Name	Muhammad Asad
Reg. #	2019-EE-383
Marks	

Experiment # 9

Introduction to sequential circuit and working of RS flip-flop

Objective:

- *Design and operation of latching circuits.*
- *Extended RS flip-flops*
- *RS flip-flop with dominant reset*

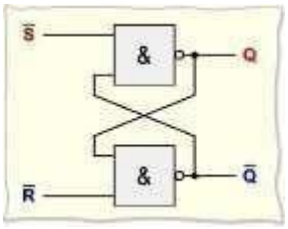
Apparatus:

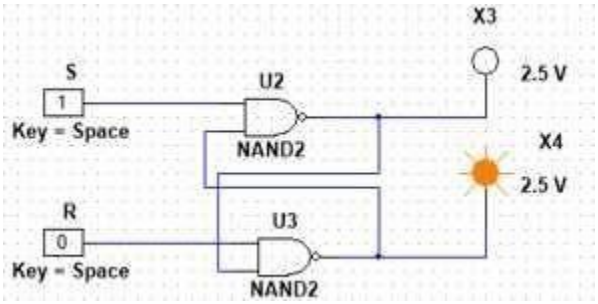
Workstation Core 2 Duo, Uni*Tr@inLucas Nulle*, *SO4201-9T*RS FFCard, Jumpers wires, Power Supplies.

Procedure:

- **RS - flip-flop**

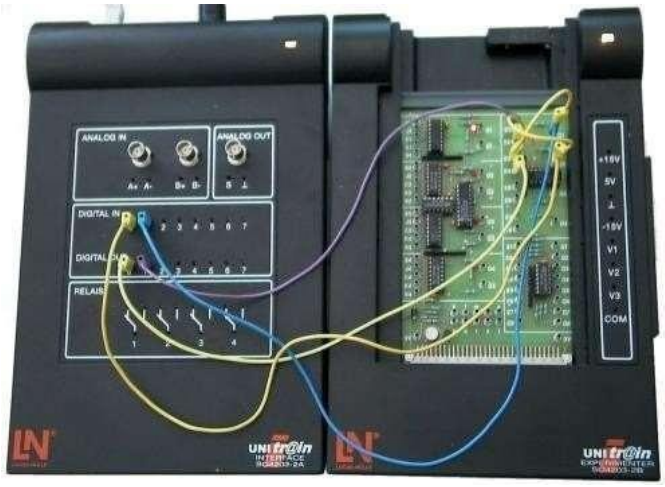
In the following experiments, a simple RS flip-flop, consisting of two connected NAND gates, is examined.





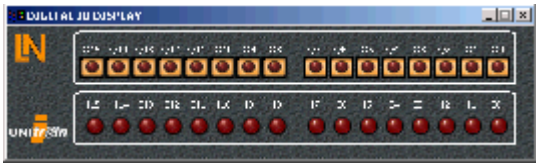
Experiment steps: RS - flip-flop	
1. Connect an Experimenter to the Uni*Tr@in*-I Interface and insert with the experiment card *SO4201-9T*. Then connect up the experiment card to the Uni*Tr@in*-	**List of connections** *From* | *To*

I Interface as shown in the illustration and as specified in the list of connections:



Digital Out 1	Terminal NAND E1
Digital Out 0	Terminal NAND E4
Digital In 1	Terminal NANDQ1
Digital In 0	Terminal NANDQ2
Terminal NAND E2	Terminal NAND Q2
Terminal NAND E3	Terminal NAND Q1

2. Select the following virtual instrument from *Instruments* menu
   - Digital Inputs and Outputs



Settings

Digital outputs	see Table

3. Use the input variables to generate the input states given in the Table and note down the values of the output variables.

Q_n designates the level **before** a new input signal combination is applied.

Q_{n+1} designates the level **after** a new input signal combination has been applied.

1	1	0	0	1	
1	1	1	1	0	

4. Demonstrate the memory (latching) capability of the circuit and summarize the results determined in the subsequent truth table.
5. Specify the individual functions of the RS flip-flop:

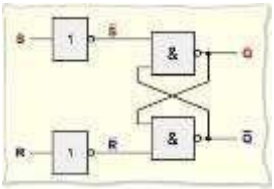
   a). disallowed
   b). set
   c). reset
   d). no change

6. What conclusions can you draw from the truth table regarding the practical applications of an RS flip-flop?

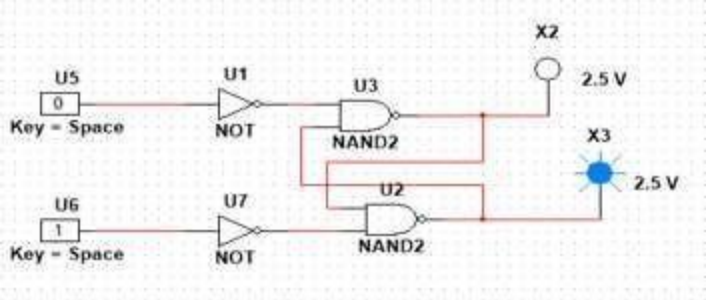
   From the truth table of RS flip fioop , when both R and S having value (0) it disallowed the output both NAND gate invert values give 1 that is not physically understandable both Q and Q' have same value 1. When the value given 0 and 1 respectively to S and R its sets the value and when values are given 1 and 0 then it reset the value and last when both values are 1 then its gives us no change and it is depend upon the present value of Q.

	S	R	Q_{n+1}	Q'_{n+1}
a).	0	0	1	1
b).	0	1	1	0
c).	1	0	0	1
d).	1	1	0	1

- **Expanded RS flip-flop**



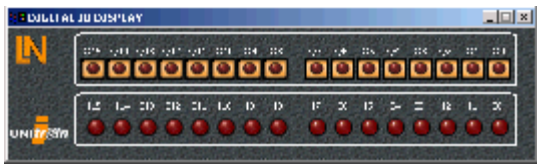
In the following experiment, we will investigate an extended RS flip-flop, consisting of two interconnected NAND gates and two inverters.



Experiment steps: Expanded RS flip-flop

1. Connect the experiment card to the sockets of the Uni*Tr@in*-I as specified in the list of connections on the right.

2. Now open the following virtual instrument from the *Instruments* menu
   - Digital inputs and outputs



3. Use the input variables to generate the input states specified in the Table and note down the values of the output variables.

   Q_n designates the level **before** a new input signal combination is applied.

   Q_{n+1} designates the level **after** a new input signal combination is applied.

4. Demonstrate the memory (latching) capability of the circuit and summarize the results determined in the following Truth table.
5. Describe the individual functions of the expanded RS flip-flop:

   a). disallowed
   b). reset
   c). set
   d). no change

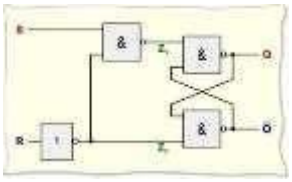
6. Describe the signal response of the expanded latch.

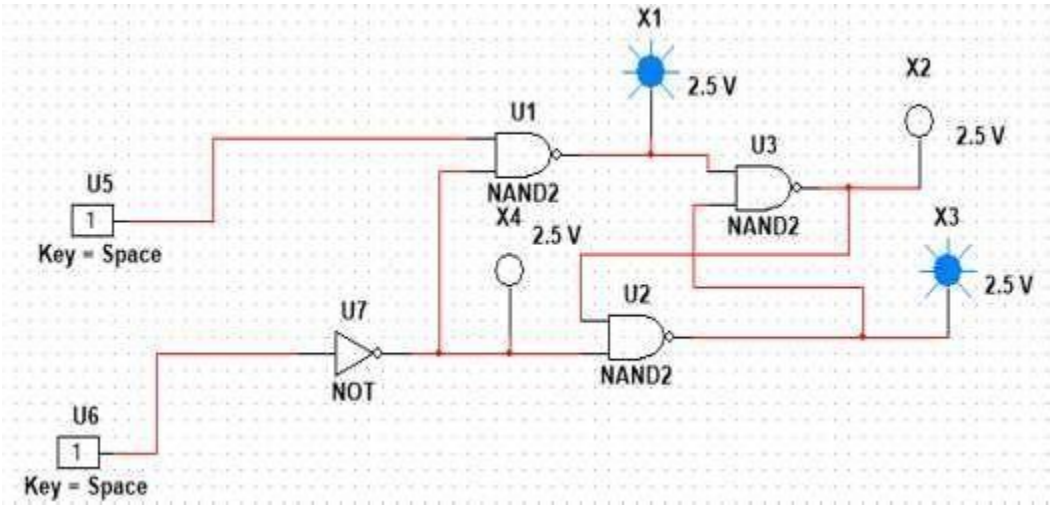
   From the truth table of RS flip flop, when both R and S having value (1) then it disallowed the output because NOT gate inverts the values and both NAND gate inverts value give 1 that is not physically

S	R	Q_{n+1}	Q'_{n+1}
1	1	1	1
0	1	0	1
1	0	1	0
0	0	1	0

- **RS flip-flop with dominant reset**

In the following experiments, we will be investigating a RS flip-flop with dominant reset consisting of three NAND gates and one inverter.





Experiment steps: RS flip-flop with dominant reset

1. Connect the experiment board to the sockets of the Uni*Tr@in*-I as specified in the list of connections on the right.

List of connections

From	*To*
Digital Out 1	Terminal NAND E5
Digital Out 0	Terminal NOT E1
Terminal NOT Q1	Terminal NAND E4
Terminal NOT Q1	Terminal NAND E6
Terminal NAND Q3	Terminal NAND E1

Terminal NAND E2	Terminal NAND Q2
Terminal NAND E3	Terminal NAND Q1
Digital In 3	Terminal NANDE1
Digital In 2	Terminal NANDE4
Digital In 1	Terminal NANDQ1
Digital In 0	Terminal NANDQ2

2. Now open the following virtual instrument from the *Instruments* menu
   - Digital inputs and outputs

Settings

Digital outputs	as specified in the Table

3. Use the input variables to generate the input states and note down the values of the output variables.

Q_n designates the level **before** the new input signal combination is applied.

Q_{n+1} designates the level **after** a new input signal combination is applied.

S	R	Q_n	Z_1	Z_2	Q_{n+1}	Q'_{n+1}
Q1	Q0	I1	I3	I2	I1	I0
0	0	0	1	1	0	1
0	0	1	1	1	1	0
0	1	0	1	0	0	1
0	1	1	1	0	0	1
1	0	0	0	1	1	0
1	0	1	0	1	1	0
1	1	0	1	0	0	1
1	1	1	1	0	0	1

4. Enter the results you have obtained into the following Table
5. Describe the individual functions of this type of RS flip-flop:

a). no change
b). reset
c). set
d). reset

6. What is the advantage of this configuration?

Ans: From the truth table of RS flip flop, when both R and S having value (1) then it disallowed the output because NOT gate inverts the values and both NAND gate inverts value give 1 that is not physically understandable both Q and Q' have same value 1. When the value given 0 and 1 respectively to S and R it sets the value and when values are given 1 and 0 then it reset the value and last when both values are 1 then its gives us no change value and it is depend upon the present value of Q.

	S	R	Z_1	Z_2	Q_{n+1}	Q'_{n+1}
a).	0	0	1	1	Q	Q'
b).	0	1	1	0	0	1
c).	1	0	0	1	1	0
d).	1	1	1	0	0	1

7. Enter your conclusions from the experiment RS flip-flops with particular respect to the two terms:

Latching : The SR flip-flop, also known as a SR Latch, can be considered as one of the most basic sequential logic circuit possible. This simple flip-flop is basically a one-bit

Undefined: