ISSN: 2277-9655
Impact Factor: 1.852

IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

Analysis and Design of Different Flip Flops, Extensions of Conventional JK-Flip Flops

Prof. Olawale J. Omotosho[*1], Engr. Samson O. Ogunlere[2]
[*1,2]Babcock University, Computer Science Department, Ilishan-Remo, Ogun State, Nigeria
ojomotosho1@yahoo.com

Abstract

The analysis and design of a 100% and 87.5% high-performance and efficient memory element (Flip-Flop) capable of being selected for the purpose of reading from and writing into it, is of crucial importance in modern digital applications such as the Very large Scale integrated circuits (VLSI). The optimization of existing structures is necessary when the requirement of the flip-flops is for low-power, high-speed or low-noise applications. In this paper, the optimization of the existing flip-flops (SR and JK) is investigated to ascertain their utilization rate. Detailed analysis of a proposed (XY flip-flop as extensions of conventional JK-Flip Flops) structure is carried out to prove whether 100% and 87.5% utilization can be achieved as against the existing, most widely used JK-Flip Flops that has 75% utilization rate. This paper also considered the use of the proposed (extensions of conventional JK–Flip-flops) in other digital device applications such as sequence detectors.

Keywords: Sequential logic (Bistable Multivibrator), Flip-Flops, Logic optimization (K-Map), Sequence Detector, Memory Element, Extension of Conventional JK-Flip Flops

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. This is examined in details in section 2 of this paper.

SR-Flip Flops

SR-Flip Flops have four binary combinations. Because of the sequential nature of Flip Flops (feedback), for each combination, there are eight transition states as shown in Table 1.

Table 1: Truth Table of SR-Flip Flop

NOR Configuration						NAND Configuration				
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:										

ISSN: 2277-9655
Impact Factor: 1.852

Q_n = previous output, Q_{n+1} = present output, D = I don't care term (0,1) For active transition, Q_{n+1} = S For active transition, Q_{n+1} = R

Design of Different Alternatives to SR-Flip Flops

When the forbidden states of an SR-Flip Flop are converted to toggling states, a JK-Flip Flop is so designed. Hence, such a JK-Flip Flop retains the rest features of an SR-Flip Flop, such as its resting and active states. Thus making a JK-Flip Flop to attain 75% utilization as against 50% utilization of an SR-Flip Flop. The remaining 25% utilization has been examined and this is being exploited in this paper.

Application of Alternatives Flip Flops to a Conventional JK-Flip Flop

The Truth Table of three other possibilities including the conventional JK-Flip Flop are presented in Tables 2 & 3. For the purpose of this paper, these different JK-Flip Flops is tagged as follows:

- **JK – 000, 001 Rest (75%):** JK-Flip Flop which has two Resting states and the Resting states are identified as JKQ = 000, 001. This is the referred to as the conventional JK-Flip Flop which is in the market today.
- **JK – 000 Rest (87.5%):** JK-Flip Flop which has only one Resting state instead of the usual two and the Resting state is identified as JKQ = 000.
- **JK – 001 Rest (87.5%):** JK-Flip Flop which has only one Resting state instead of the usual two and the Resting state is identified as JKQ = 001.
- **JK – No Rest (100%):** JK-Flip Flop which has no Resting state instead of the usual two.

Table 2: Truth Table of Different JK-Flip Flops

OPTION 1										OPTION 4			
JK-000					JK-000 (87.5%)					JK-001 (87.5%)			
Q_n	00	01	11	10	J	K	Q_n	Q_{n+1}		J	K	Q_n	Q_{n+1}
0	0	0	1	1	0	0	0	0		0	0	0	1
1	0	0	0	1	0	0	1	0		0	0	1	1
					0	1	0	0		0	1	0	0
Q_n	00	01	11	10	0	1	1	0		0	1	1	0
0	1	0	1	1	1	0	0	1		1	0	0	1
1	1	0	0	1	1	0	1	1		1	0	1	1
JK-001					1	1	0	1		1	1	0	1
					1	1	1	0		1	1	1	0

Table 3: Truth Table of Different JK-Flip Flops

OPTION 3										OPTION 2			
	XY-No Rest				XY – 100%					JK-000, 001 (75%)			
Q_n	00	01	11	10	J	K	Q_n	Q_{n+1}		J	K	Q_n	Q_{n+1}
0	1	0	1	1	0	0	0	1		0	0	0	0
1	0	0	0	1	0	0	1	0		0	0	1	1
					0	1	0	0		0	1	0	0
Q_n	00	01	11	10	0	1	1	0		0	1	1	0
0	0	0	1	1	1	0	0	1		1	0	0	1
1	1	0	0	1	1	0	1	1		1	0	1	1
JK-000, 001					1	1	0	1		1	1	0	1
					1	1	1	0		1	1	1	0

In Tables 2 & 3, the corresponding K-Maps for the different types of Flip Flops are also shown from where the logic equations are obtained as follows:

ISSN: 2277-9655
Impact Factor: 1.852

OPTION 1: JK-000 Rest (87.5%)

$$Q_{n+1} = J\bar{K} + J\bar{Q}_n = J(\bar{K} + \bar{Q}_n) \ldots\ldots\ldots\ldots (1)$$

NAND (1,2,3)

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

NOR

$$Q_{n+1} = \overline{J(\bar{K} + \bar{Q}_n)} = \overline{\bar{J} + \overline{(\bar{K} + \bar{Q}_n)}} \ldots\ldots\ldots\ldots (1b)$$

$$\bar{Q}_{n+1} = \bar{J} + KQ_n \ldots\ldots\ldots\ldots (2)$$

NAND (4,5)

$$\bar{Q}_{n+1} = \overline{\overline{\bar{J} + KQ_n}} = \overline{J.\overline{KQ_n}} \ldots\ldots\ldots\ldots (2a)$$

NOR

$$\overline{KQ_n} = \overline{\bar{K} + \bar{Q}_n} \ldots\ldots\ldots\ldots (2b)$$

Substitute equation (2b) into equation (2), we have

$$\bar{Q}_{n+1} = \bar{J} + \overline{\bar{K} + \bar{Q}_n} \ldots\ldots\ldots\ldots (2c)$$

$$Q_{n+1} = \overline{\bar{J} + \bar{K} + \bar{Q}_n} \ldots\ldots\ldots\ldots (2e)$$

Combining equations (1a) & (2a) to produce NAND gate configuration
Combining equations (1b) & (2e) to produce NOR gate configuration but equation (1b) = equation (2e). Therefore, only equations (1a) & 2(a) can be combined to obtain Figure 1

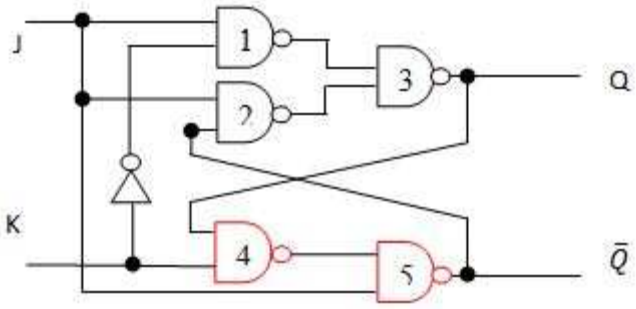


Figure 1: Logic Circuit of JK-000 Rest (87.5%) Flip Flop (NAND Gate Configuration)

OPTION 4: JK-001 Rest (87.5%)

$$\bar{Q}_{n+1} = K\bar{J} + KQ_n = K(\bar{J} + Q_n) \ldots\ldots\ldots\ldots (1)$$

NAND (1,2,3)

$$\bar{Q}_{n+1} = \overline{K\bar{J} + KQ_n} = \overline{K\bar{J}.\overline{KQ_n}} \ldots\ldots\ldots\ldots (1a)$$

NOR

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

$$Q_{n+1} = \bar{K} + J\bar{Q}_n \ldots\ldots\ldots\ldots (2)$$

NAND (4,5)

$$Q_{n+1} = \overline{\bar{K} + J\bar{Q}_n} = \overline{K.\overline{J\bar{Q}_n}} \ldots\ldots\ldots\ldots (2a)$$

ISSN: 2277-9655
Impact Factor: 1.852

NOR

$$\overline{J\bar{Q}_n} = \overline{\bar{J} + Q_n} \dots\dots\dots\dots (2b)$$

Substitute equation (2b) into equation (2), we have

$$\bar{Q}_{n+1} = \bar{K} + \overline{\bar{J} + Q_n} \dots\dots\dots\dots (2c)$$

$$Q_{n+1} = \overline{\bar{K} + \overline{\bar{J} + Q_n}} \dots\dots\dots\dots (2e)$$

Combining equations (1a) & (2a) to produce NAND gate configuration
Combining equations (1b) & (2e) to produce NOR gate configuration but equation (1b) = equation (2e)
Therefore, only equations (1a) & 2(a) can be combined to obtain Figure 2

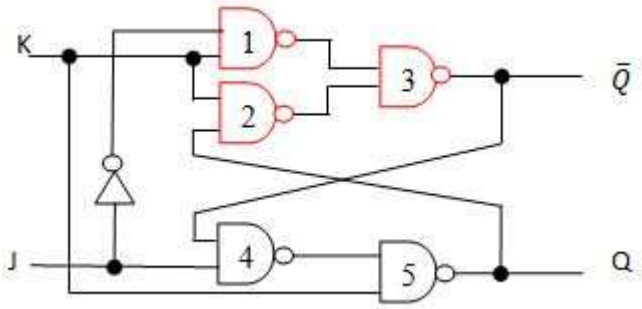


Figure 2: Logic Circuit of JK-001 Rest (87.5%) Flip Flop (NAND Gate Configuration)

OPTION 3: XY-Zero Rest (X=J, Y=K)

$$Q_{n+1} = J\bar{K} + J\bar{Q}_n + \bar{K}\bar{Q}_n = J(\bar{K} + \bar{Q}_n) + \bar{K}\bar{Q}_n \dots\dots\dots\dots (1)$$

NAND (1,2,3,4)

$$Q_{n+1} = \overline{\overline{J\bar{K} + J\bar{Q}_n + \bar{K}\bar{Q}_n}} = \overline{\overline{J\bar{K}}.\overline{J\bar{Q}_n}.\overline{\bar{K}\bar{Q}_n}} \dots\dots\dots\dots (1a)$$

$$Q_{n+1} == \overline{\overline{J\bar{K}}.\overline{J\bar{Q}_n}.\overline{\bar{K}\bar{Q}_n}} \dots\dots\dots\dots (1a)$$

NOR (5,6,7,8)

Replace

$$\overline{\overline{\bar{K}Q_n}} = \overline{K + Q_n} \dots\dots\dots\dots (1b)$$

$$\overline{\overline{J\bar{K}}} = \overline{\bar{J} + K} \dots\dots\dots\dots (1c)$$

$$\overline{\overline{J\bar{Q}_n}} = \overline{\bar{J} + Q_n} \dots\dots\dots\dots (1d)$$

Substitute equations (1b), (1c) & (1d) into equation (1b), we have

$$Q_{n+1} = \overline{\bar{J} + K} + \overline{\bar{J} + Q_n} + \overline{K + Q_n} \dots\dots\dots\dots (1e)$$

$$\bar{Q}_{n+1} = \overline{\overline{\bar{J} + K} + \overline{\bar{J} + Q_n} + \overline{K + Q_n}} \dots\dots\dots\dots (1e)$$

$$\bar{Q}_{n+1} = K\bar{J} + KQ_n + \bar{J}Q_n = K(\bar{J} + Q_n) + \bar{J}Q_n \dots\dots\dots\dots (2)$$

ISSN: 2277-9655
Impact Factor: 1.852

NAND (5,6,7,8)

$$\bar{Q}_{n+1} = \overline{K\bar{J} + KQ_n + \bar{J}Q_n} = \overline{K\bar{J}}.\overline{KQ_n}.\overline{\bar{J}Q_n} \ldots\ldots\ldots\ldots (2a)$$

NOR (1,2,3,4)

Replace

$$\overline{\overline{K\bar{J}}} = \overline{\bar{K} + J} \ldots\ldots\ldots\ldots (2b)$$

$$\overline{\overline{KQ_n}} = \overline{\bar{K} + \bar{Q}_n} \ldots\ldots\ldots\ldots (2c)$$

$$\overline{\overline{\bar{J}Q_n}} = \overline{J + \bar{Q}_n} \ldots\ldots\ldots\ldots (2d)$$

Substitute equations (2b), (2c) & (2d) into equation (2), we have

$$\bar{Q}_{n+1} = \overline{\bar{K} + J} + \overline{\bar{K} + \bar{Q}_n} + \overline{J + \bar{Q}_n} \ldots\ldots\ldots\ldots (2e)$$

$$Q_{n+1} = \overline{\overline{\bar{K} + J} + \overline{\bar{K} + \bar{Q}_n} + \overline{J + \bar{Q}_n}} \ldots\ldots\ldots\ldots (2f)$$

Combining equations (1a) & (2a) to produce NAND gate configuration
Combining equations (1e) & (2f) to produce NOR gate configuration
Therefore, equations (1a) & (2a) can be combined as shown in Figure 3

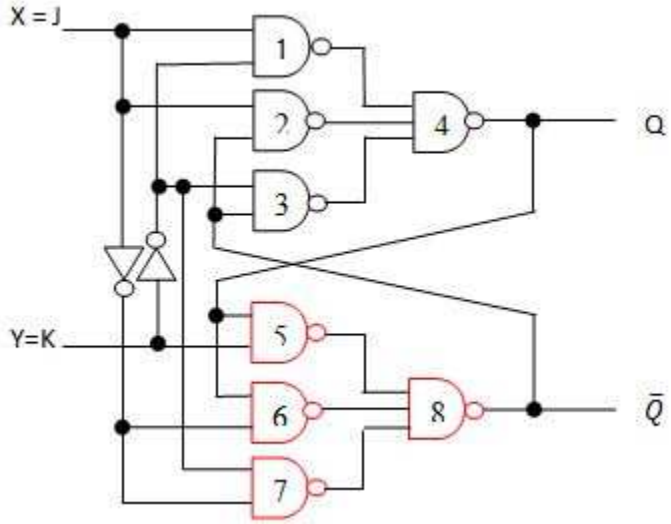


Figure 3: Logic Circuit of JK-No Rest (100%) Flip Flop (NAND Gate Configuration)

Therefore, equations (1e) & (2f) can be combined as presented in Figure 4

ISSN: 2277-9655
Impact Factor: 1.852

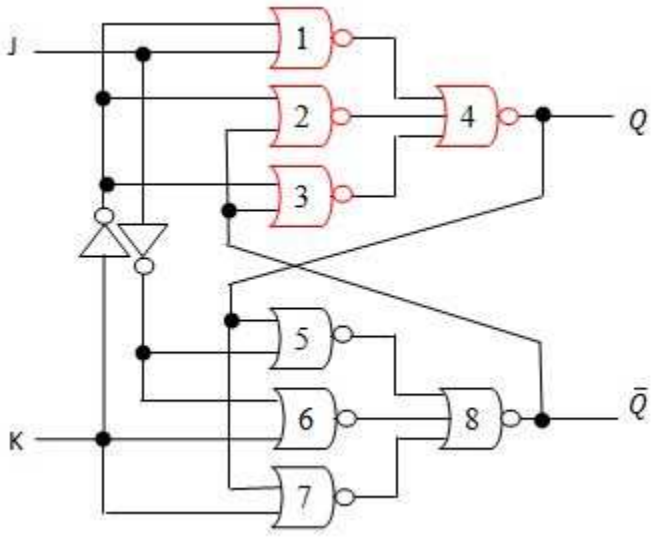


Figure 4: Logic Circuit of JK-No Rest (100%) Flip Flop (NOR Gate Configuration)

OPTION 2: JK-000, 001 Rest: This is the conventional JK Flip Flop

$$Q_{n+1} = J\bar{Q}_n + \bar{K}Q_n \ldots\ldots\ldots (1)$$

NAND (1,2,3)

$$Q_{n+1} = \overline{\overline{J\bar{Q}_n + \bar{K}Q_n}} = \overline{\overline{J\bar{Q}_n}.\overline{\bar{K}Q_n}} \ldots\ldots\ldots (1a)$$

NOR

$$\overline{\overline{J\bar{Q}_n}} = \overline{\bar{J} + Q_n} \ldots\ldots\ldots (1b)$$

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

Substitute equations (1b) & (1c) into equation (1), we have

$$Q_{n+1} = \overline{\bar{J} + Q_n} + \overline{K + \bar{Q}_n} \ldots\ldots\ldots (1d)$$

$$\bar{Q}_{n+1} = \overline{\overline{\bar{J} + Q_n} + \overline{K + \bar{Q}_n}} \ldots\ldots\ldots (1e)$$

$$\bar{Q}_{n+1} = KQ_n + \bar{J}\bar{Q}_n \ldots\ldots\ldots (2)$$

NAND (4,5,6)

$$\bar{Q}_{n+1} = \overline{\overline{KQ_n + \bar{J}\bar{Q}_n}} = \overline{\overline{KQ_n}.\overline{\bar{J}\bar{Q}_n}} \ldots\ldots\ldots (2a)$$

NOR (4,5,6)

$$\overline{\overline{\bar{J}\bar{Q}_n}} = \overline{J + Q_n} \ldots\ldots\ldots (2b)$$

$$\overline{\overline{KQ_n}} = \overline{\bar{K} + \bar{Q}_n} \ldots\ldots\ldots (2c)$$

Substitute equations (2b) & (2c) into equation (2), we have

$$\bar{Q}_{n+1} = \overline{\bar{K} + \bar{Q}_n} + \overline{J + Q_n} \ldots\ldots\ldots (2d)$$

ISSN: 2277-9655
Impact Factor: 1.852

$$Q_{n+1} = \overline{\overline{K} + \overline{Q}_n} + \overline{J + Q_n} \ldots\ldots\ldots\ldots (2e)$$

Combining equations (1a) (2a) to produce NAND gate configuration
Combining equations (1e) (2e) to produce NOR gate configuration

Therefore, equations (1a) & (2a) can be combined as shown in Figure 5

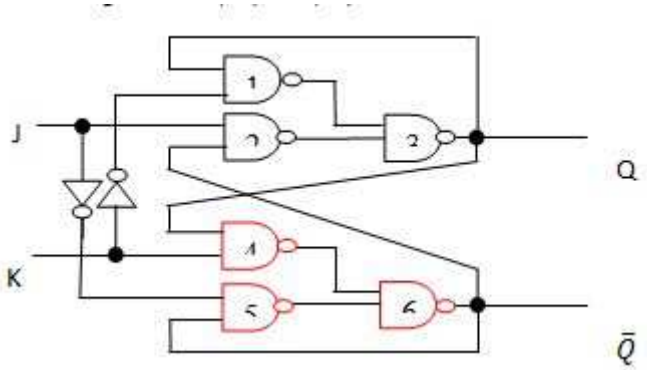
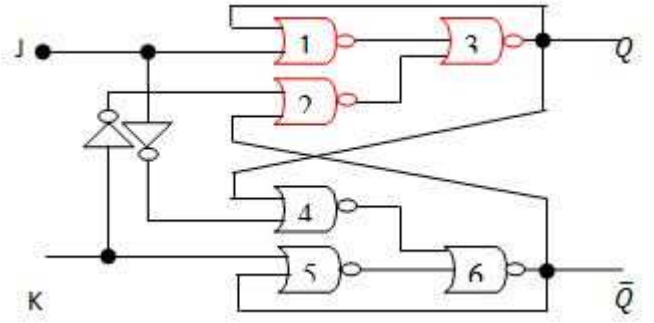


Figure 5: Logic Circuit of Conventional JK-Flip Flop - 75% (NAND Gate Configuration)

Therefore, equations (1e) & (2e) can be combined as presented in Figure 6



The summary of the design analysis is illustrated in Table 4

Table 4: Summary of Analysis of Design

S/N	TYPE OF FLIP FLOP	CONFIGURATION	NUMBER OF ACTIVE TRANSITION/STATE		NUMBER OF GATE
			Diagonal	Horizontal	
1.	**SR-Flip Flop (50%)**	NAND & NOR gates	2	2	4
2.	**JK-000 Rest (87.5%)**	Only NAND gate	3	5	5
3.	**JK-000, 001 Rest (75%)**	NAND & NOR gates	2	4	6
4.	**XY-No Rest (X=J, Y=K) (100%)**	NAND & NOR gates	2	6	8
5.	**JK-001 Rest (87.5%)**	Only NAND gate	3	5	5

NOTES:

1. Inverter gates used for the input variables (J & K) are not counted since the complements of these inputs are available from the input console.
2. The number of active transitions/states is the same with the number of gates required to achieve these transitions for 50%, 75% and 100% utilization.
3. The number of gates is less than the number of transitions/states for the 87.5% utilization. It appears that only the horizontal transitions are catered for by the five gates. This may be probably why they

ISSN: 2277-9655
Impact Factor: 1.852

cannot be configured by with NOR gates.

Application of Alternatives Flip Flops to the Conventional JK-Flip Flop

JK-Flop Flops are used for the following operations in computer/digital systems:

- Storage Devices such as RAM and any other high speed memory-driven system.
- Counters/Sequence Detectors in Decoder and clock-enabled systems.
- Counters as Frequency Dividers/Square Wave Generators
- Shift Registers.
- Data Transfer

This paper will examine how the 87.5% and 100% utilization JK-Flip Flops can be adapted in these areas of application mentioned above.

Storage Devices

All semiconductor memory devices, especially the ones where Flip Flops are employed, a memory element must be constructed from the intended Flip Flop that will have provisions for READ and WRITE commands, SELECT and DATA terminals amongst other requirements. Therefore, a memory element will be designed using the 87.5% and 100% utilization JK-Flip Flops.

Design of a Memory Element

The basic memory cell of a RAM is a Flip-Flop adequately gated. This will be designed as follows:
The following input signals will be required:

SELECT input to select a particular location in memory, designated = S_e
WRITE input command, designated = W
DATA input to be written into, designated = I
READ input command, designated = R_e
DATA output to be read from, designated = O

WRITE Command Consideration:

First, let us consider writing into the memory which requires select and data inputs.
These three inputs with the previous output of the JK-Flip Flop will determine the input combinations as presented in the Table 5.

Table 5a: Truth Table of a Memory Element And an XY-FLIP FLOP

S/N	S_e	I	W	Q_n	Q_{n+1}	$Q_n \rightarrow Q_{n+1}$	X	Y		X	Y	Q_n	Q_{n+1}
0	0	0	0	0	0	**0→0** ⟶	0	1		0	0	0	1
1	0	0	0	1	1	**1→1** ⟶	1	0		0	0	1	0
2	0	0	1	0	0	**0→0** ⟶	0	1		0	1	0	0
3	0	0	1	1	1	**1→1** ⟶	1	0		0	1	1	0
4	0	1	0	0	0	**0→0** ⟶	0	1		1	0	0	1
5	0	1	0	1	1	**1→1** ⟶	1	0		1	0	1	1
6	0	1	1	0	0	**0→0** ⟶	0	1		1	1	0	1
7	0	1	1	1	1	**1→1** ⟶	1	0		1	1	1	0
8	1	0	0	0	0	**0→0** ⟶	0	1		XY- No Rest Flip Flop (100%)			
9	1	0	0	1	1	**1→1** ⟶	1	0					
10	1	0	1	0	0	**0→0** ⟶	0	1					
11	1	0	1	1	0	**1→0** ⟶	0,0,1	0,1,1					
12	1	1	0	0	0	**0→0** ⟶	0	1					
13	1	1	0	1	1	**1→1** ⟶	1	0					
14	1	1	1	0	1	**0→1** ⟶	0,1,1	0,0,1					
15	1	1	1	1	1	**1→1** ⟶	1	0					

ISSN: 2277-9655
Impact Factor: 1.852

Table 5b: Truth Table of a Memory Element And an XY-FLIP FLOP

S/N	S_e	I	W	Q_n	Q_{n+1}	$Q_n \rightarrow Q_{n+1}$	X	Y
0	0	0	0	0	0	**0→0** ⟶	0	1
1	0	0	0	1	1	**1→1** ⟶	1	0
2	0	0	1	0	0	**0→0** ⟶	0	1
3	0	0	1	1	1	**1→1** ⟶	1	0
4	0	1	0	0	0	**0→0** ⟶	0	1
5	0	1	0	1	1	**1→1** ⟶	1	0
6	0	1	1	0	0	**0→0** ⟶	0	1
7	0	1	1	1	1	**1→1** ⟶	1	0
8	1	0	0	0	0	**0→0** ⟶	0	1
9	1	0	0	1	1	**1→1** ⟶	1	0
10	1	0	1	0	0	**0→0** ⟶	0	1
11	1	0	1	1	0	**1→0** ⟶	d	d
12	1	1	0	0	0	**0→0** ⟶	0	1
13	1	1	0	1	1	**1→1** ⟶	1	0
14	1	1	1	0	1	**0→1** ⟶	d	d
15	1	1	1	1	1	**1→1** ⟶	1	0

X	Y	Q_n	Q_{n+1}
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

XY- No Rest Flip Flop (100%)

This Table is the same as Table 5a except that S/N 11, & 14 are replaced with 'don't care terms, d'.

The values of X & Y are plotted into their respective K-Maps as shown in Table 6 from where the corresponding logic equations are derived.

Table 6: K-Maps for X & Y terminals of XY-No Rest Flip Flop

K-Map for X

X= Q_n ..... (1)

WQ_n \ S_eI	00	01	11	10
00	0^0	0^4	0^{12}	0^8
01	1^1	1^5	1^{13}	1^9
11	1^3	1^7	1^{15}	d^{11}
10	0^2	0^6	d^{14}	0^{10}

K-Map for Y

Y = $\bar{Q}_n$..... (2)

WQ_n \ S_eI	00	01	11	10
00	1^0	1^4	1^{12}	1^8
01	0^1	0^5	0^{13}	0^9
11	0^3	0^7	0^{15}	0^{11}
10	1^2	1^6	d^{14}	1^{10}

This configuration cannot be used to design **basic memory element** because the logic equations (1) & (2) are not functions of the required inputs (S_e, I & W) which are to be used to SELECT (S_e) the desired location, to WRITE (W) the required DATA (I) into a storage device. Hence, the configuration cannot be used as a storage device.

Similarly, the same analysis is repeated for JK-000 Rest Flip Flop (87.5%) and presented in Table 7

Table 7a: Truth Table of a Memory Element And a JK-000 Rest FLIP FLOP (87.5%)

S/N	S_e	I	W	Q_n	Q_{n+1}	$Q_n \rightarrow Q_{n+1}$	J	K
0	0	0	0	0	0	**0→0** ⟶	0,0	1,0
1	0	0	0	1	1	**1→1** ⟶	1	0
2	0	0	1	0	0	**0→0** ⟶	0,0	1,0
3	0	0	1	1	1	**1→1** ⟶	1	0
4	0	1	0	0	0	**0→0** ⟶	0,0	1,0
5	0	1	0	1	1	**1→1** ⟶	1	0
6	0	1	1	0	0	**0→0** ⟶	0,0	1,0

J	K	Q_n	Q_{n+1}
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1

ISSN: 2277-9655
Impact Factor: 1.852

7	0	1	1	1	1	**1→1** ⟶	1	0		1	1	1	0
8	1	0	0	0	0	**0→0** ⟶	0,0	1,0		JK- 000 Rest Flip Flop (87.5%)			
9	1	0	0	1	1	**1→1** ⟶	1	0					
10	1	0	1	0	0	**0→0** ⟶	0,0	1,0					
11	1	0	1	1	0	**1→0** ⟶	0,0,1	0,1,1		Resting State = XYQ_n = 000			
12	1	1	0	0	0	**0→0** ⟶	0,0	1,0					
13	1	1	0	1	1	**1→1** ⟶	1	0					
14	1	1	1	0	1	**0→1** ⟶	1,1	0,1					
15	1	1	1	1	1	**1→1** ⟶	1	0					

Table 7b: Truth Table of a Memory Element And a JK-000 Rest FLIP FLOP (87.5%)

S/N	**S_e**	**I**	**W**	**Q_n**	**Q_{n+1}**	**$Q_n \rightarrow Q_{n+1}$**	**J**	K		J	**K**	**Q_n**	**Q_{n+1}**
0	0	0	0	0	0	**0→0** ⟶	0	d		0	0	0	0
1	0	0	0	1	1	**1→1** ⟶	1	0		0	0	1	0
2	0	0	1	0	0	**0→0** ⟶	0	d		0	1	0	0
3	0	0	1	1	1	**1→1** ⟶	1	0		0	1	1	1
4	0	1	0	0	0	**0→0** ⟶	0	d		1	0	0	1
5	0	1	0	1	1	**1→1** ⟶	1	0		1	0	1	1
6	0	1	1	0	0	**0→0** ⟶	0	d		1	1	0	$_1$
7	0	1	1	1	1	**1→1** ⟶	1	0		1	1	1	0
8	1	0	0	0	0	**0→0** ⟶	0	d		JK- 000 Rest Flip Flop (87.5%)			
9	1	0	0	1	1	**1→1** ⟶	1	0					
10	1	0	1	0	0	**0→0** ⟶	0	d					
11	1	0	1	1	0	**1→0** ⟶	d	d		Resting State = XYQ_n = 000			
12	1	1	0	0	0	**0→0** ⟶	0	d					
13	1	1	0	1	1	**1→1** ⟶	1	0					
14	1	1	1	0	1	**0→1** ⟶	1	d					
15	1	1	1	1	1	**1→1** ⟶	1	0		This Table is the same as Table 7a except that S/N 0, 2, 4, 6, 8, 10, 11, 12, & 14 are replaced with 'don't care terms, d'.			

The values of J & K are plotted into their respective K-Maps as shown in Table 8 from where the corresponding logic equations are derived and the logic circuit of the memory element or RAM Cell is given in Figure 7a.

Table 8: K-Maps for J & K terminals of JK-000 Rest Flip Flop (87.5%)

K-Map for J					**K-Map for K**				
$J = S_eIW + Q_n$ ..... (1)					K = 0 ..... (2)				
	S_eI					**S_eI**			
WQ_n	**00**	**01**	**11**	**10**	**WQ_n**	**00**	**01**	**11**	**10**
00	0^0	0^4	0^{12}	0^8	**00**	d^0	d^4	d^{12}	d^8
01	1^1	1^5	1^{13}	1^9	**01**	0^1	0^5	0^{13}	0^9
11	1^3	1^7	1^{15}	d^{11}	**11**	0^3	0^7	0^{15}	d^{11}
10	0^2	0^6	1^{14}	0^{10}	**10**	d^2	d^6	d^{14}	d^{10}

ISSN: 2277-9655
Impact Factor: 1.852

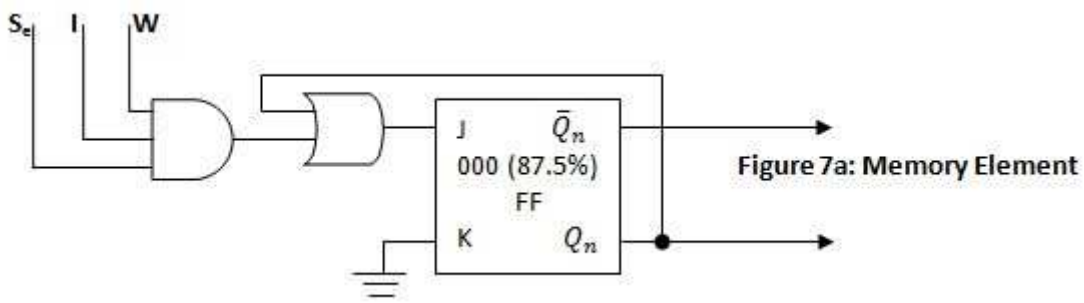


Figure 7a: Memory Element

Tables 9 & 10 contain the data employed to design the memory element using JK-001 Rest FLIP FLOP (87.5%)

Table 9a: Truth Table of a Memory Element And a JK-001 Rest FLIP FLOP (87.5%)

S/N	S_e	I	W	Q_n	Q_{n+1}	$Q_n \rightarrow Q_{n+1}$	J	K		J	K	Q_n	Q_{n+1}
0	0	0	0	0	0	**0→0** ⟶	0	1		0	0	0	1
1	0	0	0	1	1	**1→1** ⟶	0,1	0,0		0	0	1	1
2	0	0	1	0	0	**0→0** ⟶	0	1		0	1	0	0
3	0	0	1	1	1	**1→1** ⟶	0,1	0,0		0	1	1	0
4	0	1	0	0	0	**0→0** ⟶	0	1		1	0	0	1
5	0	1	0	1	1	**1→1** ⟶	0,1	0,0		1	0	1	1
6	0	1	1	0	0	**0→0** ⟶	0	1		1	1	0	1
7	0	1	1	1	1	**1→1** ⟶	0,1	0,0		1	1	1	0
8	1	0	0	0	0	**0→0** ⟶	0	1		JK-001 Rest Flip Flop (87.5%) Resting State = XYQ_n = 001			
9	1	0	0	1	1	**1→1** ⟶	0,1	0,0					
10	1	0	1	0	0	**0→0** ⟶	0	1					
11	1	0	1	1	0	**1→0** ⟶	0,1	1,1					
12	1	1	0	0	0	**0→0** ⟶	0	1					
13	1	1	0	1	1	**1→1** ⟶	0,1	0,0					
14	1	1	1	0	1	**0→1** ⟶	0,1,1	0,0,1					
15	1	1	1	1	1	**1→1** ⟶	0,1	0,0					

Table 9b: Truth Table of a Memory Element And an JK-001 Rest FLIP FLOP (87.5%)

S/N	S_e	I	W	Q_n	Q_{n+1}	$Q_n \rightarrow Q_{n+1}$	J	K		J	K	Q_n	Q_{n+1}
0	0	0	0	0	0	**0→0** ⟶	0	1		0	0	0	1
1	0	0	0	1	1	**1→1** ⟶	d	0		0	0	1	1
2	0	0	1	0	0	**0→0** ⟶	0	1		0	1	0	0
3	0	0	1	1	1	**1→1** ⟶	d	0		0	1	1	1
4	0	1	0	0	0	**0→0** ⟶	0	1		1	0	0	1
5	0	1	0	1	1	**1→1** ⟶	d	0		1	0	1	1
6	0	1	1	0	0	**0→0** ⟶	0	1		1	1	0	1
7	0	1	1	1	1	**1→1** ⟶	d	0		1	1	1	0
8	1	0	0	0	0	**0→0** ⟶	0	1		JK-001 Rest Flip Flop (87.5%) Resting State = XYQ_n = 001			
9	1	0	0	1	1	**1→1** ⟶	d	0					
10	1	0	1	0	0	**0→0** ⟶	0	1					
11	1	0	1	1	0	**1→0** ⟶	d	1					
12	1	1	0	0	0	**0→0** ⟶	0	1					
13	1	1	0	1	1	**1→1** ⟶	d	0		This Table is the same as Table 9a except that S/N			
14	1	1	1	0	1	**0→1** ⟶	d	D					

ISSN: 2277-9655
Impact Factor: 1.852

15	1	1	1	1	1	**1→1** ⟶	d	0		11, & 14 are replaced with 'don't care terms, d'.

The values of J & K are plotted into their respective K-Maps as shown in Table 10 from where the corresponding logic equations are derived.

Table 10: K-Maps for J & K terminals of JK-001 Rest Flip Flop (87.5%)

K-Map for J

X= 0……. (1)

S_eI / WQ_n	00	01	11	10
00	0^0	0^4	0^{12}	0^8
01	d^1	d^5	d^{13}	d^9
11	d^3	d^7	d^{15}	d^{11}
10	0^2	0^6	d^{14}	0^{10}

K-Map for K

Y = $\bar{Q}_n + S_e\bar{I}W +$……. (2)

S_eI / WQ_n	00	01	11	10
00	1^0	1^4	1^{12}	1^8
01	0^1	0^5	0^{13}	0^9
11	0^3	0^7	0^{15}	1^{11}
10	1^2	1^6	d^{14}	1^{10}

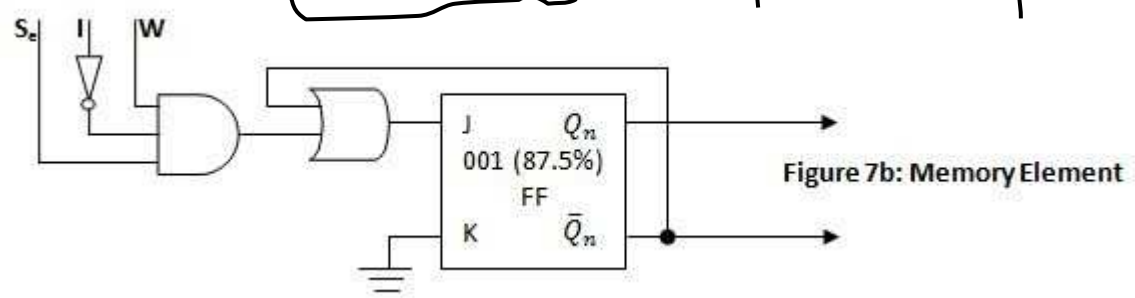


Figure 7b: Memory Element

The conventional JK-Flip Flop is also examined using Tables 11 & 12 for completeness which also resulted into memory element of Figure 7c.

Table 11a: Truth Table of a Memory Element And a Conventional JK- FLIP FLOP (75%)

S/N	S_e	I	W	Q_n	Q_{n+1}	$Q_n \rightarrow Q_{n+1}$	J	K		J	K	Q_n	Q_{n+1}
0	0	0	0	0	0	**0→0** ⟶	0,0	0,1		0	0	0	0
1	0	0	0	1	1	**1→1** ⟶	0,1	0,0		0	0	1	1
2	0	0	1	0	0	**0→0** ⟶	0,0	0,1		0	1	0	0
3	0	0	1	1	1	**1→1** ⟶	0,1	0,0		0	1	1	0
4	0	1	0	0	0	**0→0** ⟶	0,0	0,1		1	0	0	1
5	0	1	0	1	1	**1→1** ⟶	0,1	0,0		1	0	1	1
6	0	1	1	0	0	**0→0** ⟶	0,0	0,1		1	1	0	1
7	0	1	1	1	1	**1→1** ⟶	0,1	0,0		1	1	1	0
8	1	0	0	0	0	**0→0** ⟶	0,0	0,1		JK-Conventional FLIP FLOP (75%)			
9	1	0	0	1	1	**1→1** ⟶	0,1	0,0		00, J=0, K=x			
10	1	0	1	0	0	**0→0** ⟶	0,0	1,0		01, J=1, K=x			
11	1	0	1	1	0	**1→0** ⟶	0,1	1,1		10, J=x, K=1			
12	1	1	0	0	0	**0→0** ⟶	0,0	0,1		11, J=x, K=0			
13	1	1	0	1	1	**1→1** ⟶	0,1	0,0					
14	1	1	1	0	1	**0→1** ⟶	1,1	0,1					
15	1	1	1	1	1	**1→1** ⟶	0,1	0,0					

ISSN: 2277-9655
Impact Factor: 1.852

Table 11b: Truth Table of a Memory Element And an JK-Conventional FLIP FLOP (75%)

S/N	S_e	I	W	Q_n	Q_{n+1}	$Q_n \rightarrow Q_{n+1}$	J	K
0	0	0	0	0	0	**0→0**	0	d
1	0	0	0	1	1	**1→1**	d	0
2	0	0	1	0	0	**0→0**	0	d
3	0	0	1	1	1	**1→1**	d	0
4	0	1	0	0	0	**0→0**	0	d
5	0	1	0	1	1	**1→1**	d	0
6	0	1	1	0	0	**0→0**	0	d
7	0	1	1	1	1	**1→1**	d	0
8	1	0	0	0	0	**0→0**	0	d
9	1	0	0	1	1	**1→1**	d	0
10	1	0	1	0	0	**0→0**	0	d
11	1	0	1	1	0	**1→0**	d	1
12	1	1	0	0	0	**0→0**	0	d
13	1	1	0	1	1	**1→1**	d	0
14	1	1	1	0	1	**0→1**	1	d
15	1	1	1	1	1	**1→1**	d	0

J	K	Q_n	Q_{n+1}
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

JK-Conventional FLIP FLOP (75%)

The values of J & K are plotted into their respective K-Maps as shown in Table 12 from where the corresponding logic equations are derived.

Table 12: K-Maps for J & K terminals of JK-Conventional Flip Flop (75%)

K-Map for J

J= S_eIW ….. (1)

WQ_n	S_eI 00	01	11	10
00	0^0	0^4	0^{12}	0^8
01	d^1	0^5	d^{13}	d^9
11	d^3	d^7	d^{15}	d^{11}
10	0^2	0^6	1^{14}	0^{10}

K-Map for K

K = $S_e\bar{I}W$ ….. (2)

WQ_n	S_eI 00	01	11	10
00	d^0	d^4	d^{12}	d^8
01	0^1	x^5	0^{13}	0^9
11	0^3	0^7	0^{15}	1^{11}
10	d^2	d^6	d^{14}	d^{10}

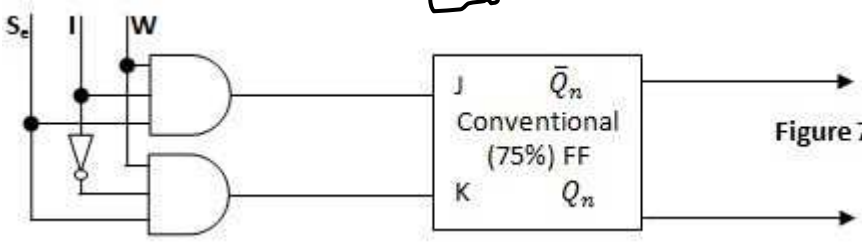


Figure 7c: Memory Element

The Summary of the above analysis is presented in Table 13

Table 13: Summary of Memory Element Design

S/N	TYPE OF FLIP FLOP	STORAGE DEVICE
1.	**JK-000 Rest (87.5%)**	This can be used to build Storage Media
2.	**JK-000, 001 Rest (75%) – Conventional JK-Flip Flop**	This can be used to build Storage Media
3.	**XY-No Rest (X=J, Y=K) (100%)**	This cannot be used to build Storage Media
4.	**JK-001 Rest (87.5%)**	This can be used to build Storage Media

NOTES:

1. Though JK-No Rest (100%) Flip Flops cannot be used to build Storage Devices but like D-Flip

ISSN: 2277-9655
Impact Factor: 1.852

Flops and T-Flip Flops which also cannot be used to design Memory Elements, they are still useful for other application areas.

2. JK-000 Rest (87.75%) Flip Flops as Memory Elements component parts has a speed advantage over SR-Flip Flops and JK- Conventional Flip Flops because it operates within only four transition states (JKQ = 000, 001, 100 & 101) instead of six (JKQ = 000, 001, 010, 011,100 & 101) for SR-Flip Flops and eight ((JKQ = 000, 001, 010, 011,100, 101, 110 & 111) for JK-Flip Flops.

Sequence Detector Devices

Let us examine a sequence detector as an example. Assume the sequence to be counted is given on Tables 14, 16 & 18 as 1,3, 0, 1.

Table 14: Sequence Detector Design Using JK-000 Rest Flip Flop

			FF1			FF2			JK-000 Rest FF TRUTH TABLE				
S/N	Q_2	Q_1	Q_1	J_1	K_1	J_2	K_2	Q_2	S/N	J	K	Q_n	Q_{n+1}
1	0	1	1→1,	1	0	1,1	0,1	0→1,	0	0	0	0→	0
3	1	1	1→0,	0.0,1	0,1, 1	0,0,1	0,1,1	1→0,	1	0	0	1→	0
0	0	0	0→1,	1,1	0,1	0,0	0,1	0→0,	2	0	1	0→	0
1	0	1	FF1 output changes & corresponding JK inputs			FF2 output changes & corresponding JK inputs			3	0	1	1→	0
Sequence of counting									4	1	0	0→	1
									5	1	0	1→	1
									6	1	1	0→	1
									7	1	1	1→	0

The values of J & K are plotted into their respective K-Maps as shown in Table 15 from where the corresponding logic equations are derived. The Logic circuits are presented in Figures 8a &8b

Table 15: K-Maps for J & K terminals of JK-000 Rest Flip Flop (87.5%)

FF1		FF2	
$J_1 = 1$		$J_2 = Q_1$	
Q_1 \ Q_2	0 1	Q_1 \ Q_2	0 1
0	1,1 d	0	0,0 d
1	1 0,0,1	1	1,1 0,0,1
$K_1 = 0$		$K_2 = 0\ or\ 1$	
Q_1 \ Q_2	0 1	Q_1 \ Q_2	0 1
0	0,1 d	0	0,1 d
1	0 0,1,1	1	0,1 0,1,1

ISSN: 2277-9655
Impact Factor: 1.852

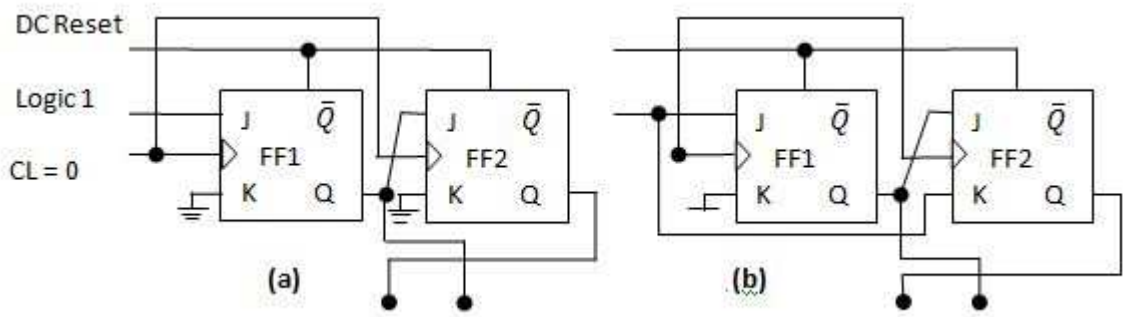
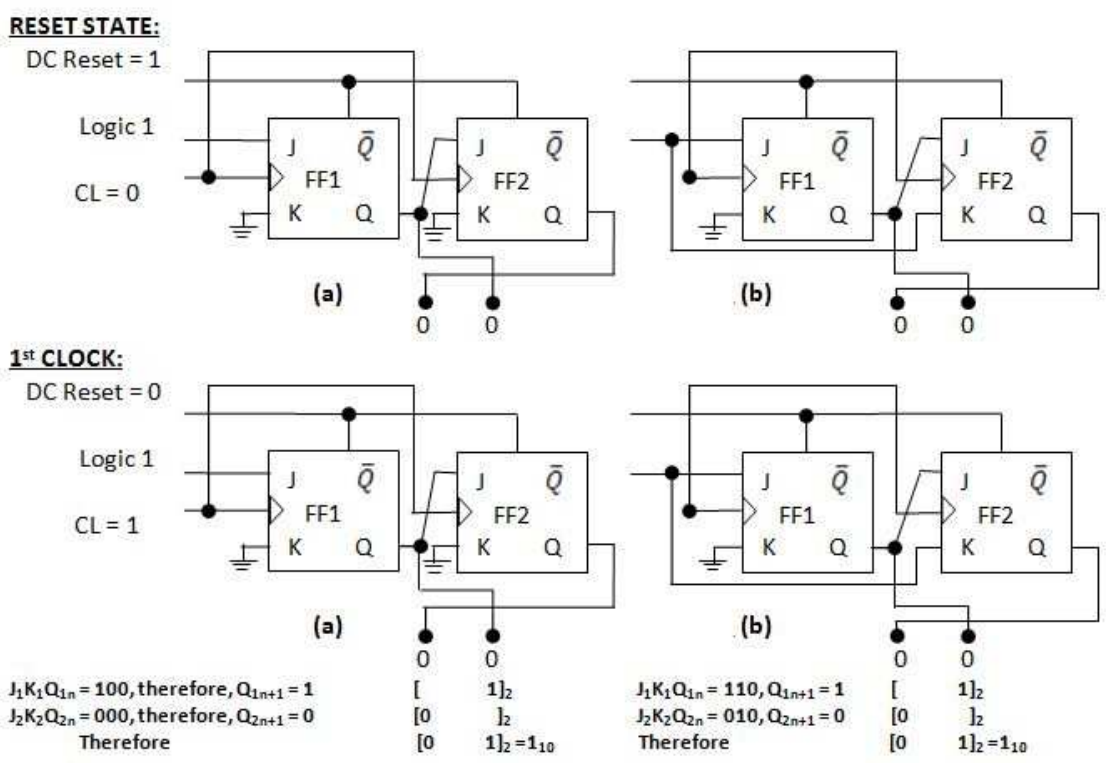


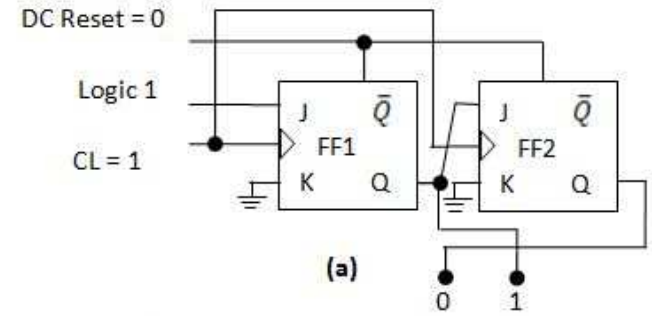
Figure 8: Sequence Detector Designed to detect Sequence 1,3,0,1 Using JK-000 Rest Flip Flop (87.5%)

Figure 8 is analysed to prove the correctness or otherwise of the design as follows:



ISSN: 2277-9655
Impact Factor: 1.852

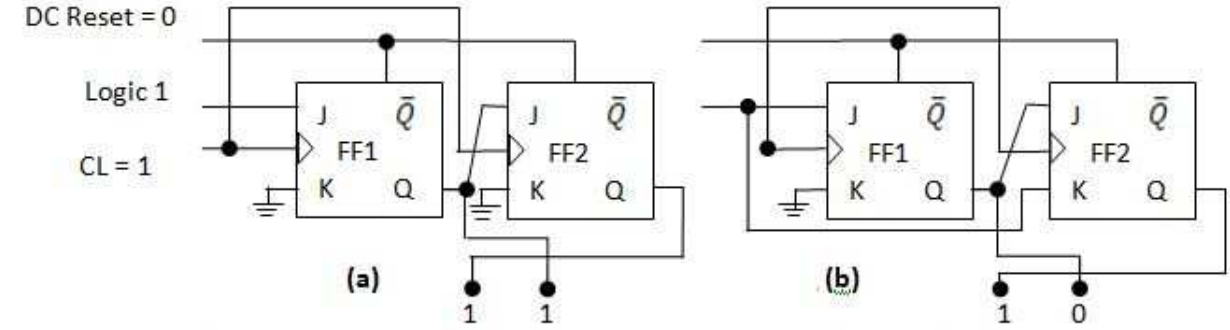
2nd CLOCK:



$J_1K_1Q_{1n} = 101$, therefore, $Q_{1n+1} = 1$ [1]$_2$
$J_2K_2Q_{2n} = 100$, therefore, $Q_{2n+1} = 1$ [1]$_2$
Therefore [1 1]$_2 = 3_{10}$

$J_1K_1Q_{1n} = 111$, $Q_{1n+1} = 0$ [0]$_2$
$J_2K_2Q_{2n} = 110$, $Q_{2n+1} = 1$ [1]$_2$
Therefore [1 0]$_2 = 2_{10}$

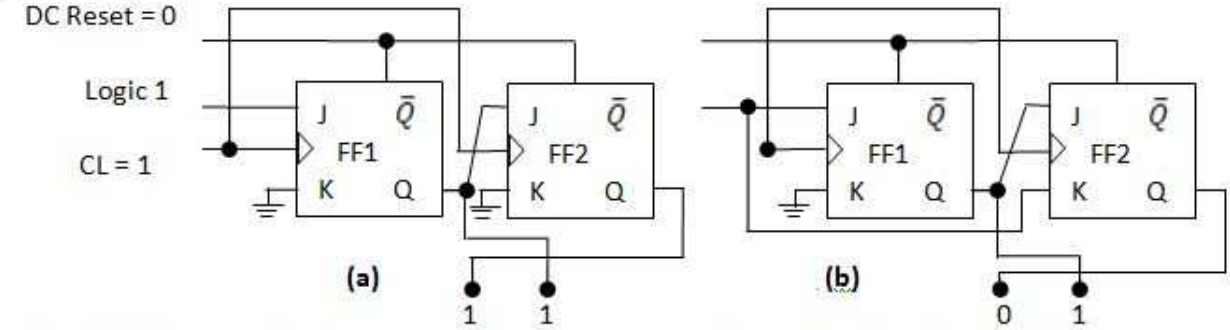
3rd CLOCK:



$J_1K_1Q_{1n} = 101$, therefore, $Q_{1n+1} = 1$ [1]$_2$
$J_2K_2Q_{2n} = 101$, therefore, $Q_{2n+1} = 1$ [1]$_2$
Therefore [1 1]$_2 = 3_{10}$

$J_1K_1Q_{1n} = 110$, $Q_{1n+1} = 1$ [1]$_2$
$J_2K_2Q_{2n} = 111$, $Q_{2n+1} = 0$ [0]$_2$
Therefore [0 1]$_2 = 1_{10}$

4th CLOCK:



$J_1K_1Q_{1n} = 101$, therefore, $Q_{1n+1} = 1$ [1]$_2$
$J_2K_2Q_{2n} = 101$, therefore, $Q_{2n+1} = 1$ [1]$_2$
Therefore [1 1]$_2 = 3_{10}$

$J_1K_1Q_{1n} = 111$, $Q_{1n+1} = 0$ [0]$_2$
$J_2K_2Q_{2n} = 110$, $Q_{2n+1} = 1$ [1]$_2$
Therefore [1 0]$_2 = 2_{10}$

Therefore, the sequence for (a) = 1,3,3,3 instead of 1,3,0....
Therefore, the sequence for (b) = 1,2,1,2 instead of 1,3,0....
Hence, this configuration cannot be used as a Sequence Detector.

Similarly, it can be proved that JK-001 Rest (87.5%) cannot be used as a Sequence Detector as follows:

ISSN: 2277-9655
Impact Factor: 1.852

Table 16: Sequence Detector Design Using JK-001 Rest Flip Flop

			FF1			FF2			JK-000 Rest FF TRUTH TABLE				
S/N	Q_2	Q_1	Q_1	J_1	K_1	J_2	K_2	Q_2	S/N	J	K	Q_n	Q_{n+1}
1	0	1	1→1,	0,1	0,0	0,1,1	0,0,1	0→1,	0	0	0	0→	1
3	1	1	1→0,	0.1	1,1	0,1	1,1	1→0,	1	0	0	1→	1
0	0	0	0→1,	0,1,1	0,0,1	0,0	1,1	0→0,	2	0	1	0→	0
1	0	1	FF1 output changes & corresponding JK inputs			FF2 output changes & corresponding JK inputs			3	0	1	1→	0
Sequence of counting									4	1	0	0→	1
									5	1	0	1→	1
									6	1	1	0→	1
									7	1	1	1→	0

The values of J & K are plotted into their respective K-Maps as shown in Table 17 from where the corresponding logic equations are derived. The Logic circuits are presented in Figures 9a &9b

Table 17: K-Maps for J & K terminals of JK-001 Rest Flip Flop (87.5%)

FF1

$J_1 = 0 \text{ or } 1$

Q_1 \ Q_2	0	1
0	0,1,1	d
1	0,1	0,1

$K_1 = Q_2$

Q_1 \ Q_2	0	1
0	0,0,1	d
1	0,0	1,1

FF2

$J_2 = 0$

Q_1 \ Q_2	0	1
0	0,0	d
1	0,1,1	0,1

$K_2 = 1$

Q_1 \ Q_2	0	1
0	0,1	d
1	0,0,1	1,1

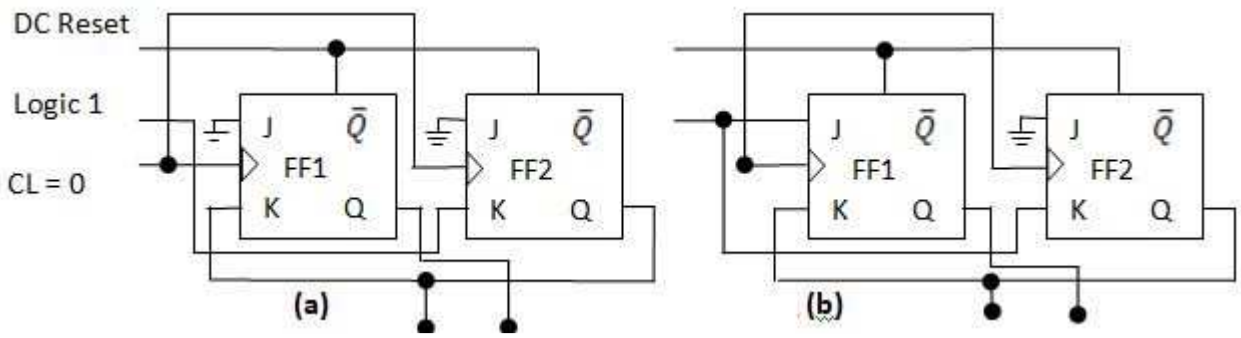


Figure 9: Sequence Detector Designed to detect Sequence 1,3,0,1 Using JK-000 Rest Flip Flop (87.5%)

Figure 9 is analysed to prove the correctness or otherwise of the design as follows:

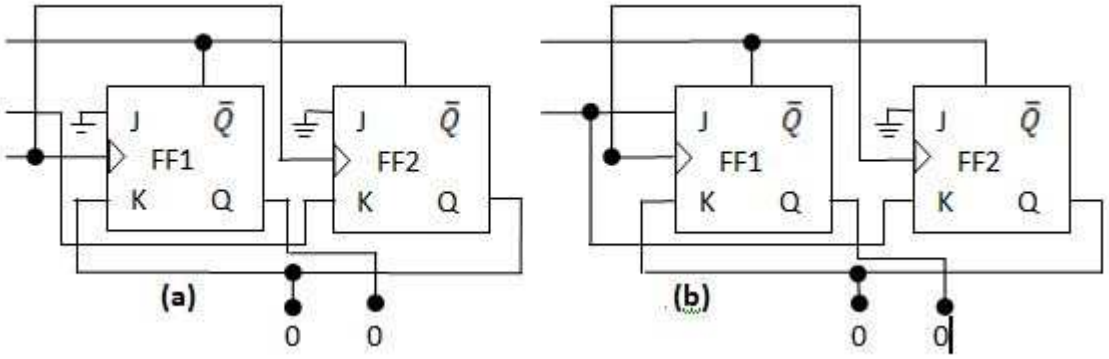
ISSN: 2277-9655
Impact Factor: 1.852

RESET STATE:

DC Reset = 1

Logic 1

CL = 0

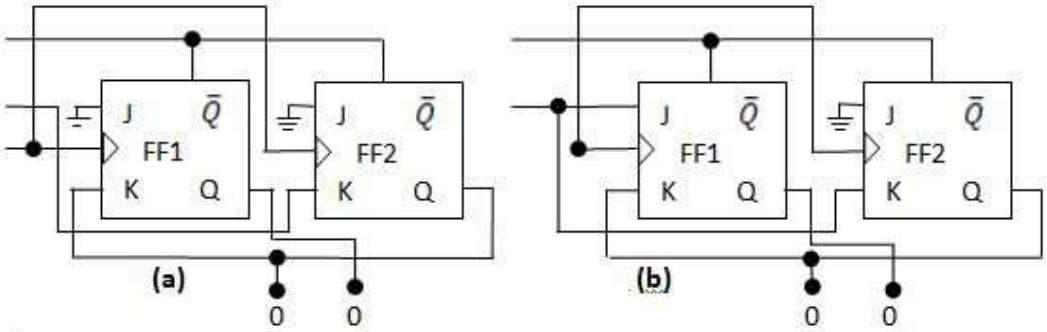


1st CLOCK:

DC Reset = 1

Logic 1

CL = 0



$J_1K_1Q_{1n} = 000$, therefore, $Q_{1n+1} = 0$ [0]$_2$
$J_2K_2Q_{2n} = 010$, therefore, $Q_{2n+1} = 0$ [0]$_2$
Therefore [0 0]$_2$ =0$_{10}$

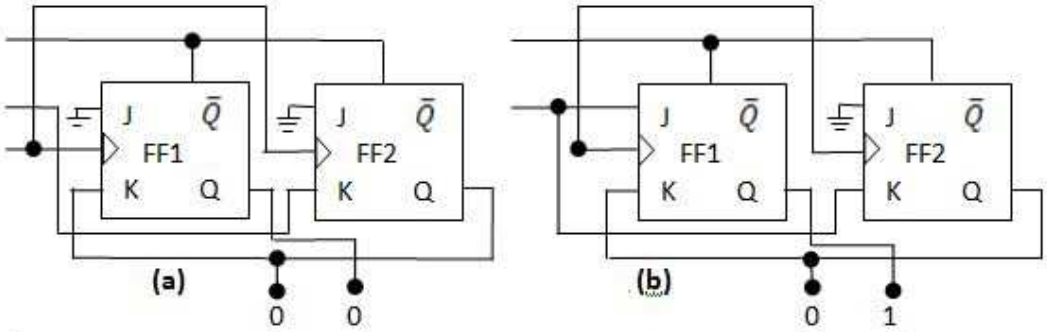
$J_1K_1Q_{1n} = 100$, $Q_{1n+1} = 1$ [1]$_2$
$J_2K_2Q_{2n} = 010$, $Q_{2n+1} = 0$ [0]$_2$
Therefore [0 1]$_2$ =1$_{10}$

2nd CLOCK:

DC Reset = 1

Logic 1

CL = 0



$J_1K_1Q_{1n} = 000$, therefore, $Q_{1n+1} = 0$ [0]$_2$
$J_2K_2Q_{2n} = 000$, therefore, $Q_{2n+1} = 0$ [0]$_2$
Therefore [0 0]$_2$ =0$_{10}$

$J_1K_1Q_{1n} = 101$, $Q_{1n+1} = 1$ [1]$_2$
$J_2K_2Q_{2n} = 010$, $Q_{2n+1} = 0$ [0]$_2$
Therefore [0 1]$_2$ =1$_{10}$

Therefore, the sequence for (a) = 0,0 instead of 1,3,0....
Therefore, the sequence for (b) = 1,1 instead of 1,3,0....
Hence, this configuration cannot be used as a Sequence Detector.

Table 18 is the design of the same Sequence Detector using JK-No Rest Flip Flop (100%).

ISSN: 2277-9655
Impact Factor: 1.852

Table 18: Sequence Detector Design Using JK-No Rest Flip Flop

			FF1			FF2			JK-No Rest FF TRUTH TABLE				
S/N	Q_2	Q_1	Q_1	J_1	K_1	J_2	K_2	Q_2	S/N	J	K	Q_n	Q_{n+1}
1	0	1	1→1,	1	0	1,1,0	0,1,0	0→1,	0	0	0	0→	1
3	1	1	1→0,	0.0,1	0,1, 1	0,0,1	0,1,1	1→0,	1	0	0	1→	0
0	0	0	0→1,	1,1,0	0,1,0	0,0	0,1	0→0,	2	0	1	0→	0
1	0	1	FF1 output changes & corresponding JK inputs			FF2 output changes & corresponding JK inputs			3	0	1	1→	0
Sequence of counting									4	1	0	0→	1
									5	1	0	1→	1
									6	1	1	0→	1
									7	1	1	1→	0

The values of J & K are plotted into their respective K-Maps as shown in Table 19 from where the corresponding logic equations are derived. The Logic circuits are presented in Figures 10a &10b

Table 19: K-Maps for J & K terminals of JK-000 Rest Flip Flop (87.5%)

FF1

$J_1 = 1$

Q_1 \ Q_2	0	1
0	1,1,0	d
1	1	0,0,1

$K_1 = 0$

Q_1 \ Q_2	0	1
0	0,1,0	d
1	0	0,1,1

FF2

$J_2 = 0$

Q_1 \ Q_2	0	1
0	0,0	d
1	1,1,0	0,0,1

$K_2 = 0 \; or 1$

Q_1 \ Q_2	0	1
0	0,1	d
1	0,1,0	0,1,1

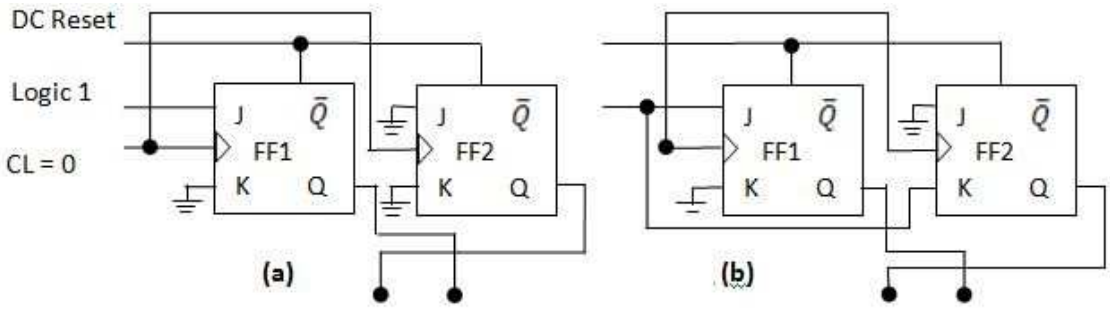


Figure 10: Sequence Detector Designed to detect Sequence 1,3,0,1 Using JK-000 Rest Flip Flop (87.5%)

Similarly, Figure 10 is analysed to prove the correctness or otherwise of the design as follows:

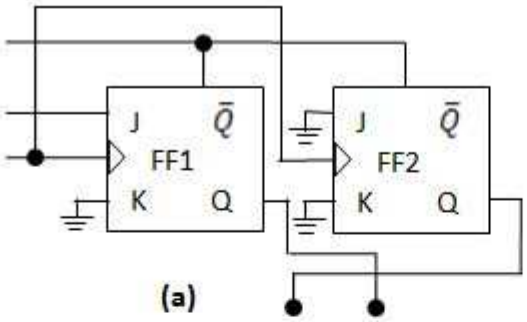
ISSN: 2277-9655
Impact Factor: 1.852

RESET STATE:

DC Reset = 1

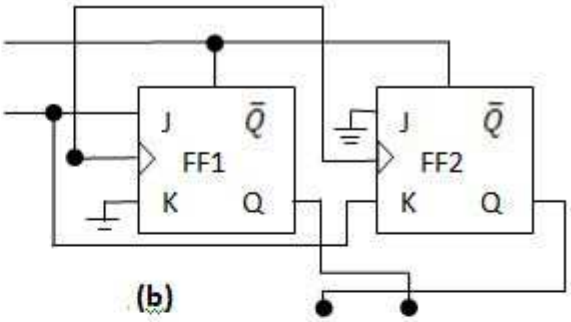
Logic 1

CL = 0



(a)

0 0



(b)

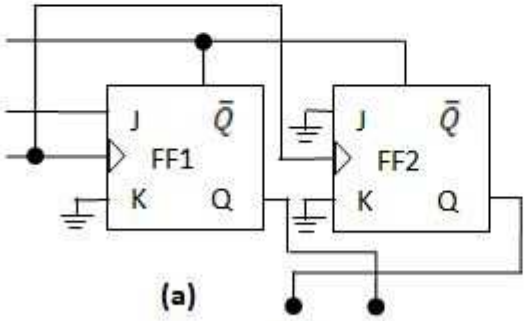
0 0

1st CLOCK:

DC Reset = 0

Logic 1

CL = 1



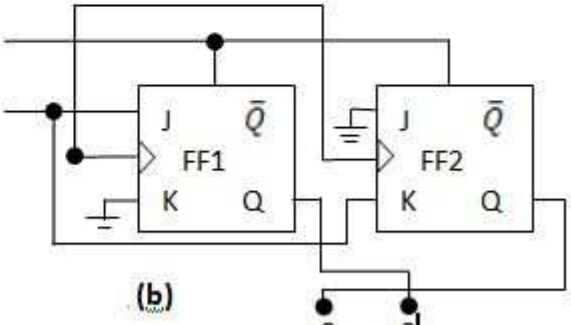
(a)

0 0

$J_1K_1Q_{1n} = 100$, therefore, $Q_{1n+1} = 1$ [1]$_2$

$J_2K_2Q_{2n} = 000$, therefore, $Q_{2n+1} = 0$ [0]$_2$

Therefore [0 1]$_2 = 1_{10}$



(b)

0 0

$J_1K_1Q_{1n} = 100$, $Q_{1n+1} = 1$ [1]$_2$

$J_2K_2Q_{2n} = 010$, $Q_{2n+1} = 0$ [0]$_2$

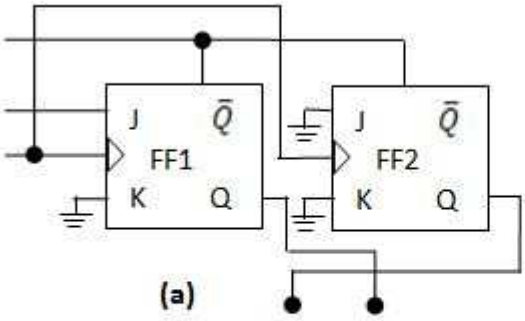
Therefore [0 1]$_2 = 1_{10}$

2nd CLOCK:

DC Reset = 0

Logic 1

CL = 1



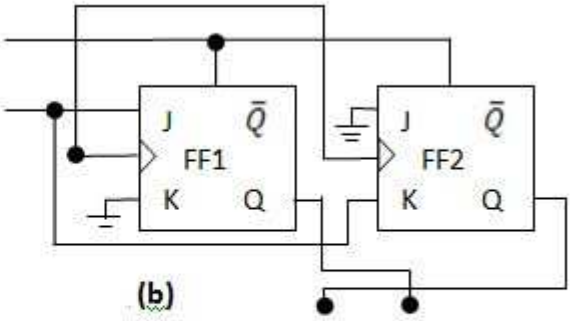
(a)

0 1

$J_1K_1Q_{1n} = 101$, therefore, $Q_{1n+1} = 1$ [1]$_2$

$J_2K_2Q_{2n} = 100$, therefore, $Q_{2n+1} = 1$ [1]$_2$

Therefore [1 1]$_2 = 3_{10}$



(b)

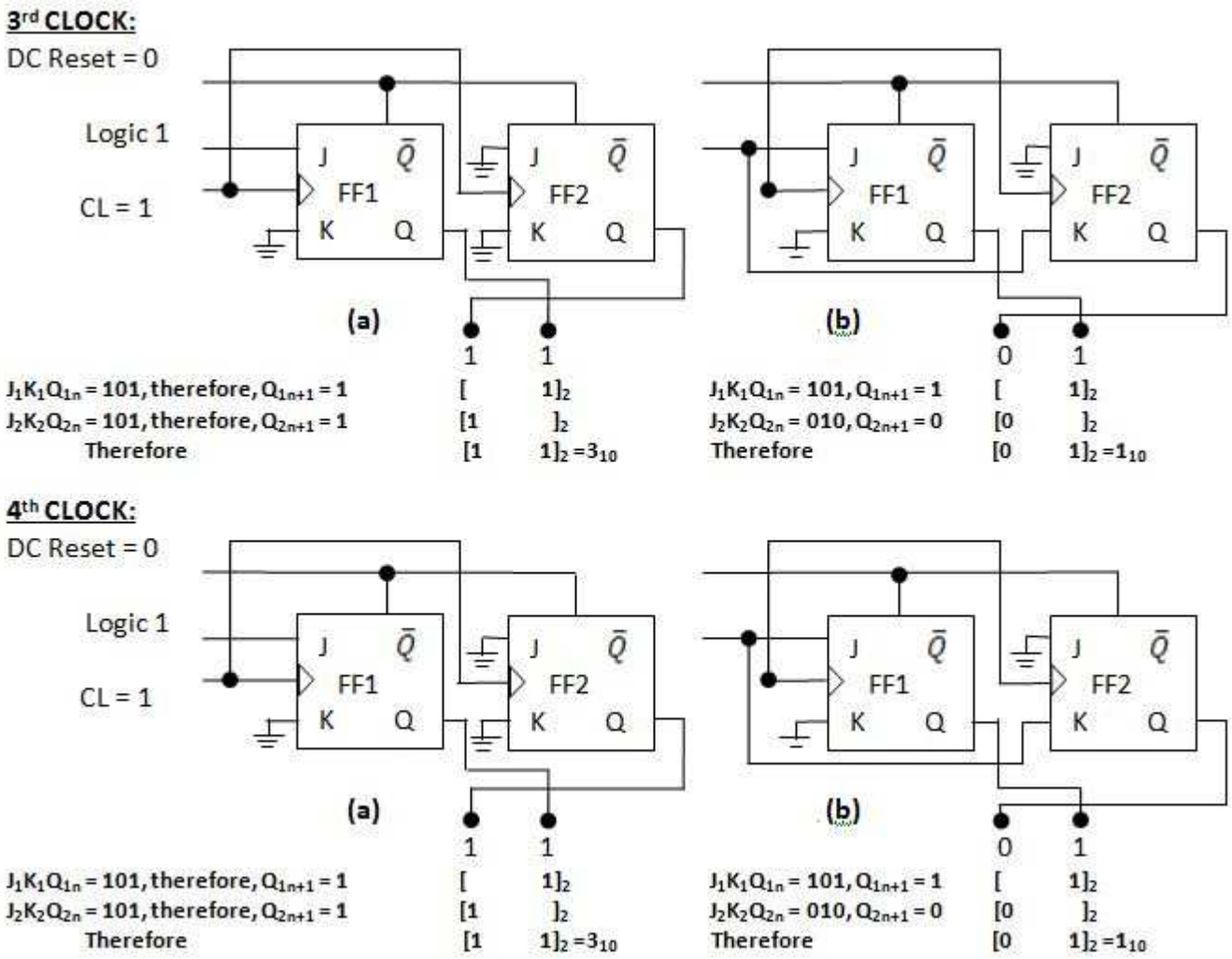
0 1

$J_1K_1Q_{1n} = 100$, $Q_{1n+1} = 1$ [1]$_2$

$J_2K_2Q_{2n} = 010$, $Q_{2n+1} = 0$ [0]$_2$

Therefore [0 1]$_2 = 1_{10}$

ISSN: 2277-9655
Impact Factor: 1.852



Therefore, the sequence for (a) = 1,3,3,3 instead of 1,3,0....
Therefore, the sequence for (b) = 1,1,1,1 instead of 1,3,0....
Hence, this configuration cannot be used as a Sequence Detector.

Figure 11 is the logic circuit of a Sequence Detector using SR-Flip Flops to detect sequence 1,3,0,1...

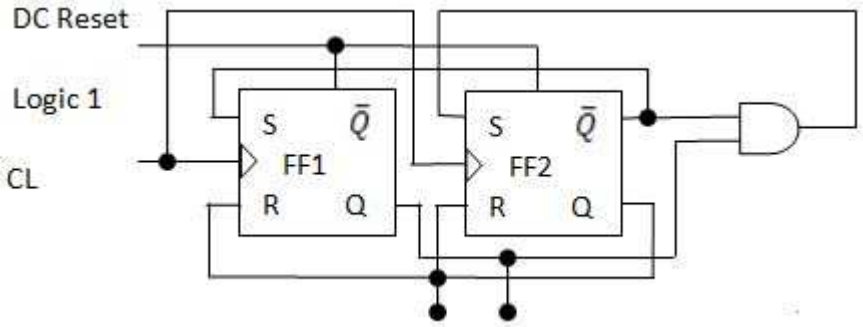


Figure 11: Sequence Detector Designed to detect Sequence 1,3,0,1 Using SR-Flip Flop

The proof that Figure 11 as a Sequence Detector, detecting sequence 1,3,0,1.. is presented as follows:

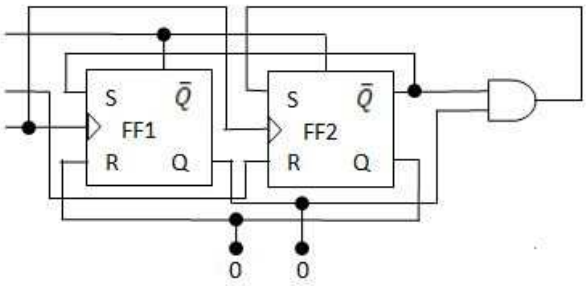
ISSN: 2277-9655
Impact Factor: 1.852

RESET STATE

DC Reset = 1

Logic 1

CL = 0

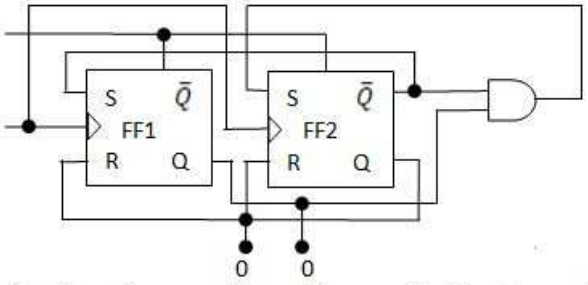


1st CLOCK

DC Reset = 0

Logic 1

CL = 1



$S_1R_1Q_{1n} = 100$, therefore, $Q_{1n+1} = 1$ [1]$_2$ $S_1 = \bar{Q}_2 = 1$ $R_1 = Q_2 = 0$

$S_2R_2Q_{2n} = 000$, therefore, $Q_{2n+1} = 0$ [0]$_2$ $S_2 = Q_1\bar{Q}_2 = 0$ $R_2 = R_1 = 0$

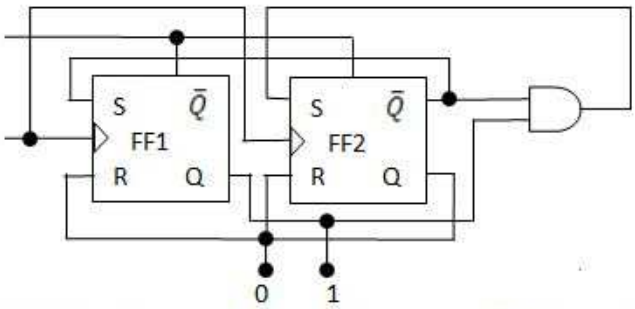
Therefore [0 1]$_2$ =1$_{10}$

2nd CLOCK

DC Reset = 0

Logic 1

CL = 1



$S_1R_1Q_{1n} = 101$, therefore, $Q_{1n+1} = 1$ [1]$_2$ $S_1 = \bar{Q}_2 = 1$ $R_1 = Q_2 = 0$

$S_2R_2Q_{2n} = 100$, therefore, $Q_{2n+1} = 1$ [1]$_2$ $S_2 = Q_1\bar{Q}_2 = 1$ $R_2 = R_1 = 0$

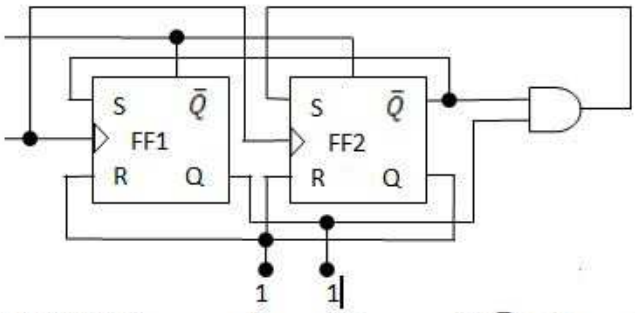
Therefore [1 1]$_2$ =3$_{10}$

3rd CLOCK

DC Reset = 0

Logic 1

CL = 1

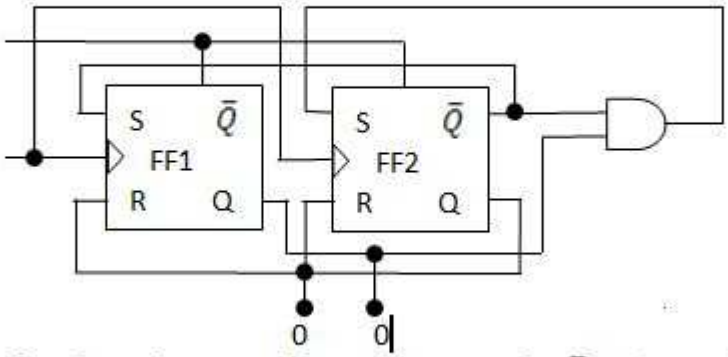


$S_1R_1Q_{1n} = 011$, therefore, $Q_{1n+1} = 0$ [0]$_2$ $S_1 = \bar{Q}_2 = 0$ $R_1 = Q_2 = 1$

ISSN: 2277-9655
Impact Factor: 1.852

$S_2R_2Q_{2n} = 011$, therefore, $Q_{2n+1} = 0$ [0]$_2$ $S_2 = Q_1\bar{Q}_2 = 0$ $R_2 = R_1 = 1$
Therefore [0 0]$_2$ =0_{10}

4th CLOCK
DC Reset = 0
Logic 1
CL = 1



$S_1R_1Q_{1n} = 100$, therefore, $Q_{1n+1} = 1$ [1]$_2$ $S_1 = \bar{Q}_2 = 1$ $R_1 = Q_2 = 0$
$S_2R_2Q_{2n} = 000$, therefore, $Q_{2n+1} = 0$ [0]$_2$ $S_2 = Q_1\bar{Q}_2 = 0$ $R_2 = R_1 = 0$
Therefore [0 1]$_2$ =1_{10}

Therefore, the sequence for Figure 8c = 1,3,0,1 as required
Hence, this configuration, Figure 8c can be used as a Sequence Detector.

Conclusion

In this paper, analysis and design of different Flip Flops as extensions of conventional JK-Flip Flops is presented. It is observed from the analysis of the different design, that the number of active transitions/states is the same with the number of gates required to achieve these transitions for SR at 50%, JK at 75% and JK Extension (XY- No rest) at 100% utilization. But it is obvious that the JK Extension (XY-No rest) at 100% Flip Flops cannot be used to build storage devices; but like the D-Flip Flops and the T- Flip Flops, which also cannot be used to design memory elements, these types of Flip Flops are still useful for other digital device applications.

It is also observed that the number of gates is less than the number of transitions/states for the 87.5% utilization which can only be configured by NAND gates. The JK-000 Rest (87.75%) Flip Flops have been proven beyond doubt that they are capable of being used to design Memory Elements among other digital device applications. They also possess speed advantage over SR- and JK- Conventional Flip Flops because they operate within only four transition states (JKQ = 000, 001, 100 & 101) instead of six (JKQ = 000, 001, 010, 011,100 & 101) for SR-Flip Flops and eight ((JKQ = 000, 001, 010, 011,100, 101, 110 & 111) for JK-Flip Flops.

From the numerous examples shown in section 6 of this paper, it is highly convincing that the JK-Flip Flops Extension either as 87.5% or 100% utilization cannot be used effectively to design Sequence Detectors. The possibilities of using the different types of Flip Flips designed here in other application areas can be similarly exploited.

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

ISSN: 2277-9655
Impact Factor: 1.852

[8] P. K. Meher, "Efficient implementation of accumulation in finite field over *GF* (2*m*) and its applications", *IEEE Transactions on Very Large Scale Integration Systems*, accepted for future publication. [2] N. H. E. Weste and D. Harris, *CMOS VLSI Design.*

[9] B. P. Singh, N. Arora, K. Mehta, "Low Power Efficient D Flip Flop Circuit", *International Symposium on Devices MEMS, Intelligent Systems & Communication (ISDMISC) 2011 Proceedings published by International Journal of Computer* Applications® *(IJCA)*