

Logic Gates

Sequential Logic Circuits

1

Course Outline

- | | |
|----------------------------------|-------------------------------------|
| 1. Introduction | 8. Sequential Circuit I |
| 2. Logic Gates & Related Devices | 9. Sequential Circuit II |
| 3. Combination Circuits I | 10. Sequential Circuit III |
| 4. Combination Circuits II | 11. Sequential Circuit IV |
| 5. Combination Circuits III | 12. FPGA + Logic circuit project I |
| 6. Combination Circuits IV, Kmap | 13. FPGA + Logic circuit project II |
| 7. Combination Circuits IV, Kmap | 14. Lab Exam |

2

Logic Circuit Laboratory#8

Sequential Circuit II

Flip Flop

1. LATCH Flip Flop
2. Clocked Flip Flop

Sequential Circuit II

1. RS Flip Flop

- NAND
- NOR

2. JK Flip Flop

3. D Flip Flop

4. T Flip Flop

5

SR Flip Flop

NAND Latch

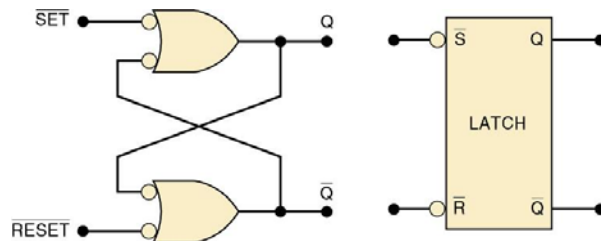
6

Sequential Circuit II

RS Flip Flop = NAND Flip Flop = NAND Latch

Active Low

เมื่อ Input SET/RESET เป็น Low ค่า Q จะออกตาม Input
(SET 0, Q 1) และ (RESET 0, Q' 1)



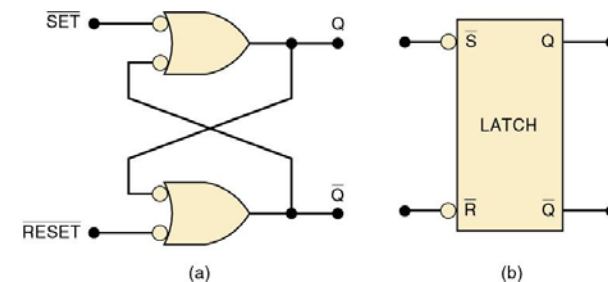
7

Sequential Circuit II

Active Low

จาก (SET 0, Q 1) และ (RESET 0, Q' 1)

ดังนั้น ถ้า SET และ RESET เป็น 0 ทั้งคู่ NAND Latch จะ Error
(คุณสมบัติของ Flip-flop ค่า Q และ Q' มีค่าไม่เหมือนกัน)

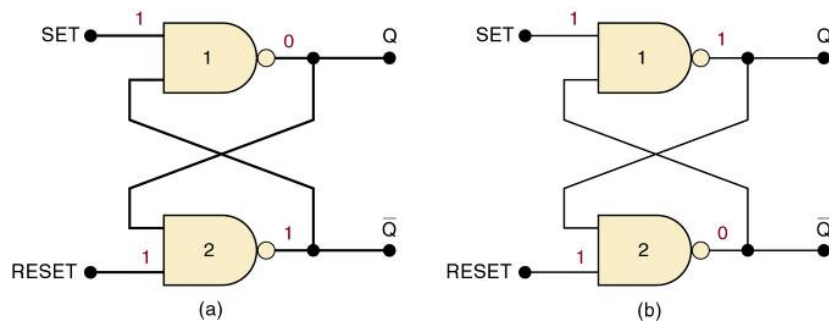


ดังนั้น การคงสถานะเดิม (Memory) คือการ Input SET และ RESET เป็น 1 ทั้งคู่

8

Sequential Circuit II

การคงสถานะเดิม (Memory) คือการ Input SET และ RESET เป็น 1 ทั้งคู่



9

Sequential Circuit II

สรุป RS Flip Flop แบบ NAND Gates

- SET = 0, RESET = 1 → Q = 1
- SET = 1, RESET = 0 → Q' = 1
- SET = 1, RESET = 1 → Memory (คงสถานะเดิม)
- SET = 0, RESET = 0 → Error

10

SR Flip Flop NOR Latch

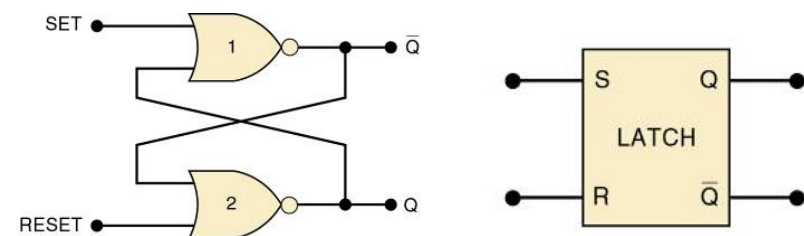
11

Sequential Circuit II

RS Flip Flop = NOR Flip Flop = NOR Latch

Active High

เมื่อ Input SET/RESET เป็น HIGH ค่า Q จะออกตาม Input (SET 1, Q 1) และ (RESET 1, Q' 1)



12

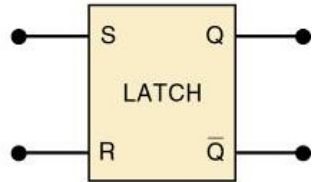
Sequential Circuit II

Active High

จาก (SET 1, Q 1) และ (RESET 1, Q' 1)

ดังนั้น ถ้า SET และ RESET เป็น 1 ทั้งคู่ NAND Latch จะ Error

(คุณสมบัติของ Flip-flop ค่า Q และ Q' มีค่าไม่เหมือนกัน)



Set	Reset	Output
0	0	No change
1	0	Q = 1
0	1	Q = 0
1	1	Invalid*

*Produces Q = Q-bar = 0.

ดังนั้น การคงสถานะเดิม (Memory) คือการ Input SET และ RESET เป็น 0 ทั้งคู่

13

Sequential Circuit II

สรุป RS Flip Flop แบบ NOR Gates

- SET = 0, RESET = 1 → Q' = 1
- SET = 1, RESET = 0 → Q = 1
- SET = 1, RESET = 1 → Error
- SET = 0, RESET = 0 → Memory (คงสถานะเดิม)

14

SR Flip Flop

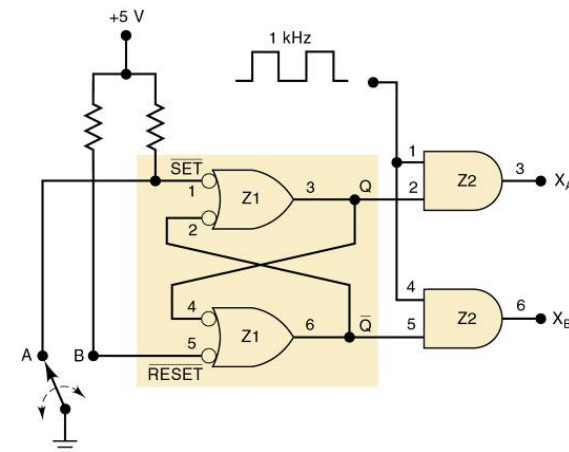
แบบ Clock Flip Flop

15

Sequential Circuit II

SR Flip Flop จะต่อแบบ Latch แต่สามารถออกแบบให้ต่อกับ Clock ได้เช่นกัน

ตัวอย่างการต่อใช้งาน Clocked Flip Flop แบบ SR Flip Flop

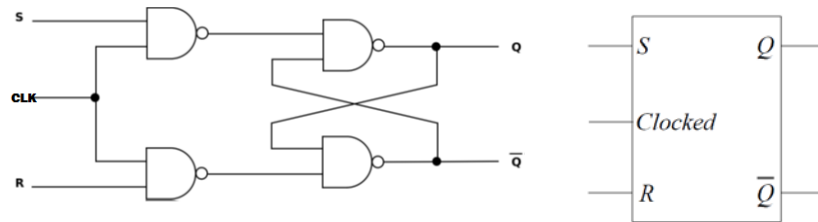


Switch position	X _A	X _B
A	Pulses	LOW
B	LOW	Pulses

16

Sequential Circuit II

SR Flip Flop จะต่อแบบ Latch แต่สามารถออกแบบให้ต่อกับ Clock ได้เช่นกัน
ตัวอย่างการต่อใช้งาน Clocked Flip Flop แบบ SR Flip Flop



17

Sequential Circuit II

SR Flip Flop จะต่อแบบ Latch แต่สามารถออกแบบให้ต่อกับ Clock ได้เช่นกัน
ตัวอย่างการต่อใช้งาน Clocked Flip Flop แบบ SR Flip Flop

CLK	S	R	Q	Q'
0	X	X	Memory (Not Change)	
1	0	1	0	1
1	1	0	1	0
1	0	0	Memory (Not Change)	
1	1	1	Error	

18

Sequential Circuit II

1. RS Flip Flop

- NAND
- NOR

2. JK Flip Flop ←

3. D Flip Flop

4. T Flip Flop

19

JK Flip Flop

Clock Flip Flop

20

Sequential Circuit II

JK Flip Flop ปรับปรุงวงจร RS Flip Flop

*ปรับในส่วนของ Error จะเปลี่ยน Error ให้ Output ออกตรงกันข้าม

RS FF ปกติ

S	R	Q	Q'
0	1		1
1	0	1	
0	0	Q_n	\bar{Q}_n
1	1	Error	

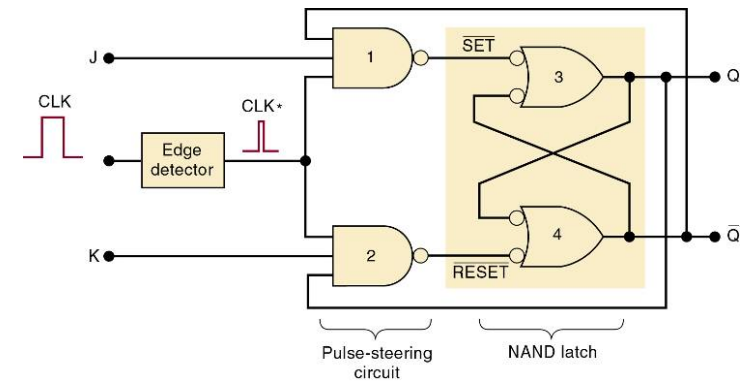
JK FF

J	K	Q	Q'
0	1		1
1	0	1	
0	0	Q_n	\bar{Q}_n
1	1	\bar{Q}_n	Q_n

21

Sequential Circuit II

JK Flip Flop ปรับปรุงวงจร RS Flip Flop

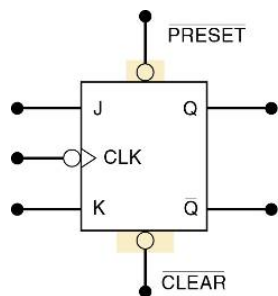


22

Sequential Circuit II

*ปรับในส่วนของ Error จะเปลี่ยน Error ให้ Output ออกตรงกันข้าม

** เพิ่ม PRE และ CLR -> Active Low ทำงานแบบ Asynch ไม่ขึ้นกับ Clock



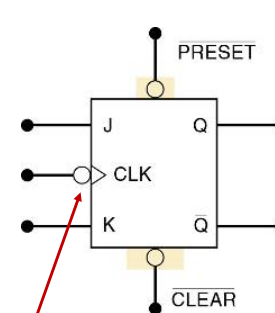
J	K	Clk	PRE	CLR	Q
0	0	↓	1	1	Q (no change)
0	1	↓	1	1	0 (Synch reset)
1	0	↓	1	1	1 (Synch set)
1	1	↓	1	1	\bar{Q} (Synch toggle)
x	x	x	1	1	Q (no change)
x	x	x	1	0	0 (asynch clear)
x	x	x	0	1	1 (asynch preset)
x	x	x	0	0	(Invalid)

23

Sequential Circuit II

การใช้งาน ถ้าต้องการใช้งานแบบ Synchronous (ขา PRE และ CLR ต้อง = 1 ทั้งคู่)

* เมื่อเป็น Synchronous -> Check Clock



J	K	Clk	PRE	CLR	Q
0	0	↓	1	1	Q (no change)
0	1	↓	1	1	0 (Synch reset)
1	0	↓	1	1	1 (Synch set)
1	1	↓	1	1	\bar{Q} (Synch toggle)
x	x	x	1	1	Q (no change)
x	x	x	1	0	0 (asynch clear)
x	x	x	0	1	1 (asynch preset)
x	x	x	0	0	(Invalid)

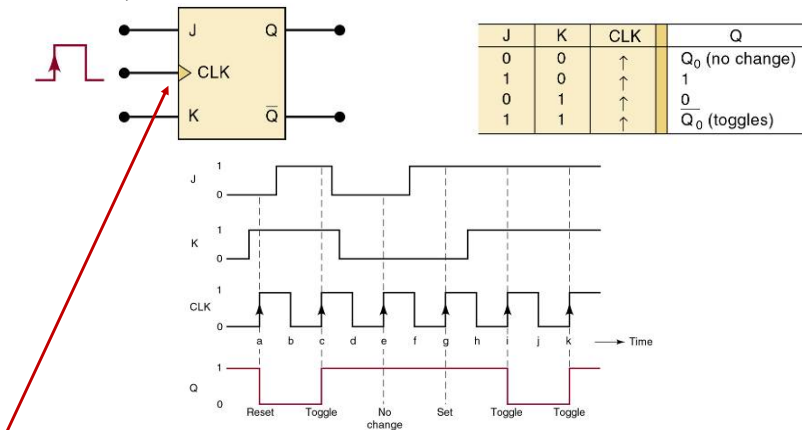
สัญลักษณ์ Not ก่อนถึง Clock หมายถึง Falling edge trigger ตรวจสอบ Input เมื่อพบขอบขา

24

Sequential Circuit II

การใช้งาน ถ้าต้องการใช้งานแบบ Synchronous (ขา PRE และ CLE ต้อง = 1 ทั้งคู่)

* เมื่อเป็น Synchronous -> Check Clock



ถ้าไม่มีสัญลักษณ์ Not ก่อนถึง Clock หมายถึง Rising edge trigger ตรวจสอบ Input เมื่อพบขอบขาขึ้น

25

Sequential Circuit II

1. RS Flip Flop

■ NAND

■ NOR

2. JK Flip Flop

3. D Flip Flop ←

4. T Flip Flop

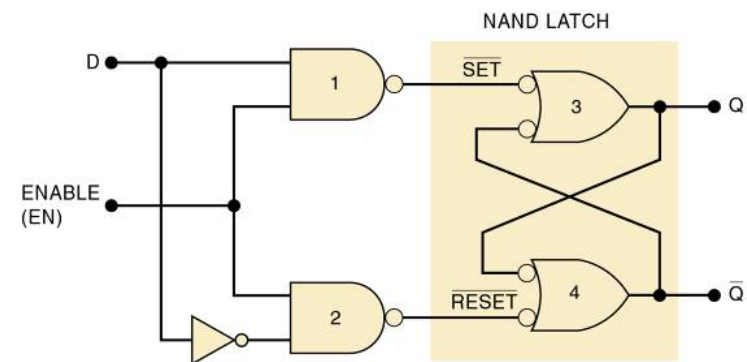
26

D Flip Flop

Clock Flip Flop

Sequential Circuit II

D Flip Flop สร้างจาก SR FF แบบ NAND Gate ใช้งานง่าย การออกแบบดังนี้

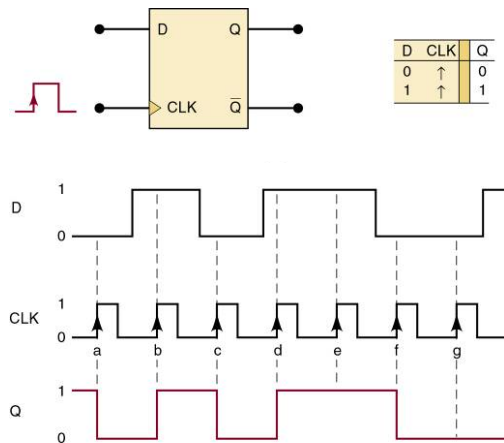


27

28

Sequential Circuit II

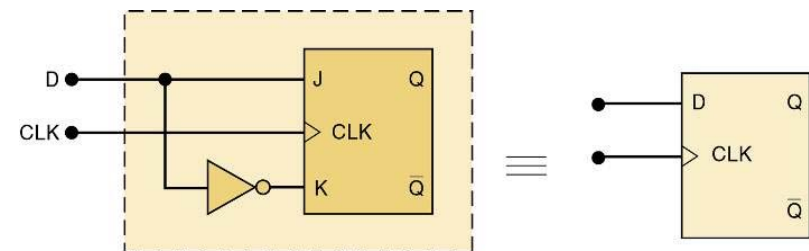
D Flip Flop สร้างจาก NAND Gate ใช้งานง่าย การออกแบบดังนี้



29

Sequential Circuit II

D Flip Flop สามารถสร้างจาก JK FF ได้เช่นกัน



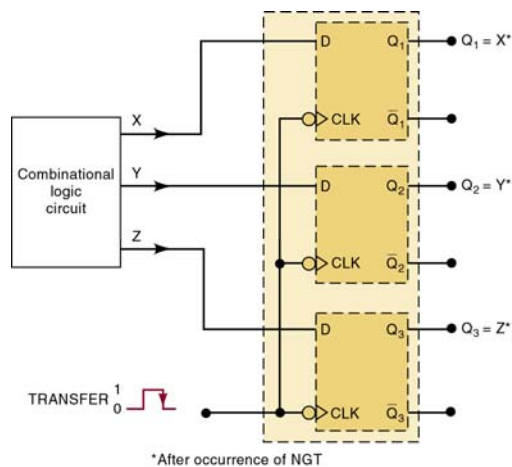
** JK Flip Flop ขา PRE และ CLR ให้ Input logic High (1 ทั้งคู่)

30

Sequential Circuit II

D Flip Flop นำเอาประยุกต์ในการออกแบบได้สะดวก ถ้าไม่ใช้ Function Asynch

Basic Memory
3 Bit



31

Sequential Circuit II

1. RS Flip Flop

- NAND
- NOR

2. JK Flip Flop

3. D Flip Flop

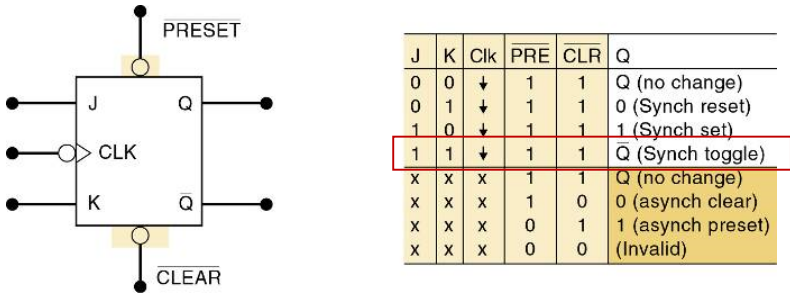
4. T Flip Flop ←

32

T Flip Flop Clock Flip Flop

Sequential Circuit II

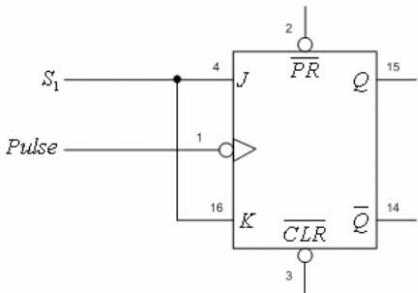
T Flip Flop = Toggle Flip Flop
 T Flip Flop = JK ในสถานะ Toggle มาออกแบบ (J = 1, K = 1)



ดังนั้น J = 0 , K = 0 จะเป็น No Change
 ดังนั้น J = 1 , K = 1 จะเป็น Toggle

Sequential Circuit II

T Flip Flop = Toggle Flip Flop
 T Flip Flop = JK ในสถานะ Toggle มาออกแบบ (J = 1, K = 1)



ต่อขา J และ K ของ JK Flip Flop เข้าด้วยกัน จะได้ T Flip Flop

START

LAB7 SequentialCircuits2

- D Flip Flop ← 74LS74 D Flip Flop
- JK Flip Flop ← 74LS73 JK Flip Flop
- T Flip Flop ← 74LS73 JK Flip Flop