

# REPORT

---

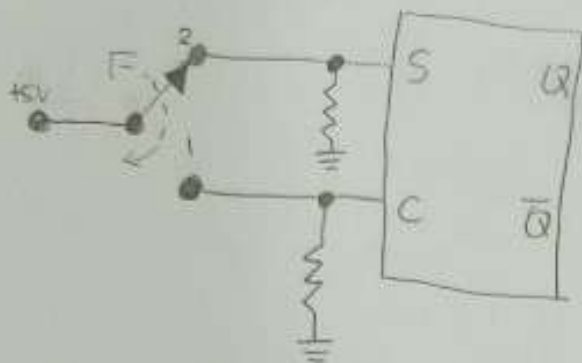


인하대학교  
INHA UNIVERSITY

과목명 | 논리회로  
담당교수 | 최성용  
학과 | 컴퓨터공학과  
학년 | 2  
학번 | 12171661  
이름 | 윤혁  
제출일 |

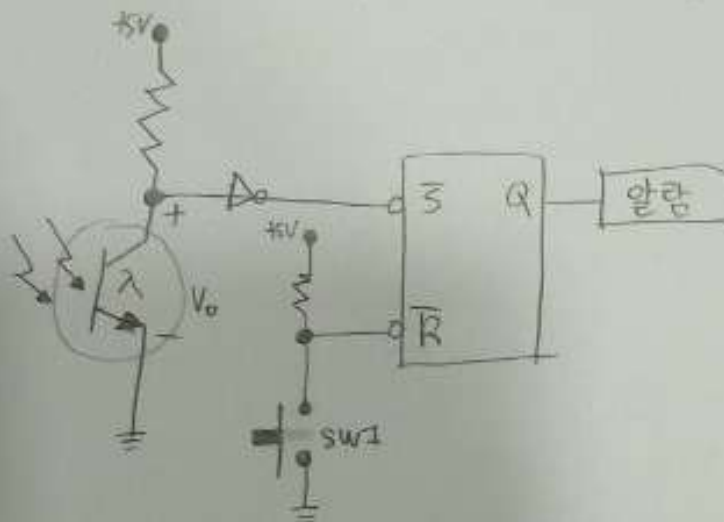


5-4) NOR gate 래치는 NAND gate 래치와 반대로 Active-High로 동작한다



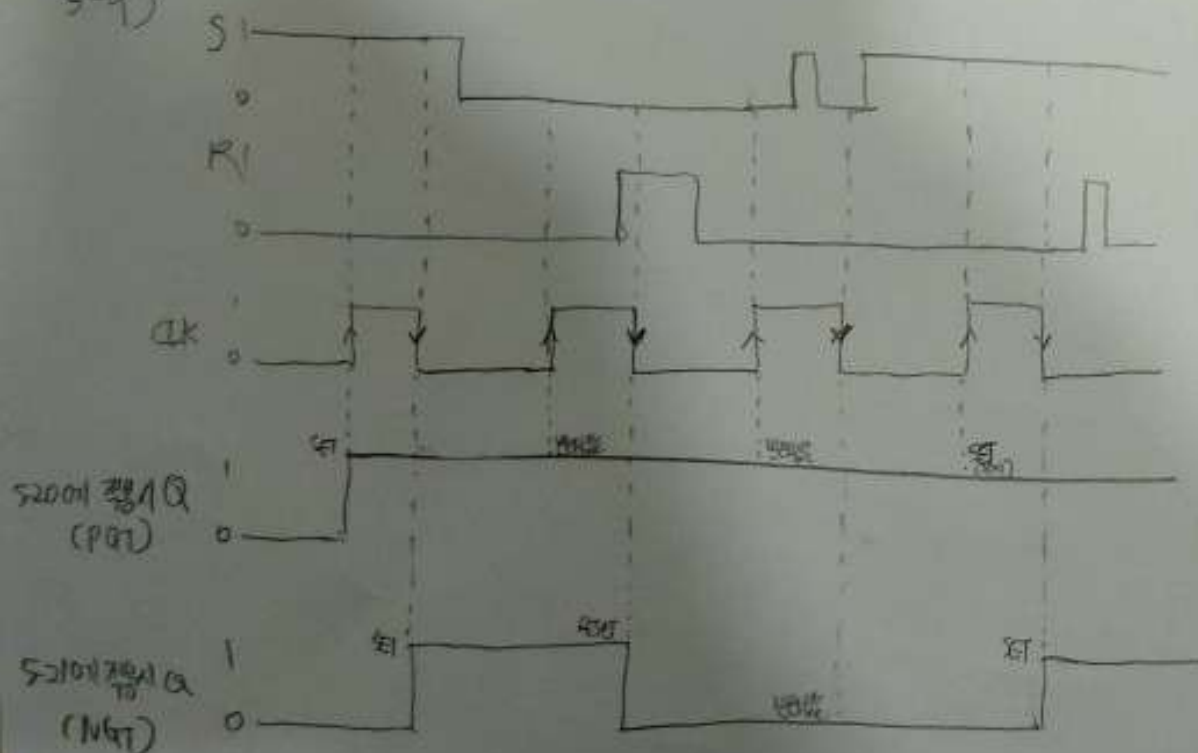
스위치가 잠금 고이 있을 때 Q가 1이 되도록 한다

5-5)

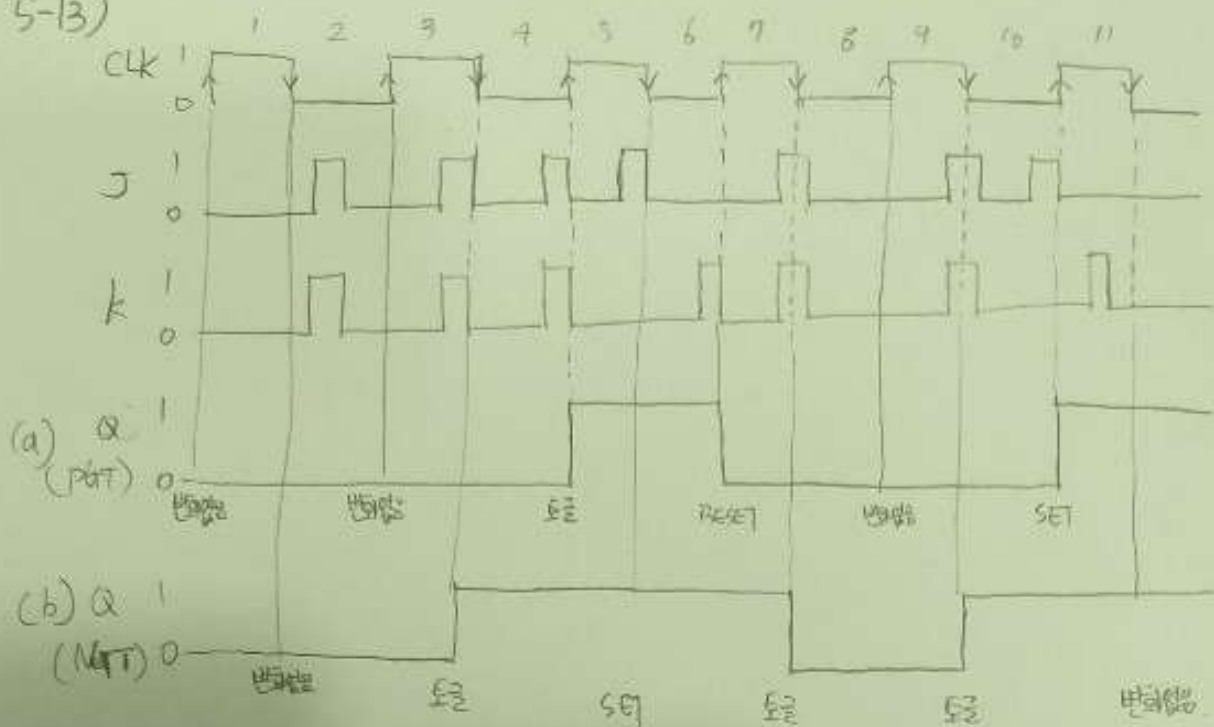


NAND gate 래치 이므로 Active-Low로 설계하고 평소 상태는 High로 되게끔 S input에는 저항을, R은 SW를 연 상태를 설계합니다

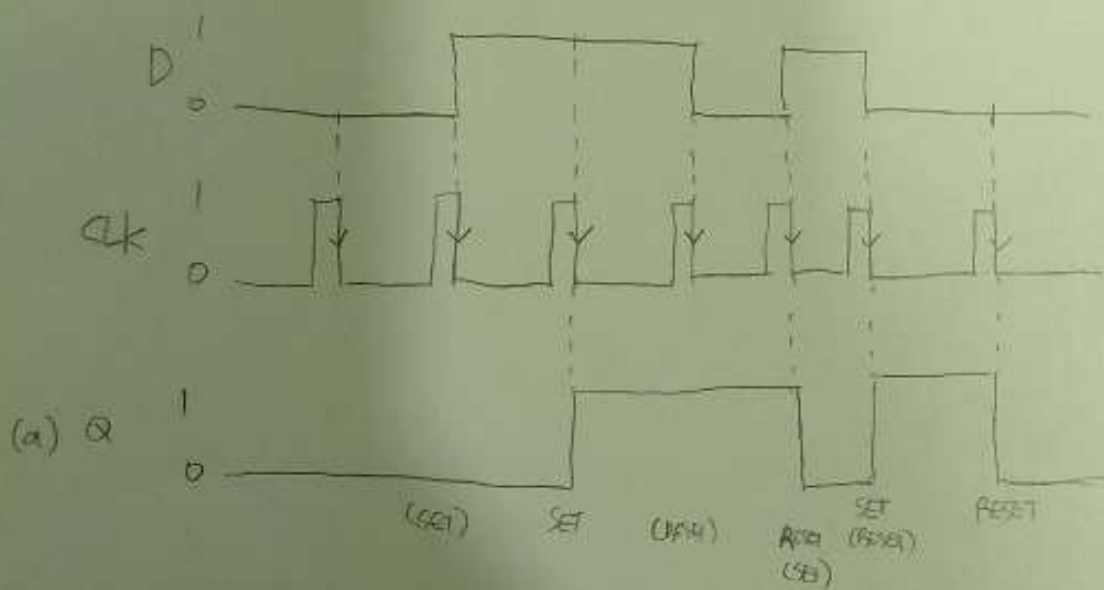
5-9)



5-13)



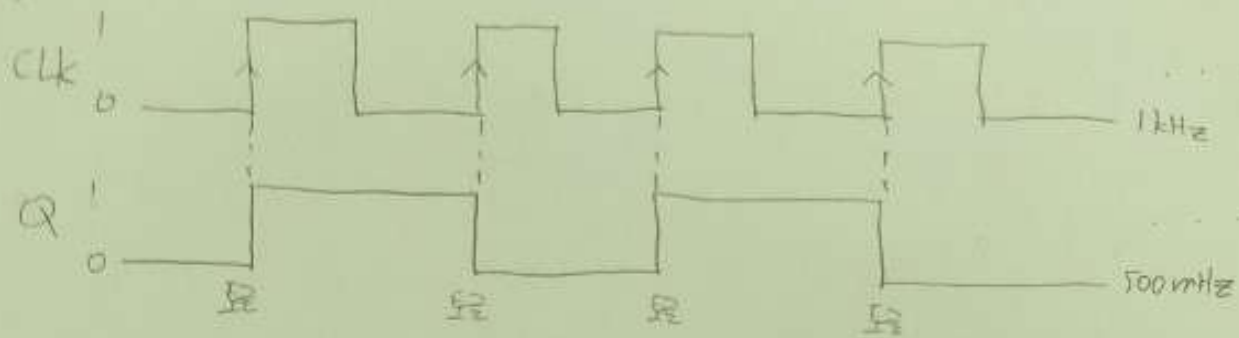
5-14)



\* ( ) 안의 내용은 입력데이터 상에서 SET 또는 RESET 이지만 제논시간에 의해  
바로 반영되지 않았음을 뜻한다

(b) 2개 클럭 주기만큼 지연을 얻기 위해서는 두번째 D-플립플롭을 만들고  
기준에 있던 5-12 의 플립플롭이 아니라 Q를 두번째 플립플롭의 2번째 D에  
연결한다. 클럭은 동일하게 연결한다. 이렇게 하면 2개 클럭 주기만큼 지연된다.

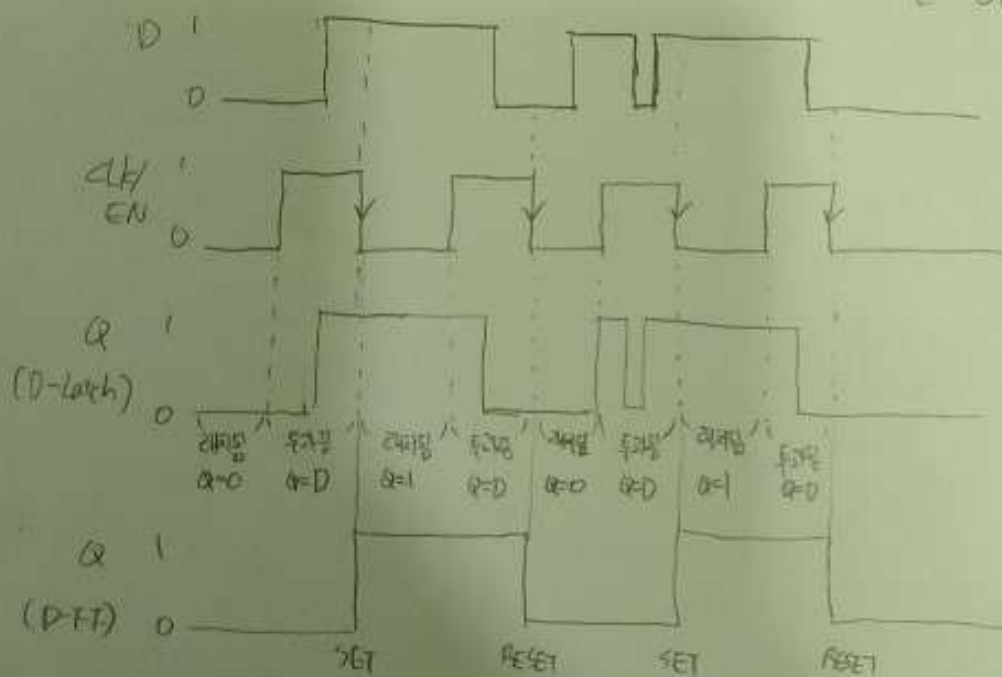
5-16)



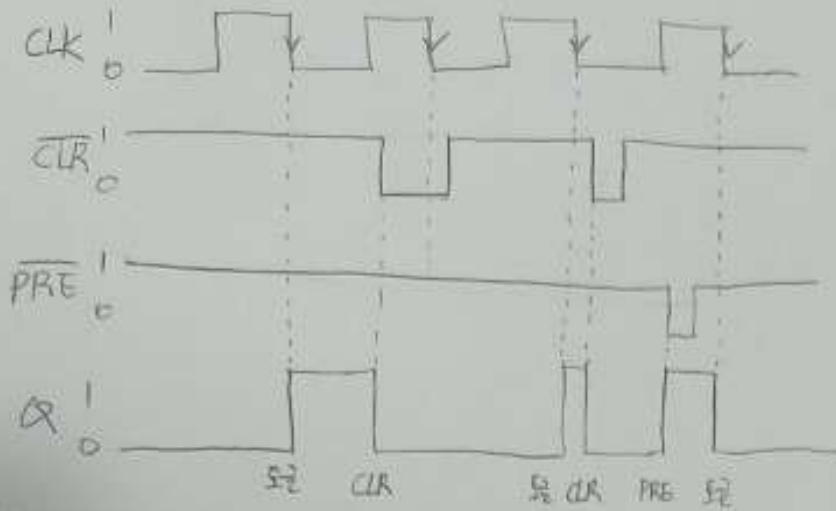
# clk가 PGT 일 때마다 Q는 0이 되어 Q값이 반대가 된다.

5-18)

1. D-Latch는 EN이 1일때만 D가 통과되어 Q=D 가 되고 아닐 때에는 Q가 0이 되어 Q값이 유지가 되고 D 입력에 영향을 받지 않는다.  
하반주기에는 D-FF는 NBT 일때 그 시점 D값이 Q값이 된다.



5-20)

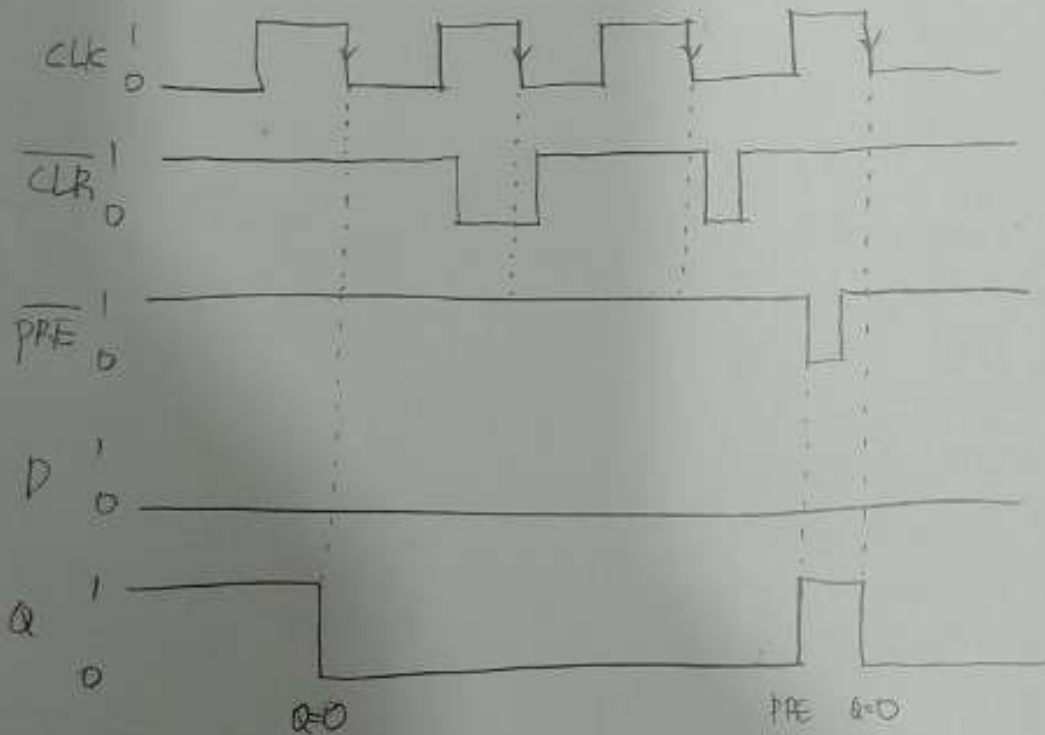


\*  $J, K = 1$  이므로  $\overline{PRE} = 1, \overline{CLR} = 1$  일때 NGT 발생시 1이 된다.

$\overline{PRE} = 0, \overline{CLR} = 1$  일때 클럭에 관계없이 0이 된다.

$\overline{PRE} = 1, \overline{CLR} = 0$  일때 클럭에 관계없이 0이 된다.

5-22)



\*  $D = 0$  이므로  $\overline{PRE} = 1, \overline{CLR} = 1$  일때 NGT 발생시 0이 된다.

$\overline{PRE} = 0, \overline{CLR} = 1$  일때 클럭에 관계없이 0이 된다.

$\overline{PRE} = 1, \overline{CLR} = 0$  일때 클럭에 관계없이 0이 된다.

5-26)

(a) Y가 HIGH 이라면 C.CLK가 PGT 일때 X가 HIGH 여야 한다

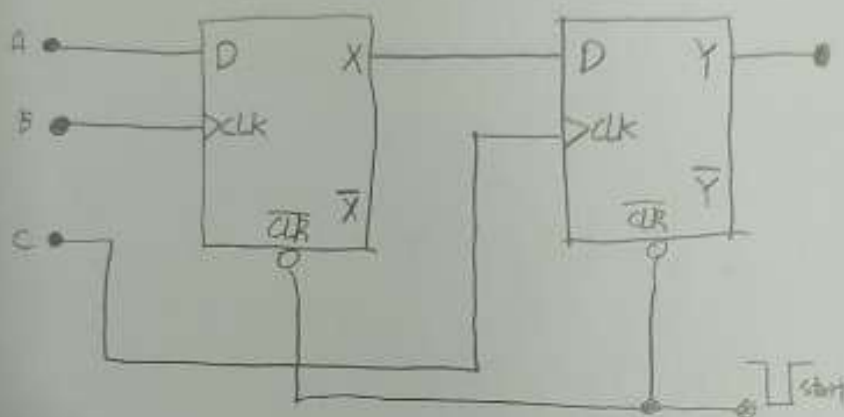
X가 HIGH 이라면 B.CLK가 PGT 일때 A가 HIGH 여야 한다

따라서 A가 HIGH로 변하지 되어 있고 B가 HIGH 여야 X가 HIGH가 되고, X가 HIGH 일때 C가 HIGH가 되어 Y가 HIGH가 되므로 순서는 A → B → C 이다.

(b) START pulse는 X와 Y를 0으로 초기화 시켜준다.

초기에는 X가 값이 결정되지 않은 상태에 때문에 초기화 되지 않으면 잘못된 결과를 얻을 수 있다.

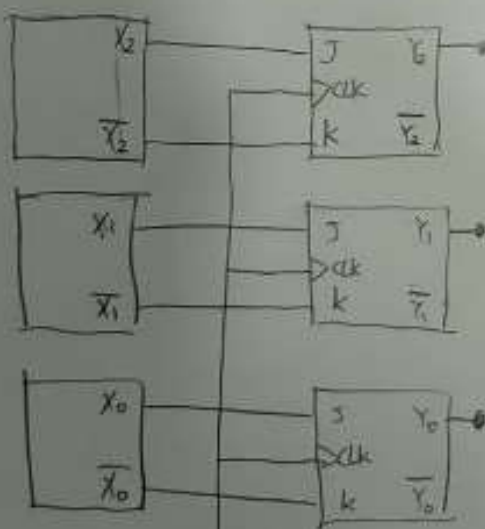
(c)



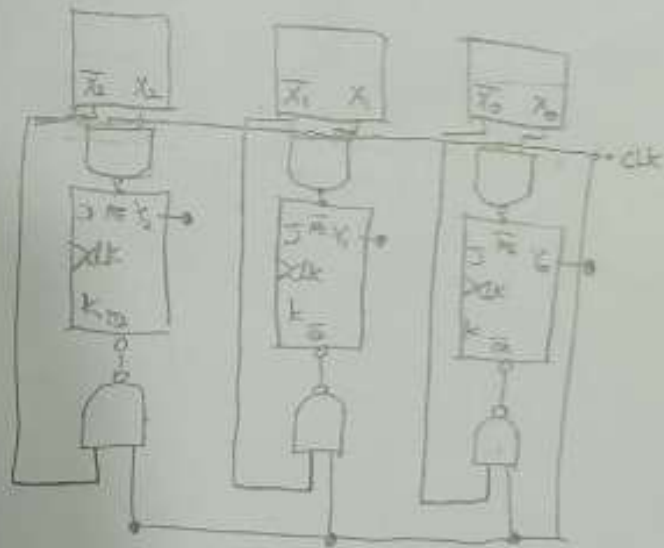
\* 기종 > K77에서 K가 값지에 연결되어 있으므로 J 입력에 따라 출력이 결정되는 것을 알 수 있다. A를 D 입력으로 바꾸면 구현할 수 있다.

5-27)

(a)



(b)



전송 CLK



5-33) (a) n개의 플립플롭은  $2^n - 1$  가지의 수를 나타낼 수 있다.

$511 = 2^n - 1$  이므로  $n = 9$ , ∴ 9개가 필요하다.

(b) 마지막 플립플롭의 출력 주파수는 입력클럭 주파수를 2로 나눈 것과 같다.

$$\frac{f}{2} (\text{마지막 FF}) = \frac{4\text{kHz}}{2^9} = \frac{4\text{kHz}}{512} = 7.8125\text{kHz}$$

(c) counter의 모든 수는  $0 \sim 511 \rightarrow 512$  가지의 수를 표현하기 때문에 ∴ 512.

(d) 이 카운터는 Mod-512 이다. 그러므로  $3500 \% 512 = 428$  (나머지) 이다 ∴ 428

5-34) (a) 이리각 FF의 출력 주파수는 입력클럭 주파수를 2로 나눈 것과 같다.

$$\frac{8.192\text{MHz}}{2^k} = 4\text{kHz} \rightarrow 2^k = \frac{8.192\text{MHz}}{4\text{kHz}} = 2048, k = 11$$

(b) Mod-4가 11 이므로 최대  $2^{11} - 1 = 2047$  ∴  $0 \sim 2047$  가지 수를 나타낸다.

5-35)

① 이 홀의 최대 500명의 손님이 들어올 수 있기 때문에  $2^9 \leq 500 \leq 2^{10} - 1$  이므로 최소 10개의 bit가 필요하다.

② 홀의 용량이 2배가 될 경우 최대 1000명이 입장할 수 있다.  $2^{10} \leq 1000 \leq 2^{11} - 1$  이므로 최소 11개의 bit가 필요하다.

③ 계속 적절한 카운트를 하기 위해서는 홀의 용량에 따라 'FF'의 수를 늘려야 한다.

5-37)

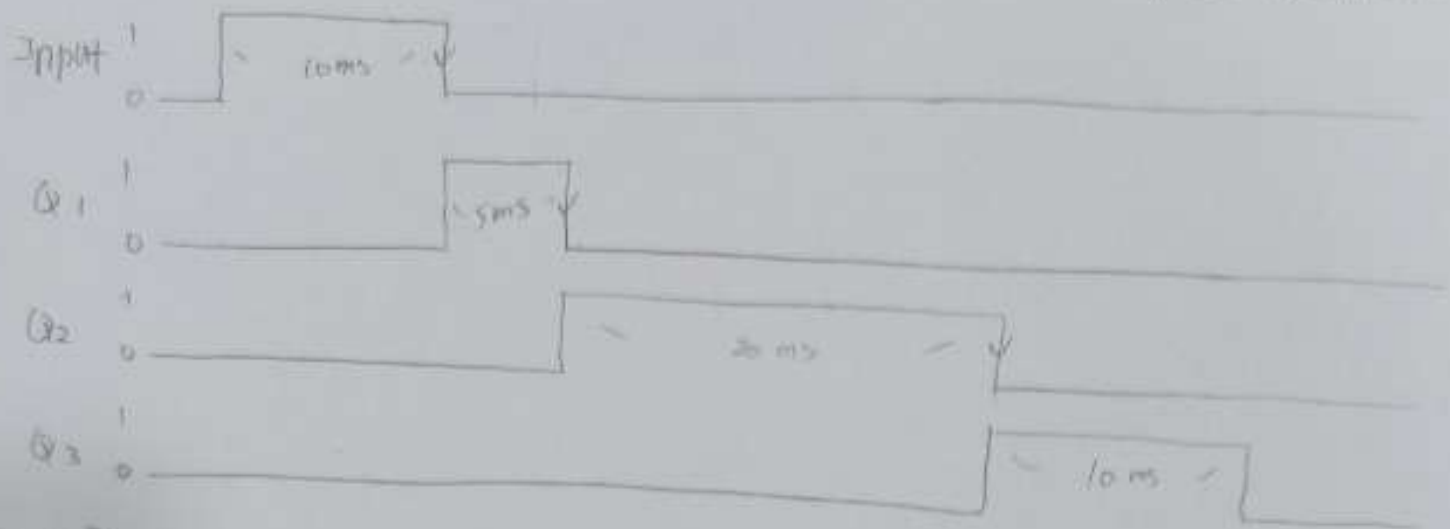
Interval

- short
  - ① A Inverter (Input) short to Gnd
  - ② A Inverter (Output) short to Vcc
  - ③ 1-A short to Vcc
  - ④ 1-output short to Vcc
- open
  - ⑤ 2-Input short to Vcc
  - ① A Inverter (Output) open
  - ② 1-output open
  - ③ 2-output open

Global

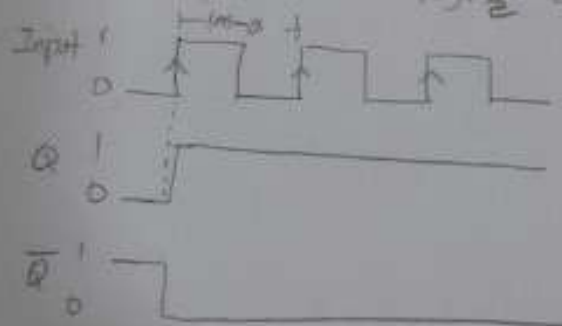
- short
  - ① A Inverter - 1 short to Vcc
  - ② 1-2 (gate) short to Vcc
  - ③ 2번 gate 입력 레이어 short to Vcc
- open
  - ① A Inverter - 1 open
  - ② 1-2 (gate) open
  - ③ 2번 gate 입력 레이어 open (4번 레이어 전)

5-41) 각 OS는  $t_p$  시간 동안 잠입정 상태로 유지시키고 다시 안정 상태로 되돌아간다.



5-42)

(a) 입력 주파수가 1kHz 이상이면 Input의 주기는  $\frac{1}{1\text{kHz}} = 1\text{ms}$ 보다 작아지게 된다. 그러면 Q는  $t_p$  시간 안에 항상 재트리버 되므로 Input의 첫 PGT 이후 High를 유지한다.



Q가 클럭으로 들어갔는데 PGT 없이 계속 Low 이므로 클럭은 Low가 된다.

(b) 입력 주파수가 1kHz 이하이면 Input의 주기는  $\frac{1}{1\text{kHz}} = 1\text{ms}$ 보다 크게 된다.



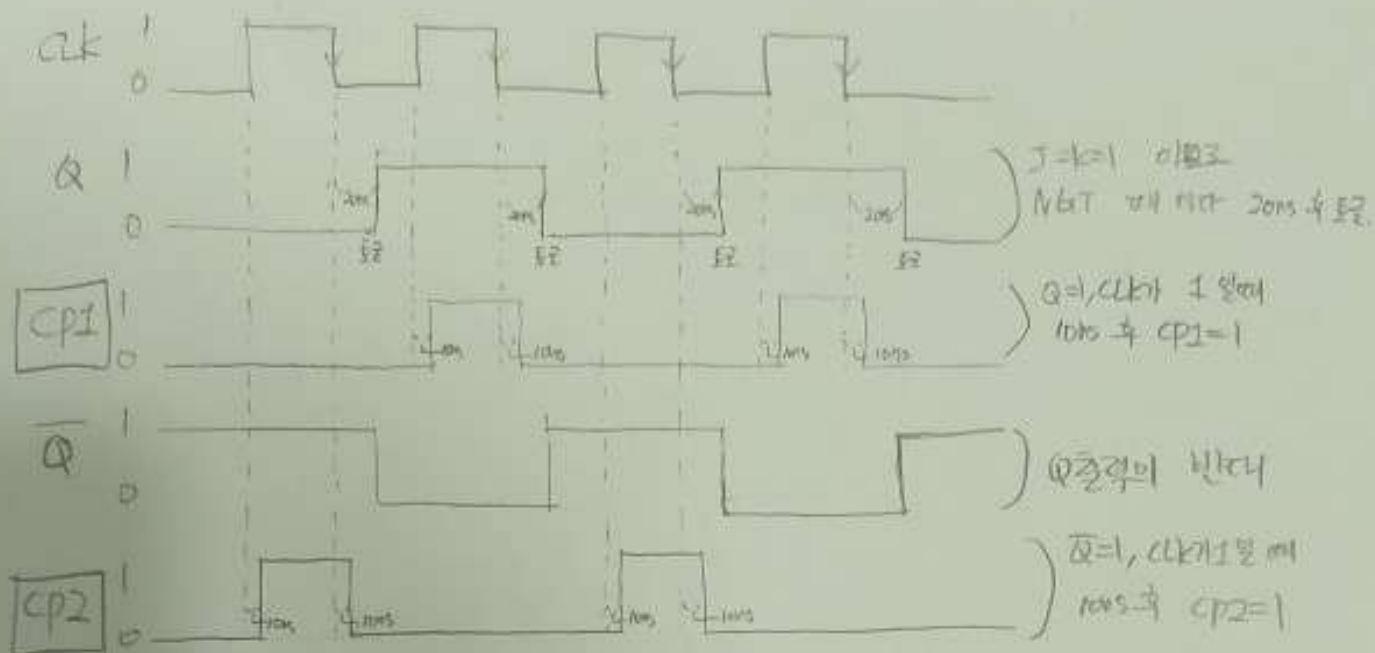
Q가 클럭으로 들어갔는데 PGT가 발생하므로 50kHz 이므로 클럭은 High가 된다.

(c) 입력 주파수 50kHz 일때 주기는  $\frac{1}{50\text{kHz}} = 20\mu\text{s}$  이므로  $t_p$  시간인 20ns 보다 OS 4정리만된다.

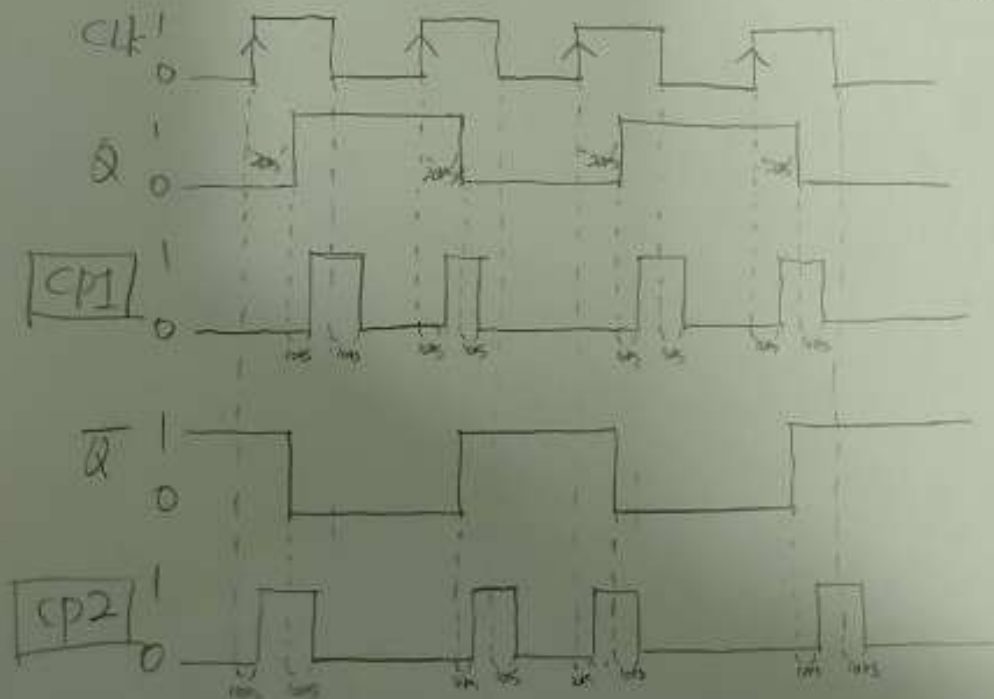


5-49)

(a) FF:  $t_{puH} = t_{pHL} = 20\text{ns}$ , AND:  $t_{puH} = t_{pHL} = 20\text{ns}$ , NOT



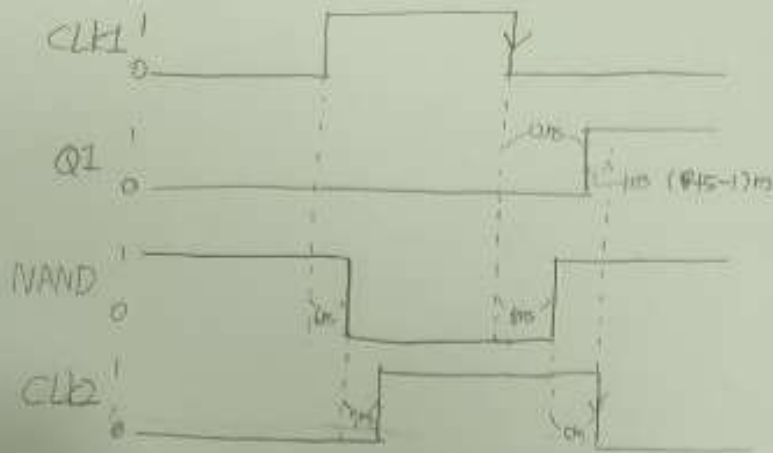
(b) FF, AND 게이트에서  $t_{puH}, t_{pHL}$  시간 60ns와 동일, NOT



\* 이 회로는 문제점을 가진다. CP1과 CP2가 겹치지 않아야 하는데  
같은 부분이 생기기 때문이다.

5-51)

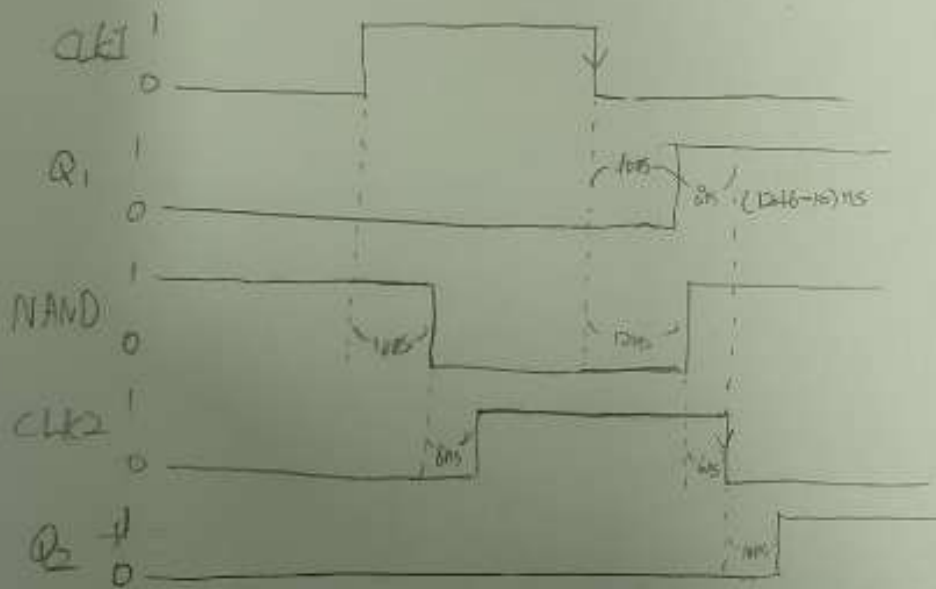
① (a), (b), (c) 조건 일 때



\* 조건 (a)의  $t_s = 5ns$  일 때 위 그림을 보면 1ns로 주어졌었다

두번째 클럭(CLK2)이 발생했을 때 적힌 5ns 동안 값이 흔들리기 때문에 값을 측정할 수 없다! Setup time 위반

② (d), (e), (f) 조건 일 때



\* 조건 (b)의  $t_s = 5ns$  일 때 위 그림을 보면 8ns로 주어졌었다

그러므로 Setup time을 위반하지 않는다 (오류는 안남)

하지만 정상적인 시프트 레지스터 작동하기 위해서는  $Q_2$ 가 Low로 유지되어야 한다  
그러기 위해서는  $Q_1$ 과  $Q_2$  사이에  $\Delta$ 를 달아 최소 8ns 만큼 더 밀어야 한다.

## 5-54) 4시

1. RESET 스위치를 잠깐 활성화시켜 CLR을 0으로 만들어 FF의 동작을 할 준비를 해 놓는다.
  2. ENTER 스위치는 NO일때 HIGH, NC일때 LOW를 출력한다.  
Q1 출력하는 FF에 SWA open, SWB closed, SWC open 상태  
ENTER 스위치를 압력으로 푸는 시기 PGT를 발생하여 Q1이 HIGH를 출력하게 한다.
  3. 이번엔 Q2 출력하는 FF에 D를 HIGH 만들기 위해 Q1이 HIGH가 되었으므로 SWA closed, SWB open, SWC closed 상태로 ENTER 스위치를 압력으로 푸는 시기 PGT를 발생하여 Q2를 HIGH로 출력하게 한다.
- : 자문외가 완료다