ISSN: 2277-9655
Impact Factor: 1.852

IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

Design Analysis and Circuit Enhancements of SR-Flip Flops

Prof. Olawale J. Omotosho[*1], Engr. Samson O. Ogunlere[2]
[*1,2]Babcock University, Computer Science Department, Ilishan-Remo, Ogun State, Nigeria
ogunlere@yahoo.com

Abstract

There are twelve possible design options for SR-Flip Flops but conventionally, two design options are in use. This paper thus focuses, on all the design options with a comparative analysis and implementation on why the two most used options are considered feasible in the design of digital device applications. From the analysis presented in this paper, it is obvious that conventional SR-Flip Flops that contains only two gates will not function fully as an SR-Flip Flop because there are four Active/transitional states in any SR-Flip Flop which suggests, a minimum number of gates that may be required to meet this requirement. The Active states are identified and analysed from the SR-Flip Flop Truth Table using the NOR or NAND gates which are the two design active elements of SR-Flip Flip. Using the conventional circuit diagram of SR-Flip Flops with two cross-couple gates, the number of transitions is observed to be less than the number of input combinations. Hence, it obvious that one of the transition states will be attached to two input combinations which in effect increase the usefulness of such Flip Flop from 25% to 50% utilization.

Key words - Flip Flops (Bistable Multivibrators), Conventional, Active state, Resting state, Forbidden state, NOR and NAND gates, K-Maps, Truth Table.

1. Introduction

Flip Flops are important digital electronic devices that have found very many uses in the development of computer systems. Flip Flops are electronically referred to as Bistable Multivibrators (BMs) which are configured with two appropriately biased transistors that are connected back-to-back to produce two stable state outputs. The detailed electronic circuitry of BMs is not the purpose of this paper. The digital counterpart is built up by using digital gates rather than using discrete electronic elements (Transistors, Resistors, Capacitors, etc) which is the approach this paper will adopt to present the analysis and design of Flip Flops.

Four types of Flip Flops are currently in the market, design of which has been established many years back. A close study of the design reveals that there is basically only one type of Flip Flop referred to as SR-Flip Flop. The other three types are derived from this basic one. An SR-Flip Flop is examined in details in section 2 of this paper.

2. Analysis of SR-Flip Flops

SR-Flip Flops are two-input and two-output devices diagrammatically presented in Figure 1.

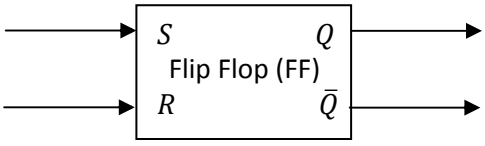

Figure 1: SR-Flip Flop Symbol

ISSN: 2277-9655
Impact Factor: 1.852

The inputs and outputs are defined as follows:
S = SET responsible to set the Flip Flop into an output of logic 1
R = RESET responsible to reset the Flip Flop into an output of logic 0
$\bar{Q}$= Complemented output of the Flip Flop
Q= Uncomplemented output of the Flip Flop which is normally referred to as the actual Flip Flop output.

Since there are two input variables, there are four possible combinations of these variables that will cause possible transitions of the output of the Flip Flop to take place. These are presented in Table 1

Table 1: FF Input Combinations

SR (Letters)	SR (Binary)	SR (Decimal)
$\bar{S}\bar{R}$	00	0
$\bar{S}R$	01	1
$S\bar{R}$	10	2
SR	11	3

Three transitions of the output of SR-Flip Flops are well known as follows:

- **Resting State:** Occurs when the output of the Flip Flop does not change state whenever it receives any input combination.
- **Active State:** Occurs when the output of the Flip Flop change state whenever it receives any input combination.
- **Forbidden State:** Occurs when the output of the Flip Flop does not know whether to change state or not whenever it receives any input combination. That is, the Flip Flop is in a state of confusion. Hence, such conditions MUST be prevented. Thus, input combinations that give rise to such a state are not allowed. These are forbidden.

Since the number of transitions is less than the number of input combinations, it becomes obvious that one of the transition states will be attached to two input combinations. It is reasonable to expect that it is the **Active State** that will require two transitions in order to increase the usefulness of such Flip Flop from 25% to 50% utilization. Consequent upon this logic, the number of configurations of SR-Flip Flops that can be derived is presented in Tables 2 & 3

Table 2: Possible Design Options of an SR-Flip Flop

Transition States	SR (Decimal)	SR (Decimal)	SR (Decimal)	SR (Decimal)	SR (Decimal)	SR (Decimal)
RESTING	2	3	1	0	0	0
ACTIVE	0,1	0, 2	0, 3	1, 2	1, 3	2, 3
FORBIDDEN	3	1	2	3	2	1

Table 3: Possible Design Options of an SR-Flip Flop

Transition States	SR (Decimal)	SR (Decimal)	SR (Decimal)	SR (Decimal)	SR (Decimal)	SR (Decimal)

ISSN: 2277-9655
Impact Factor: 1.852

RESTING	3	1	2	3	2	1
ACTIVE	0,1	0, 2	0, 3	1, 2	1, 3	2, 3
FORBIDDEN	2	3	1	0	0	0

From Tables 2 & 3, there are twelve possible design options for SR-Flip Flops but the two most popular design options are the ones shaded on the table whose Truth Tables are shown in Table 4. Note that because of the sequential nature of Flip Flops (feedback), for each transition, there are eight transition states.

Table 4: Truth Table of SR-Flip Flop

Using Table 2						Using Table 3				
S	R	Q_n	Q_{n+1}	Transition State		S	R	Q_n	Q_{n+1}	Transition State
0	0	0	0	Resting		0	0	0	d	Forbidden
0	0	1	1	Resting		0	0	1	d	Forbidden
0	1	0	0	Active		0	1	0	1	Active
0	1	1	0	Active		0	1	1	1	Active
1	0	0	1	Active		1	0	0	0	Active
1	0	1	1	Active		1	0	1	0	Active
1	1	0	D	Forbidden		1	1	0	0	Resting
1	1	1	D	Forbidden		1	1	1	1	Resting

NOTES:
Q_n = previous output, Q_{n+1} = present output, d = I don't care term (0,1)
For active transition, using Table 1.2, Q_{n+1} = S
For active transition, using Table 1.2, Q_{n+1} = R

2.1 Logic Circuit of an SR-Flip Flop

The Truth Table of Table 4 is converted into a K-Map in order to obtain the minimized logic equations of the Flip Flop as follows: Table 5

Table 5: K-Map of Table 4

SR						SR				
Q_n	00	01	11	10		Q_n	00	01	11	10
0	0	0	d	1		0	d	1	0	0
1	1	0	D	1		1	d	1	1	0
$Q_{n+1} = S + \bar{R}Q_n \ldots\ldots (1a)$ $\bar{Q}_{n+1} = R + \bar{S}\bar{Q}_n \ldots\ldots (1b)$						$Q_{n+1} = \bar{S} + RQ_n \ldots\ldots (2a)$ $\bar{Q}_{n+1} = \bar{R} + S\bar{Q}_n \ldots\ldots (2b)$				

Logic equations (1a, 1b) & (2a, 2b) are derived from the K-Map and they can be used to construct the Flip Flop using the different gates as given by the equations. However, it is customary to use NOR and NAND gates to construct logic equations (1a, 1b) and (2a, 2b) respectively by converting the equations into these gates as follows:

$$Q_{n+1} = S + \bar{R}Q_n \ldots\ldots (1a)$$

$$\overline{\bar{R}Q_n} = \overline{R + \bar{Q}_n} \ldots\ldots (1c)$$

Put equation (1c) into (1a), we have

ISSN: 2277-9655
Impact Factor: 1.852

$$Q_{n+1} = S + \overline{R + \bar{Q}_n} \ldots\ldots (1d)$$

Complement equation (1d), we have

$$\bar{Q}_{n+1} = \overline{S + \overline{R + \bar{Q}_n}} \ldots\ldots (1e)$$

$$\bar{Q}_{n+1} = R + \bar{S}\bar{Q}_n \ldots\ldots (1b)$$

$$\overline{\bar{S}\bar{Q}_n} = \overline{S + Q_n} \ldots\ldots (1f)$$

Put equations (1f) into (1b), we have

$$\bar{Q}_{n+1} = R + \overline{S + Q_n} \ldots\ldots (1g)$$

Complement equation (1h), we have

$$Q_{n+1} = \overline{R + \overline{S + Q_n}} \ldots\ldots (1h)$$

Equations (1e) & (1h) are combined to construct the logic circuit of SR-Flip Flop using only NOR gates as shown in Figure 2

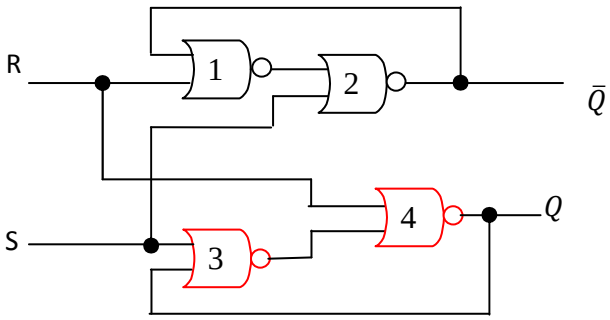

Figure 2: Logic Circuit of SR-Flip Flop Using NOR Gates

The same procedure is followed to obtain the logic circuit shown in Figure 3

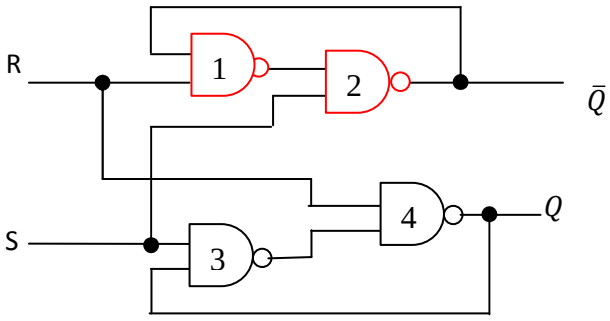

Figure 3: Logic Circuit of SR-Flip Flop Using NAND Gates

2.2 Alternative Logic Circuit of an SR-Flip Flop

Figures 2 & 3 are often reduced to only two gates that are connected as shown in Figure 4.1 & 5.1 without any logical explanation how it meets the requirements and functions of an SR-Flip flop. Figures 4.1 & 5.1 are analysed to prove that it does not represent fully an SR-Flip Flop.

2.3.1 SR-Flip Flop (NOR Gate Configuration)

This SR-Flip Flop will be examined by trying to answer the following question:

Given the two configurations shown in Figures 4.1 & 5.1, identify the configuration that is probable to be used as an SR-Flip Flip stating the conditions considered for your

ISSN: 2277-9655
Impact Factor: 1.852

choice. Can this choice of yours be used to satisfy fully the characteristics of a conventionally accepted SR-Flip Flop? If not, what modifications are necessary to meet the design requirements? If yes, how does it achieve the required characteristics?

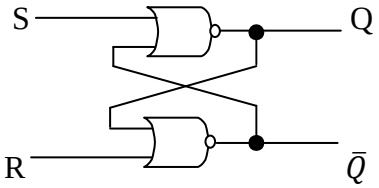

Figure 4.1

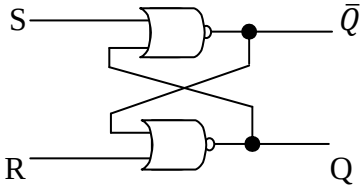

Figure 5.1

From Figure 1.4.1,

$$Q_{n+1}(1) = \overline{S + \bar{Q}_n} = \bar{S}Q_n = 0 - 1 = 001,011 \equiv \langle 1 \rangle \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (a)$$
$$\bar{Q}_{n+1}(0) = \overline{R + Q_n} = \bar{R}\bar{Q}_n = -00 = 000,100 \equiv \langle 0 \rangle \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (b)$$

From SR-Flip Flop (+ve Logic configuration), $SRQ_n = 110,111 \equiv \langle d \rangle$ …….. (c)

Plot equations (a), (b) & (c) into a K-Map and Truth Table as follows (see Table 6):

Table 6: Truth Table for Figure 4.1 Using Equations (a) Through (c) Above

S	R	Q_n	Q_{n+1}		REMARKS
0	0	0	0	The Value of Q_{n+1} is determined from equation (b)	Resting States
0	0	1	1	The Value of Q_{n+1} is determined from equation (a)	
0	1	0	A	This value of Q_{n+1} is indeterminate	A = 0 or 1
0	1	1	1	The Value of Q_{n+1} is determined from equation (a)	The output follows 'R' That is, Q_{n+1} = R
1	0	0	0	The Value of Q_{n+1} is determined from equation (b)	The output follows 'R' That is, Q_{n+1} = R
1	0	1	B	This value of Q_{n+1} is indeterminate	B = 0 or 1
1	1	0	d	The Value of Q_{n+1} is determined from equation (c)	Forbidden State
1	1	1	d	The Value of Q_{n+1} is determined from equation (c)	

K-MAP

SR

Q_n	00	01	11	10
0	0	A	d	0
1	1	1	d	B

$Q_{n+1} = \bar{S}Q_n = \overline{S + \bar{Q}_n}$

NOTES:

1. The Resting & Forbidden states are satisfied as an SR-Flip Flop
2. The Active states partly follows 'R' instead of 'S' for an SR-Flip Flop
3. The Active states partly give 2 indeterminate outputs, 'A' and 'B'. That is, the outputs can either follow 'S' or 'R' depending on the unspecified states
4. From K-Map, the equation for the Flip Flop output does not agree with that of a conventionally acceptable SR-Flip Flop.

Hence, this configuration (Figure 4.1) does not satisfy the required conditions for an SR-Flip Flop.

From Figure 5.1,

$$Q_{n+1}(1) = \overline{R + \bar{Q}_n} = \bar{R}Q_n = -01 = 001,101 \equiv \langle 1 \rangle \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (a)$$
$$\bar{Q}_{n+1} = \overline{S + Q_n} = \bar{S}\bar{Q}_n = 0 - 0 = 000,010 \equiv \langle 0 \rangle \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (b)$$

From SR-Flip Flop (+ve Logic configuration), $SRQ_n = 110,111 \equiv \langle d \rangle$ …… (c)

Plot equations (a), (b) & (c) into a K-Map and Truth Table as shown in Figure 7

Table 7: Truth Table for Figure 5.1 Using Equations (a) Through (c) above

S	R	Q_n	Q_{n+1}		REMARKS
0	0	0	0	The Value of Q_{n+1} is determined from equation (b)	Resting States

K-MAP

SR

Q_n	00	01	11	10

ISSN: 2277-9655
Impact Factor: 1.852

0	0	1	1	The Value of Q_{n+1} is determined from equation (a)			**0**	0	0	d	B
0	1	0	0	The Value of Q_{n+1} is determined from equation (b)	The output follows 'S'		**1**	1	A	d	1
0	1	1	A	This value of Q_{n+1} is indeterminate	A = 0 or 1						
1	0	0	B	This value of Q_{n+1} is indeterminate	B = 0 or 1	$Q_{n+1} = \bar{R}Q_n = \overline{R + \bar{Q}_n}$					
1	0	1	1	The Value of Q_{n+1} is determined from equation (a)	The output follows 'S'						
1	1	0	d	The Value of Q_{n+1} is determined from equation (c)	Forbidden State						
1	1	1	d	The Value of Q_{n+1} is determined from equation (c)							

NOTES:

1. The Resting & Forbidden states are satisfied as an SR-Flip Flop
2. The Active states partly follows 'S' as required for an SR-Flip Flop. That is, $Q_{n+1} = S$
3. The Active states partly give 2 indeterminate outputs,' A' and 'B'. That is, the outputs can either follow 'S' or 'R' depending on the unspecified states
4. From K-Map, the equation for the Flip Flop output does not agree with that of a conventionally acceptable SR-Flip Flop.

Hence, this configuration (Fig 5.1) does not fully satisfy the characteristics of an SR-Flip Flop because only half of its Active states will produce the required output while the other half may or may not produce the required output. However, this configuration is more acceptable than Fig 1 because half of its output follows 'S' as required for an SR-Flip Flop.

The modification required for Figure 5.1 is chosen from the four possible options as presented in Table 8

Table 8: Four Possible Options to Modify Figure 5.1

$Q_{n+1} = \bar{R}Q_n$

SR

Q_n	00	01	11	10
0	0	0	d	**0**
1	1	**0**	d	1

$Q_{n+1} = Q_n$

SR

Q_n	00	01	11	10
0	0	0	d	**0**
1	1	**1**	d	1

K-MAP (Figure 5)

AB = 00 AB = 01

AB = 10 AB = 11

$Q_{n+1} = S + \bar{R}Q_n$....(2)

SR

Q_n	00	01	11	10
0	0	0	D	**1**
1	1	**0**	d	1

$Q_{n+1} = S + Q_n$

SR

Q_n	00	01	11	10
0	0	0	d	**1**
1	1	**1**	d	1

The required modification is only AB = 01 for an SR-Flip Flop that is designed with positive logic (NOR Gates Configuration). That is, all the Active states are required to follow 'S'

Similarly, the modification required for Figure 4.1 is chosen from the four possible options is presented in Table 9

Table 9: Four Possible Options to Modify Figure 4.1

$Q_{n+1} = RQ_n$	K-MAP (Figure 1.4)	

ISSN: 2277-9655
Impact Factor: 1.852

		$Q_{n+1} = \bar{S} + RQ_n$
SR		**SR**
Q_n 00 01 11 10	AB = 00 AB = 10	**Q_n** 00 01 11 10
0 d **0** 0 0		0 d **1** 0 0
1 d 1 1 **0**		1 d 1 1 **0**
$Q_{n+1} = Q_n$		$Q_{n+1} = \bar{S} + Q_n$
SR		**SR**
Q_n 00 01 11 10	AB = 01 AB = 11	**Q_n** 00 01 11 10
0 d **0** 0 0		0 d **1** 0 0
1 d 1 1 **1**		1 d 1 1 **1**

The required modification is only AB = 10 for an SR-Flip Flop that is designed with negative logic (NAND Gates Configuration). That is, all the Active states are required to follow 'R'. In addition, the Resting and Forbidden states are to be exchanged. That is, SRQ_n (Forbidden states) = 000, 001 = d, and SRQ_n (Resting States) = 110 ≡ 0, 111 ≡ 1.

Alternative Solution

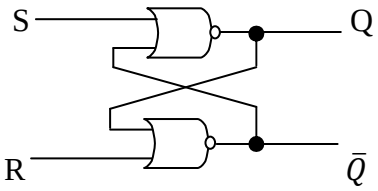

Figure 4.2

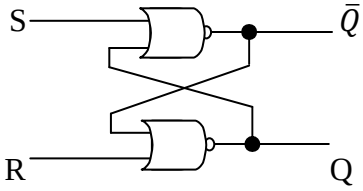

Figure 5.2

From Figure 4.2,

$$Q_{n+1} = \overline{S + \bar{Q}_n} \ldots\ldots\ldots\ldots\ldots\ldots\ldots (a)$$

$$\bar{Q}_{n+1} = \overline{R + Q_n} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (b)$$

If $Q_{n+1} = Q_n$, substituting equation (a) into equation (b) and vice-visa, we have

$$Q_{n+1}(1) = \overline{S + \overline{R + Q_n}} = \overline{S + \bar{R}\bar{Q}_n} \equiv \langle 1 \rangle \ldots\ldots\ldots\ldots\ldots\ldots\ldots (a1)$$

$$\bar{Q}_{n+1}(0) = \overline{R + \overline{S + \bar{Q}_n}} = \overline{R + \bar{S}Q_n} \equiv \langle 0 \rangle \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (b1)$$

From SR-Flip Flop (NOR gate configuration), SRQ_n = 110,111 ≡ $\langle d \rangle$ …… (c)

Plot equations (a1), (b1) & (c) into a K-Map and Truth Table (Table 10) as follows:
Examples: For S = 0, R = 1, Q_n = 0, we have from equation (a1), $Q_{n+1} = \bar{0} = 1$ and from equation (b1), $\bar{Q}_{n+1} = \bar{R} = \bar{1} = 0$. This procedure is continued for SRQ_n = 011, 100 & 101

Table 10: Truth Table for Figure 4.2 Using Equations (a) through (c) Above					
S	**R**	**Q_n**	**Q_{n+1}**	**$\bar{Q}_{n+1}$**	**REMARKS**
0	0	0	0	1	Resting State
0	0	1	1	0	
0	1	0	1	0	The output follows 'R'. That is, Q_{n+1} = R
0	1	1	1	0	The output follows 'R'. That is, Q_{n+1} = R
1	0	0	0	1	The output follows 'R'. That is, Q_{n+1} = R

K-MAP

SR

Q_n	**00**	**01**	**11**	**10**
0	0	1	d	0
1	1	1	d	0

$Q_{n+1} = R + \bar{S}Q_n$

ISSN: 2277-9655
Impact Factor: 1.852

1	0	1	0	1	The output follows 'R'. That is, Q_{n+1} = R	
1	1	0	d	D	Forbidden State	
1	1	1	d	D		

NOTES:

1. The Resting states are satisfied as an SR-Flip Flop
2. The Active states partly follow 'R' as required for an SR-Flip Flop designed with –ve logic (NAND Gate Configuration).
3. The forbidden states do not agree with NAND Gate Configuration.

Hence, this configuration (Fig 4.2) does not fully satisfy the characteristics of an SR-Flip Flop because its Active states follow 'R' instead of 'S' and its forbidden states are wrongly assigned.

Note that this analysis is only possible if it can be ascertained that at all times, the signals arriving at terminals 'S' and 'R' are received at different time and that the propagation times (τ) of both NOR gates are the same. That is, $Q_n = Q_{n+1}$ and $\tau_S = \tau_R$.

From Figure 5.2,

$$Q_{n+1} = \overline{R + \bar{Q}_n} \ldots\ldots\ldots\ldots\ldots\ldots\ldots (a)$$

$$\bar{Q}_{n+1} = \overline{S + Q_n} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (b)$$

If $Q_{n+1} = Q_n$, substituting equation (a) into equation (b) and vice-visa, we have

$$Q_{n+1}(1) = \overline{R + \overline{S + Q_n}} = \overline{R + \bar{S}\bar{Q}_n} \equiv \langle 1 \rangle \ldots\ldots\ldots\ldots\ldots\ldots\ldots (a1)$$

$$\bar{Q}_{n+1}(0) = \overline{S + \overline{R + \bar{Q}_n}} = \overline{S + \bar{R}Q_n} \equiv \langle 0 \rangle \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (b1)$$

From SR-Flip Flop (+ve Logic configuration), $SRQ_n = 110,111 \equiv \langle d \rangle$ …… (c)

Plot equations (a1), (b1) & (c) into a K-Map and Truth Table (Table 11) as follows:
Examples: For S = 0, R = 1, Q_n = 0, we have from equation (a1), $Q_{n+1} = \bar{1} = 0$ and from equation (b1), $\bar{Q}_{n+1} = \bar{0} = 1$. This procedure is continued for SRQ_n = 011, 100 & 101

Table 11: Truth Table for Figure 5.2 Using Equations (a) through (c) Above

S	R	Q_n	Q_{n+1}	$\bar{Q}_{n+1}$	REMARKS
0	0	0	0	0	Resting State
0	0	1	1	0	
0	1	0	0	1	The output follows 'S'. That is, Q_{n+1} = S
0	1	1	0	1	The output follows 'S'. That is, Q_{n+1} = S
1	0	0	1	0	The output follows 'S'. That is, Q_{n+1} = S
1	0	1	1	0	The output follows 'S'. That is, Q_{n+1} = S
1	1	0	d	d	Forbidden State
1	1	1	d	d	

K-MAP

Q_n \ SR	00	01	11	10
0	0	0	d	1
1	1	0	d	1

$Q_{n+1} = S + \bar{R}Q_n$

NOTES:

1. The Resting states are satisfied as an SR-Flip Flop
2. The Active states follow 'S' as required for an SR-Flip Flop designed with +ve logic (NOR Gate Configuration).
3. The Forbidden states are properly assigned.

Hence, this configuration (Fig 5.2) does fully satisfy the characteristics of an SR-Flip Flop because the three required conditions stated above are fully met.

NOTE: This is only possible if it can be ascertained that at all times, the signals arriving at terminals 'S' and 'R' are received at different time and that the propagation times (τ) of both NOR gates are the same. That is, $Q_n = Q_{n+1}$ and $\tau_S = \tau_R$.

2.3.2 SR-Flip Flop (NAND Gate Configuration)

This SR-Flip Flop will be examined by trying to answer the following question:

ISSN: 2277-9655
Impact Factor: 1.852

Given the two configurations shown in Figures 6.2 & 7.2, identify the configuration that is probable to be used as an SR-Flip Flip stating the conditions considered for your choice. Can this choice of yours be used to satisfy fully the characteristics of a conventionally accepted SR-Flip Flop? If not, what modifications are necessary to meet the design requirements? If yes, how does it achieve the required characteristics?

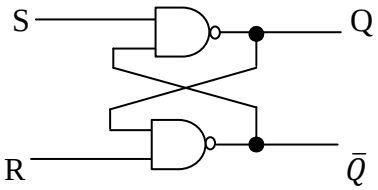

Figure 6.2

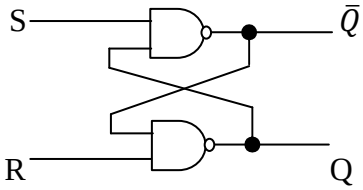

Figure 7.2

From Figure 7.2,

$$Q_{n+1}(1) = \overline{S\bar{Q}_n} = \overline{1-0} = 011,001 \equiv \langle 1 \rangle \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (a)$$
$$\bar{Q}_{n+1}(0) = \overline{RQ_n} = \overline{-11} = 100{,}000 \equiv \langle 0 \rangle \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (b)$$

From SR-Flip Flop (+ve Logic configuration), $SRQ_n = 000{,}001 \equiv \langle d \rangle$ …….. (c)

Plot equations (a), (b) & (c) into a K-Map and Truth Table as follows (see Table 12):

Table 12: Truth Table for Figure 7.2 Using Equations (a) through (c) Above

S	R	Q_n	Q_{n+1}		REMARKS
0	0	0	0,d	The Value of Q_{n+1} is determined from equations (b) & (c)	Forbidden State, d. For this state, X is preferred.
0	0	1	1,d	The Value of Q_{n+1} is determined from equations (a) & (c)	Forbidden State, d. For this state, X is preferred.
0	1	0	A	**This value of Q_{n+1} is indeterminate**	A = 0 or 1
0	1	1	1	The Value of Q_{n+1} is determined from equation (a)	The output follows 'R' That is, $Q_{n+1} = R$
1	0	0	0	The Value of Q_{n+1} is determined from equation (b)	The output follows 'R' That is, $Q_{n+1} = R$
1	0	1	B	**This value of Q_{n+1} is indeterminate**	(B = 0 or 1)
1	1	0	C	**This value of Q_{n+1} is indeterminate**	Resting State, 0 (C = 0 or 1)
1	1	1	D	**This value of Q_{n+1} is indeterminate**	Resting State, 1(D = 0 or 1)

K-MAP

SR

Q_n	00	01	11	10
0	0,d	A	C	0
1	1, d	1	D	B

$Q_{n+1} = \bar{S}Q_n = \overline{S + \bar{Q}_n}$

NOTES:

1. The Resting & Forbidden states do not satisfy a NAND-gate SR-Flip Flop
2. Only two Active states follows 'R' as expected of a NAND-gate SR-FF. The other two Active states are indeterminate.
3. From K-Map, the equation for the Flip Flop output does not agree with that of a conventionally acceptable SR-Flip Flop.

Hence, this configuration (Figure 6.2) does not satisfy the required conditions for a NAND-gate SR-Flip Flop.

From Figure 7.2,

$$Q_{n+1}(1) = \overline{R\bar{Q}_n} = \overline{-10} = 101,001 \equiv \langle 1 \rangle \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (a)$$
$$\bar{Q}_{n+1}(0) = \overline{SQ_n} = \overline{1-1} = 010,000 \equiv \langle 0 \rangle \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (b)$$

From SR-Flip Flop (+ve Logic configuration), $SRQ_n = 000{,}001 \equiv \langle d \rangle$ …… (c)

ISSN: 2277-9655
Impact Factor: 1.852

Plot equations (a), (b) & (c) into a K-Map and Truth Table as follows (see Table 13):

Table 13: Truth Table for Figure 7.2 Using Equations (a) through (c) Above

S	R	Q_n	Q_{n+1}		REMARKS
0	0	0	0,d	The Value of Q_{n+1} is determined from equation (b)	Forbidden State, d. For this state, X is preferred.
0	0	1	1,d	The Value of Q_{n+1} is determined from equation (a)	Forbidden State, d. For this state, X is preferred.
0	1	0	0	The Value of Q_{n+1} is determined from equation (b)	The output follows 'S'
0	1	1	A	**This value of Q_{n+1} is indeterminate**	A = 0 or 1
1	0	0	B	**This value of Q_{n+1} is indeterminate**	B = 0 or 1
1	0	1	1	The Value of Q_{n+1} is determined from equation (a)	The output follows 'S'
1	1	0	C	**This value of Q_{n+1} is indeterminate**	Resting State, 0 (C = 0 or 1)
1	1	1	D	**This value of Q_{n+1} is indeterminate**	Resting State, (D = 0 or 1)

K-MAP

Q_n	00	01	11	10
0	0,d	0	C	B
1	1,d	A	D	1

$Q_{n+1} = \bar{R}Q_n = \overline{R + \bar{Q}_n}$

NOTES:

1. The Resting & Forbidden states failed to satisfy any SR-Flip Flop
2. Only two Active states follows 'S' as required for a NOR-gate SR-Flip Flop.
3. From K-Map, the equation for the Flip Flop output does not agree with that of a conventionally acceptable SR-Flip Flop.

Comparing the analysis of both Figure 6.2 & Figure 7.2, Figure 6.2 is closer to a NAND-gate SR-FF because its two Active states that are determinable follow 'R' as required for a NAND-gate SR-FF.

The modification required for Figure 6.2 is chosen from the sixteen possible options presented in Table 15 or alternatively, it can be determined from the known Truth Table of NAND-gate SR-FF. That is, determine the values of A, B, C & D as shown in Table 14 (ABCD = $1001_2 = 9_{10}$).

Table 14: Truth Table for Figure 6.2

S	R	Q_n	Q_{n+1}		REMARKS
0	0	0	d	The Value of Q_{n+1} is determined from equations (c)	Forbidden State, X. For this state, X is preferred.
0	0	1	d	The Value of Q_{n+1} is determined from equations (c)	Forbidden State, X. For this state, X is preferred.
0	1	0	1		A = 1
0	1	1	1	The Value of Q_{n+1} is determined from equation (a)	The output follows 'R' That is, Q_{n+1} = R
1	0	0	0	The Value of Q_{n+1} is determined from equation (b)	The output follows 'R' That is, Q_{n+1} = R
1	0	1	0		(B = 0)
1	1	0	0		Resting State, 0 (C = 0)
1	1	1	1		Resting State, 1 (D = 1)

K-MAP

SR

Q_n	00	01	11	10
0	d	1	0	0
1	d	1	1	0

$Q_{n+1} = \bar{S} + RQ_n$

Table 15: Sixteen Possible Options to Modify Figure 6.2

Q_{n+1} = 0 …(0)					K-MAP		$Q_{n+1} = SR\bar{Q}_n$ + ..(1)				
	00	01	11	10	ABCD = 0000 (0)	ABCD = 0001 (1)		00	01	11	10

ISSN: 2277-9655
Impact Factor: 1.852

Q_n				
0	d	0	0	0
1	d	0	0	0

Q_n				
0	D	0	0	0
1	D	0	1	0

$Q_{n+1} = SR\bar{Q}_n$.... (2)

Q_n	00	01	11	10
0	d	0	1	0
1	d	0	0	0

ABCD = 0010 (2)

ABCD = 0011 (3)

$Q_{n+1} = SR$ ....(3)

Q_n	00	01	11	10
0	D	0	1	0
1	D	0	1	0

$Q_{n+1} = \bar{R}$ ....(4)

Q_n	00	01	11	10
0	d	0	0	1
1	d	0	0	1

K-MAP

ABCD = 0100 (4)

ABCD = 0101 (5)

$Q_{n+1} = SQ_n + \bar{R}$....(5)

Q_n	00	01	11	10
0	d	0	0	1
1	d	0	1	1

$Q_{n+1} = \bar{R} + S\bar{Q}_n$..(6)

Q_n	00	01	11	10
0	d	0	1	1
1	d	0	0	1

ABCD = 0110 (6)

ABCD = 0111 (7)

$Q_{n+1} = S + \bar{R}$ + ...(7)

Q_n	00	01	11	10
0	d	0	1	1
1	d	0	1	1

$Q_{n+1} = \bar{S}$ ....(8)

Q_n	00	01	11	10
0	d	1	0	0
1	d	1	0	0

K-MAP

ABCD = 1000 (8)

ABCD = 1001 (9)

Equation (9) is the only one that agrees with the equation of a NAND-gate SR-FF. Hence, the modifications required is given by ABCD = 1001.

$Q_{n+1} = \bar{S} + RQ_n$ ....(9)

Q_n	00	01	11	10
0	d	1	0	0
1	d	1	1	0

$Q_{n+1} = \bar{S} + R\bar{Q}_n$ ...(10)

Q_n	00	01	11	10
0	d	1	1	0
1	d	1	0	0

ABCD = 1010 (10)

ABCD = 1011 (11)

$Q_{n+1} = \bar{S} + R$ ....(11)

Q_n	00	01	11	10
0	d	1	1	0
1	D	1	1	0

$Q_{n+1} = \bar{S} + \bar{R}$ ...(12)

Q_n	00	01	11	10
0	X	1	0	1
1	X	1	0	1

K-MAP

ABCD = 1100 (12)

ABCD = 1101 (13)

$Q_{n+1} = Q_n + \bar{S} + \bar{R}$ ...(13)

Q_n	00	01	11	10
0	X	1	0	1
1	X	1	1	1

$Q_{n+1} = \bar{R} + \bar{S} + \bar{Q}_n$ ...(14)

Q_n	00	01	11	10
0	X	1	1	1
1	X	1	0	1

ABCD = 1110 (14)

ABCD = 1111 (15)

$Q_{n+1} = 1$ ...(15)

Q_n	00	01	11	10
0	X	1	1	1
1	X	1	1	1

The required modifications of Figure 7.2 are too many. Hence, it is not engineering-wise to modify Figure 7.2 since the modification of Fig 6.2 gives a perfect design.

Alternative Solution

ISSN: 2277-9655
Impact Factor: 1.852

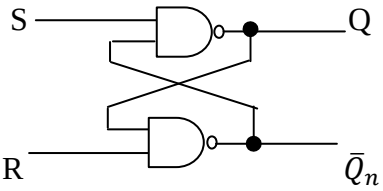

Figure 6.3

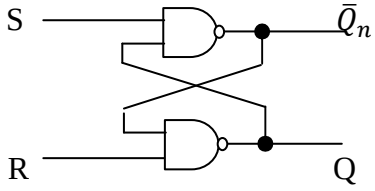

Figure 7.3

From Figure 6.3,

$$Q_{n+1} = \overline{S\bar{Q}_n} \ldots\ldots\ldots\ldots\ldots\ldots\ldots (a)$$
$$\bar{Q}_{n+1} = \overline{RQ_n} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (b)$$

If $Q_{n+1} = Q_n$, substituting equation (a) into equation (b) and vice-visa, we have

$$Q_{n+1}(1) = \overline{S\overline{RQ_n}} \equiv \langle 1\rangle \ldots\ldots\ldots\ldots\ldots\ldots\ldots (a1)$$
$$\bar{Q}_{n+1}(0) = \overline{R\overline{S\bar{Q}_n}} \equiv \langle 0\rangle \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (b1)$$

From SR-Flip Flop (-ve Logic configuration), $SRQ_n = 00,001 \equiv \langle d\rangle$ …… (c)

Plot equations (a), (b1) & (c) into a K-Map and Truth Table (Table 16) as follows:
Examples: For S = 0, R = 1, $Q_n = 0$, we have from equation (a1), $Q_{n+1} = \bar{S} = \bar{0} = 1$ and from equation (b1), $\bar{Q}_{n+1} = \bar{R} = \bar{1} = 0$. This procedure is continued for SRQ_n = 011, 100 & 101

Table 16: Truth Table for Figure 6.3 Using Equations (a) through (c) Above

S	R	Q_n	Q_{n+1}	$\bar{Q}_{n+1}$	REMARKS
0	0	0	d	D	Forbidden States
0	0	1	d	D	
0	1	0	1	0	The output follows 'R'. That is, $Q_{n+1} = R$
0	1	1	1	0	The output follows 'R'. That is, $Q_{n+1} = R$
1	0	0	0	1	The output follows 'R'. That is, $Q_{n+1} = R$
1	0	1	0	1	The output follows 'R'. That is, $Q_{n+1} = R$
1	1	0	0	1	Resting States
1	1	1	1	0	

K-MAP

Q_n	00	01	11	10
0	d	1	0	0
1	d	1	1	0

$Q_{n+1} = \bar{S} + RQ_n$
This is the same as equation (9) earlier obtained.

NOTES:

1. The Resting and Forbidden states satisfy the conditions required for a NAND-gate SR-FF
2. The Active states partly follow 'R' as required for a NAND-gate SR-Flip Flop.

Hence, this configuration (Fig 6.3) fully satisfy the characteristics of a NAND-gate SR-Flip Flop because its Active states follow 'R' instead of 'S' and its forbidden states are correctly assigned.
This analysis is only possible if it can be ascertained that, at all times, the signals arriving at terminals 'S' and 'R' are received at different times and that the propagation times (τ) of both NAND gates are the same. That is, $Q_n = Q_{n+1}$ and $\tau_S = \tau_R$.

From Figure 7.3,

$$Q_{n+1} = \overline{R\bar{Q}_n} \ldots\ldots\ldots\ldots\ldots\ldots\ldots (a)$$
$$\bar{Q}_{n+1} = \overline{SQ_n} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (b)$$

If $Q_{n+1} = Q_n$, substituting equation (a) into equation (b) and vice-visa, we have

$$Q_{n+1}(1) = \overline{R\overline{SQ_n}} \equiv \langle 1\rangle \ldots\ldots\ldots\ldots\ldots\ldots\ldots (a1)$$
$$\bar{Q}_{n+1}(0) = \overline{S\overline{R\bar{Q}_n}} \equiv \langle 0\rangle \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (b1)$$

From SR-Flip Flop (-ve Logic configuration), SRQ_n = 000, 001 $\equiv \langle d\rangle$ …… (c)

ISSN: 2277-9655
Impact Factor: 1.852

Plot equations (a1), (b1) & (c) into a K-Map and Truth Table (Table 17) as follows: Examples: For S = 0, R = 1, Q_n = 0, we have from equation (a1), $Q_{n+1} = \bar{R} = \bar{1} = 0$ and from equation (b1), $\bar{Q}_{n+1} = \bar{0} = 1$. This procedure is continued for SRQ_n = 011, 100 & 101

Table 17: Truth Table for Figure 7.3 Using Equations (a) through (c) Above

S	R	Q_n	Q_{n+1}	$\bar{Q}_{n+1}$	REMARKS
0	0	0	d	d	Forbidden States
0	0	1	d	d	
0	1	0	0	1	The output follows 'S'. That is, Q_{n+1} = S
0	1	1	0	1	The output follows 'S'. That is, Q_{n+1} = S
1	0	0	1	0	The output follows 'S'. That is, Q_{n+1} = S
1	0	1	1	0	The output follows 'S'. That is, Q_{n+1} = S
1	1	0	0	1	Resting States
1	1	1	1	0	

K-MAP

SR / Q_n	00	01	11	10
0	d	0	0	1
1	d	0	1	1

$Q_{n+1} = \bar{R} + SQ_n$

NOTES:

1. The Resting & Forbidden states are satisfied as a NAND-gate SR-Flip Flop
2. The Active states follow 'S' as if it were for a NOR-gate SR-Flip Flop. Hence, Fig 2 fails to meet the requirements of a NAND-gate SR-FF.

3 Conclusion

From the analysis presented in this paper, it is proved beyond any doubt that Figures 2 & 3 are the valid logic circuits for SR-Flip Flops while any other that contains only two gates will not function fully as an SR-Flip Flop. There are four Active/transition states in any SR-Flip Flop which suggests, as a rule of thumb, the minimum number of gates that may be required to meet this requirement.

These Active states are identified and analysed from the SR-Flip Flop Truth Table shown in Table 18 as follows:

- Active state $2_{10} = 010_2$ where $Q_n \rightarrow Q_{n+1}$. That is, 0 → 0 (No transition).
- Active state $3_{10} = 011_2$ where $Q_n \rightarrow \bar{Q}_{n+1}$. That is, 1 → 0 (There is a transition).
- Active state $4_{10} = 010_2$ where $\bar{Q}_n \rightarrow Q_{n+1}$. That is, 0 → 1 (There is a transition).
- Active state $5_{10} = 010_2$ where $Q_n \rightarrow Q_{n+1}$. That is, 1 → 1 (No transition).

From the above list of possible transition states, it appears as if there is no transition taken place in the Active states 2_{10} & 5_{10}. This may be what is responsible for the two missing gates in Figures 4.1, 5.1, 4.2, 5.2, 6.2, 7.2, 6.3 & 7.3. However, a close examination of the transitions presented in Table 18, reveals as follows:

Table 18: Truth Table of SR-Flip Flop

S/N	S	R	Q_n	Q_{n+1}		Transition State
0	0	0	0	0		Resting
1	0	0	1	1		Resting
2	0	1	0	0	1 diagonal transition	Active
3	0	1	1	0	1 diagonal transition, 1 horizontal transition	Active
4	1	0	0	1	1 horizontal transition	Active
5	1	0	1	1	No transition	Active
6	1	1	0	d		Forbidden
7	1	1	1	d		Forbidden

ISSN: 2277-9655
Impact Factor: 1.852

- Active state $2_{10} = 010_2$ where $Q_n \rightarrow Q_{n+1}$.
  That is, $0 \rightarrow 0$ (No horizontal transition but there is one diagonal transition, $1 \rightarrow 0$).
- Active state $3_{10} = 011_2$ where $Q_n \rightarrow \bar{Q}_{n+1}$.
  That is, $1 \rightarrow 0$ (There is a horizontal and a diagonal transition, $0 \rightarrow 1$).
- Active state $4_{10} = 010_2$ where $\bar{Q}_n \rightarrow Q_{n+1}$.
  That is, $0 \rightarrow 1$ (There is a horizontal transition).
- Active state $5_{10} = 010_2$ where $Q_n \rightarrow Q_{n+1}$. That is, $1 \rightarrow 1$ (No transition).

This make a total four transitions as expected and represented by the four-gate logic circuits of Figures 2 & 3. This analysis further supports the position of this paper.

References

[1] O. J. Omotosho, (2012), "fundamentals of Digital Systems", Franco-Ola publishers
[2] S. O. Ogunlere, (2011), "Digital Logic and Design", France-Ola publishers
[3] Ralph J. Smith; (1984), "Circuits, Devices and Systems", 4th Edition
[4] W. H. Eccles and F. W. Jordan, (19 September 1919), "A trigger relay utilizing three-electrode thermionic vacuum tubes," *The Electrician*, vol. 83, page 298. Reprinted in: *Radio Review*, vol. 1, no. 3, pages 143–146 (December 1919).
[5] Montgomery Phister (1958), *"Logical Design of Digital Computers"*. http://books.google.com/books?
[6] K. G. Sharma, T. Sharma, B. P. Singh and M. Sharma in Dept of Electronics and Communication Engg. FET-MITS Lakshmangarh, Rajasthan, (INDIA).
[7] P. K. Meher, , "Extended Sequential Logic for Synchronous Circuit Optimization and its Applications" IEEE journal