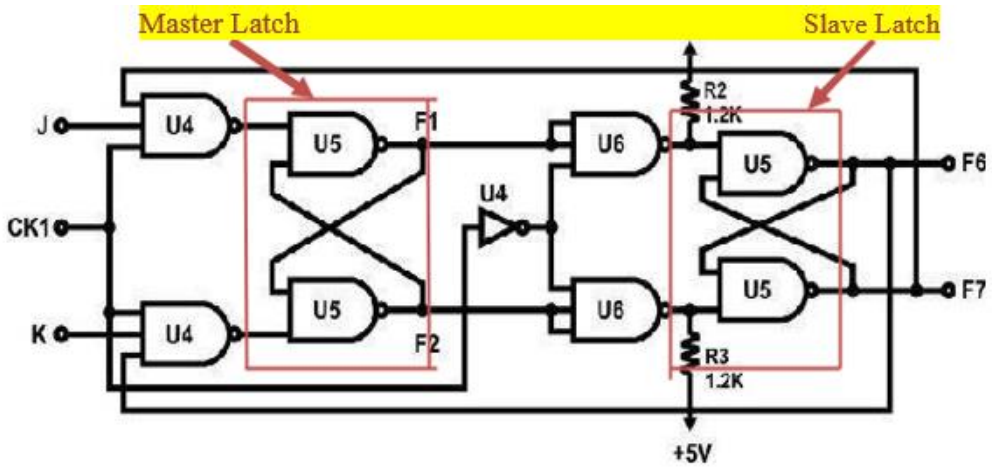

Figure 5.14: JK Flip-Flop.

Table 5.8: Data for part 5.6.1 (E).

INPUT			OUTPUT			
CK2	**K**	**J**	**F1**	**F2**	**F6**	**F7**
	**0**	**0**				
	**0**	**1**				
	**1**	**0**				
	**1**	**1**				
	**1**	**1**				

5.6.2 Registers

A) Constructing Shift Register with D Flip-Flops

1. Block Shift Register 1 of module IT-3008 will be used to construct the circuit shown in ***Figure 5.15***.

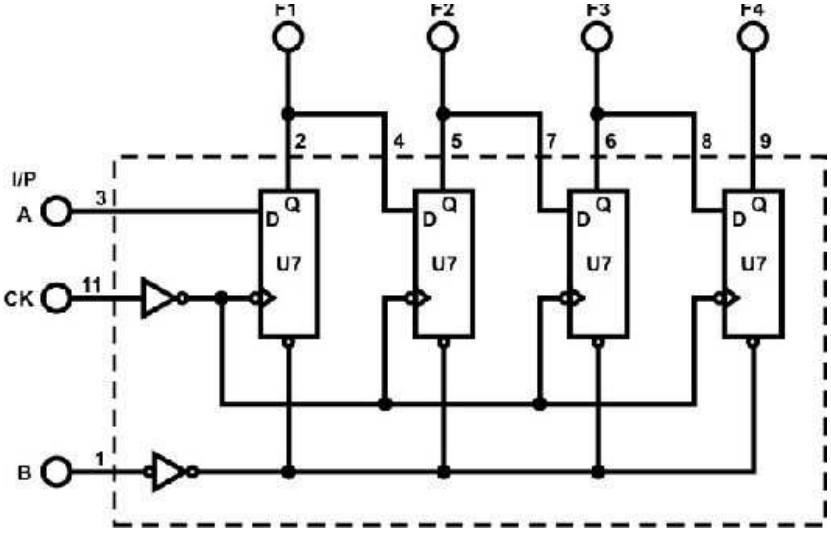

Figure 5.15: Shift Right Register.

2. Connect B (clear) to SW0; A (I/P) to SW1; CK to SWA A output; F1, F2, F3, F4 to L1, L2, L3, L4 respectively.
3. Set SW0 to "0" to clear B and then set SW0 to "1".
4. Follow the input sequence for A(I/P) below, observe output display at F1, F2, F3 and F4, and fill ***Table 5.9***:
   a) at A= "1", send in a CK signal from SWA
   b) at A= "0", send in a CK signal from SWA
   c) at A= "0", send in a CK signal from SWA
   d) at A= "1", send in a CK signal from SWA

Table 5.9: Data for part 5.6.2 (A).

INPUT		OUTPUT			
A	**CK**	**F1**	**F2**	**F3**	**F4**
1					
0					
0					
1					

B) 4-Bit Shift Register with serial and parallel load

Use Shift Register 2 module in IT-3008 which is 4-Bit Shift Register with serial and parallel synchronous operating modes, it has serial input (B1) and four parallel (A-D) Data inputs, and four Parallel Data outputs (QA–QD) as shown in ***Figure 5.16***.

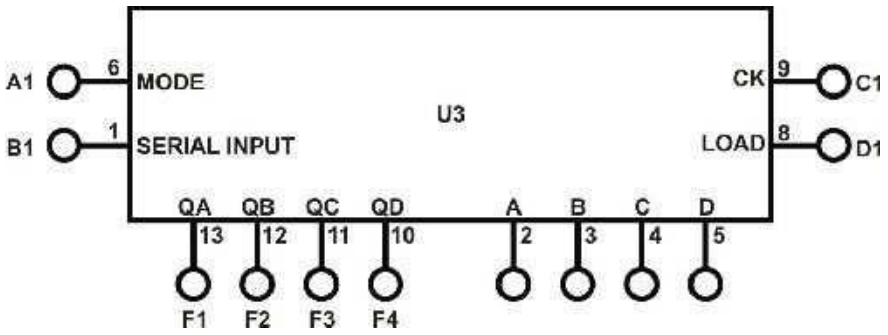

Figure 5.16: shift register with serial and parallel load.

1. Complete the following connections:

   - Connect Inputs A, B, C, D to SW0, SW1, SW2, SW3 Outputs F1, F2, F3, F4 to L0, L1, L2, L3, respectively.
   - B1 (I/P) to DIP2.0
   - A1 (MODE) to DIP2.1 as in ***Table 5.10***.

Table 5.10: A1 modes.

Input		
MODE Control (A1)	CK	
	C1	D1
L	⎍	X
H	X	⎍

2. Connect CK (C1) to the clock generator TTL level output at 1Hz and change data at B1 with DIP2.0. Follow the input sequences for A1 in ***Table 5.11***. Observe and record the outputs.

Table 5.11: Data for part 5.6.2 (B-2).

INPUT		**OUTPUT**			
A1	**C1**	**L3**	**L2**	**L1**	**L0**
0	⎍				
0	⎍				
0	⎍				
1	⎍				

3. Connect LOAD (D1) to the clock generator TTL level output at 1Hz. Set A1 to "1" and follow the input sequences for A, B, C and D in ***Table 5.12***. Observe and record the outputs.

Table 5.12: Data for part 5.6.2 (B-3).

Input					Output			
D1	**D**	**C**	**B**	**A**	**L3**	**L2**	**L1**	**L0**
	**0**	**0**	**1**	**0**				
	**1**	**0**	**1**	**0**				
	**1**	**1**	**1**	**0**				
	**0**	**1**	**1**	**1**				
	**0**	**1**	**1**	**0**				

5.6.3 Counters

A) 2-bit Synchronous Counter

1. Use IT-3007 module to implement the 2-bit synchronous counter shown in ***Figure 5.17***.
2. Connect +5V of module IT-3007 to the +5V output of fixed power supply and GND of module IT-3007to GND output of fixed power supply.
3. Connect CLK input to pulser switch SWA.
4. Connect counter outputs Q1 and Q0 to indication lamps L1, L2, respectively.
5. Apply clock pulses to CLK input. Observe and record the outputs (in binary) in ***Table 5.13***(a).
6. Apply counter outputs Q1 and Q0 to **seven segment** display. Observe and record the outputs (in decimal) in ***Table 5.13***(b).

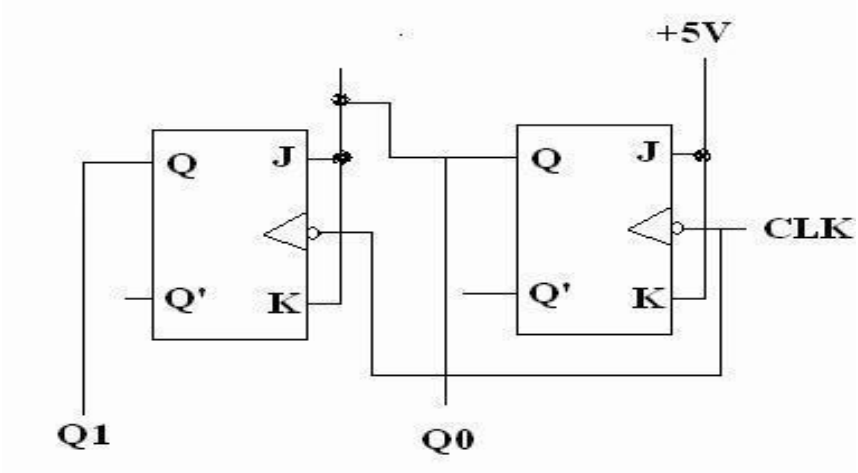

Figure 5.17: 2-bit Synchronous Counter.

Table 5.13: Data for part 5.6.3 (A).

Input	OUTPUT	
CLK	Q1	Q0
⎍		
⎍		
⎍		
⎍		
⎍		
⎍		
⎍		
⎍		

(a)

Input	OUTPUT
CLK	D
⎍	
⎍	
⎍	
⎍	
⎍	
⎍	
⎍	
⎍	

(b)

B) 3-bit (divide-by-eight) Ripple Counter

Divide-by-8 counter is 3-bit counter that counts from 0-to-7:

1. Use the IT-3007 module to implement the 3-bit (divide by eight) Ripple counter shown in ***Figure 5.18***.
2. Connect CLK input to pulser switch.
3. Connect counter outputs Q2, Q1 and Q0 to indication lamps.
4. Apply clock pulses to CLK input. Observe and record the outputs in ***Table 5.14***(a).
5. Apply counter outputs Q2, Q1 and Q0 to **seven segment** display. Observe and record the outputs in ***Table 5.14***(b).

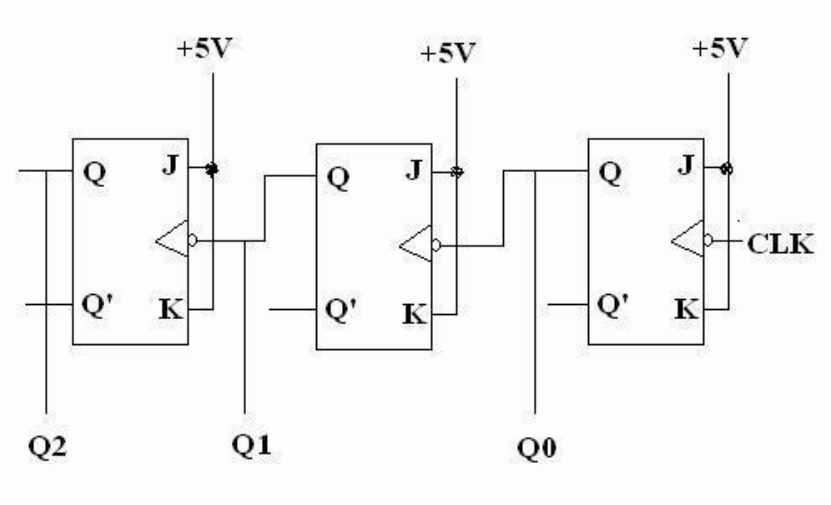

Figure 5.18: 3-bit Ripple Counter.

Table 5.14: Data for part 5.6.3 (B).

Input	OUTPUT		
CLK	Q2	Q1	Q0
⎍			
⎍			
⎍			
⎍			
⎍			
⎍			
⎍			
⎍			
⎍			
⎍			

(a)

Input	OUTPUT
CLK	D
⎍	
⎍	
⎍	
⎍	
⎍	
⎍	
⎍	
⎍	
⎍	
⎍	

(b)

Task2: Modify the circuit in *Figure 5.17* to be 3-bit **Synchronous Counter**. Attach the design with this experiment report.

C) BCD Counter

Locate the BCD counter (IC 7490) on IT-3008 module, which is shown in ***Figure 5.19***. Functional block diagram of U1 is shown in ***Figure 5.20***.

1. Connect +5V of module IT-3008 to the +5V output of fixed power supply and GND of module IT-3008 to GND output of fixed power supply.
2. Connect C3, C4 to SW0 and SW1; D1, D2 to SW2 and SW3; F1~F4 to L1~L4, A2 to SWA **A** output.
3. Connect F1 to B2, set C3, C4, D1 and D2 to ground and A2 to SWA **A** pulse. Measure and record the outputs F1, F2, F3, F4.
4. Set SW2 and SW3 to 0.

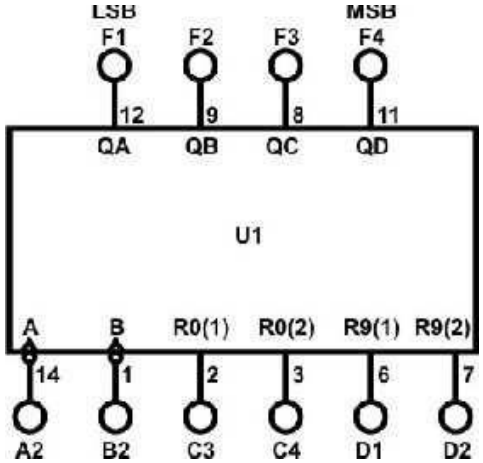

Figure 5.19: IC 7490 BCD Counter.

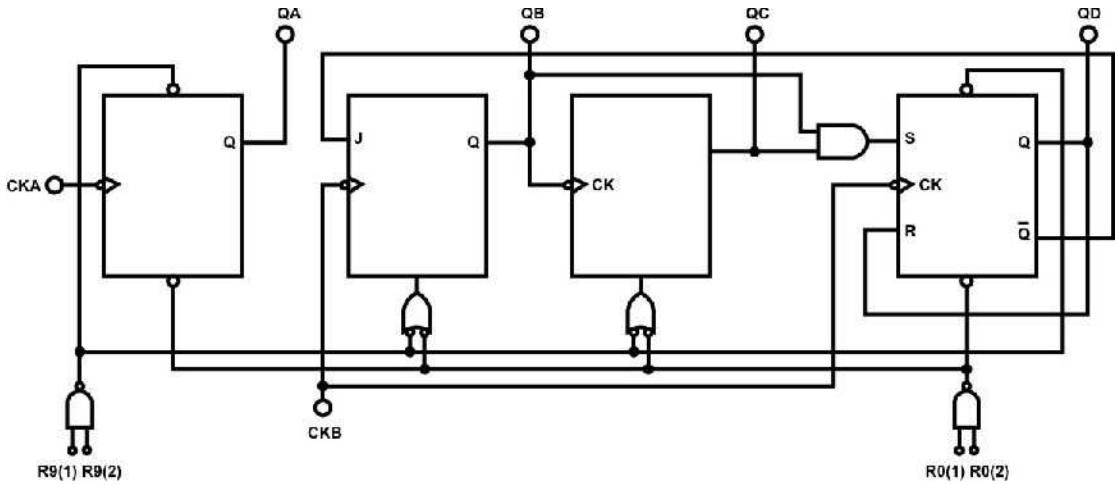

Figure 5.20: IC 7490 BCD Counter.

D) Divide-by-8 counter using BCD chip counter:

On the same circuit, do these modifications/changes then observe the result:

1. Change R0(2) (pin3) to +5V, and connect R0(1) (pin2) to QD (pin11) output. This will make counter reset after 111 (or 7). **WHY**?
2. Connect clock A2 (pin14) to pulser switch.
3. Connect the outputs A, B, C, and D to indication lamps.
4. Apply clock pulses to A2 and observe the count sequence (0000-0111).

Task3: change the connection of counter in Figure 5.19 **to count from:**

1. **0 – to – 5**
2. **0 – to – 4**

5.7 DISCUSSION

Answer the following questions:

1. Although latches are useful for storing binary information, they are rarely used in sequential circuit design, why?
2. What is the disadvantage of the RS flip flop?
3. What is the difference between “synchronous” and “ripple” counters?