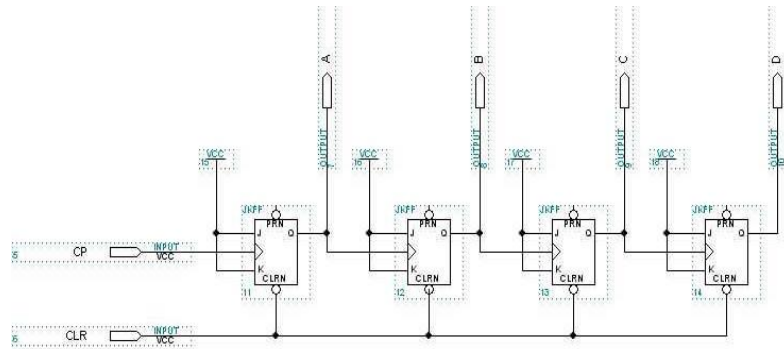
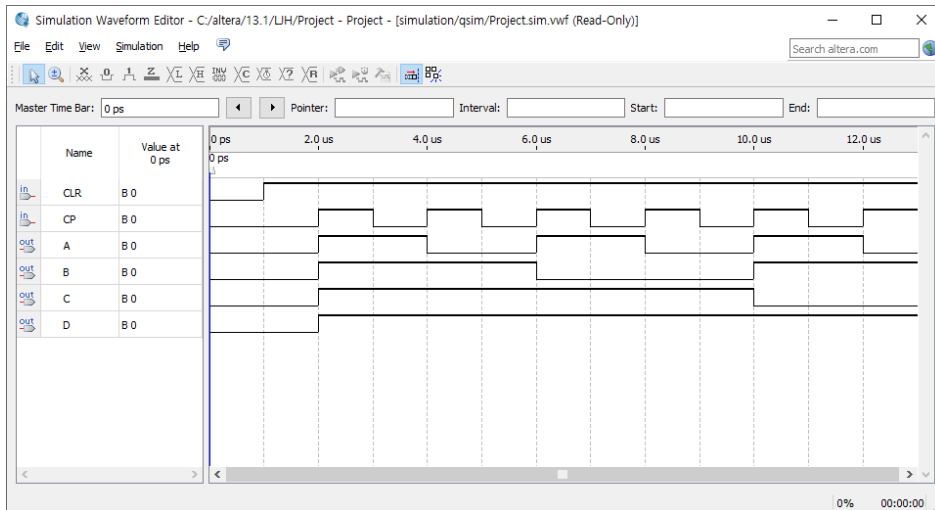


7476 IC 핀 배치도를 참조하여 아래 그림과 같은 비동기식 카운터 회로를 구성한다. 여기서 7476은 2개를 사용한다. 7476의 13번 핀은 접지하고, 5번 핀은 +5V의 전압을 인가한다. CLR을 Low에서 High로 하여 모든 플립플롭의 출력을 0으로 초기화한다. 클록펄스(CP)를 순차적으로 인가하면서 출력 A, B, C, D의 상태를 표에 기록하여라. 또 그 결과를 타이밍도로 그려 보아라.



CP	D	C	B	A	CP	D	C	B	A
1	1	1	1	1	9	0	1	1	1
2	1	1	1	0	10	0	1	1	0
3	1	1	0	1	11	0	1	0	1
4	1	1	0	0	12	0	1	0	0
5	1	0	1	1	13	0	0	1	1
6	1	0	1	0	14	0	1	1	0
7	1	0	0	1	15	0	0	0	1
8	1	0	0	0	16	0	0	0	0

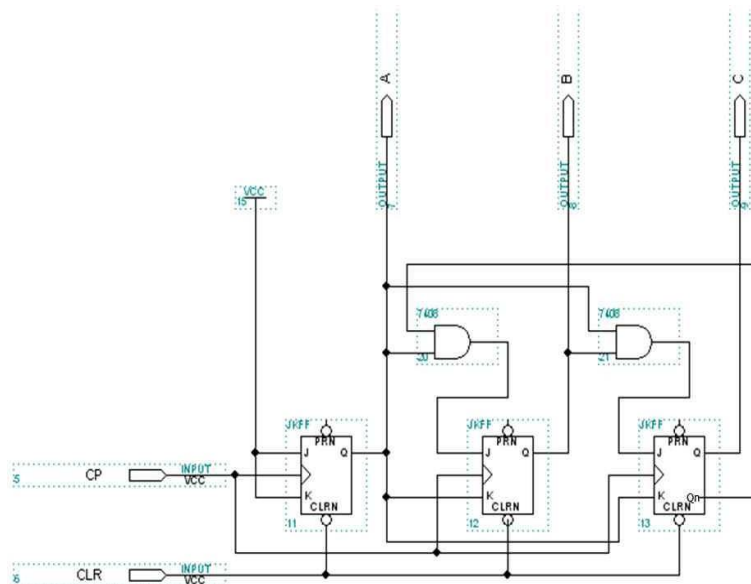


[검토] 이 카운터의 모듈러스(modulus)는 얼마인가? 더불어 이 카운터는 상향 카운터인지, 하향 카운터인지 검토하여라.

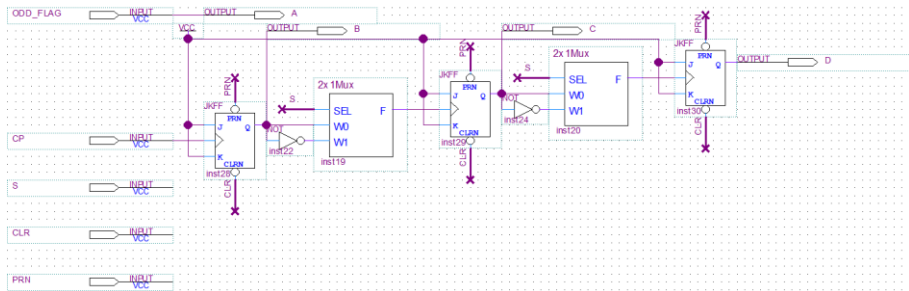
총 16개의 상태를 가지는 카운터이므로 모듈러스는 16이다. 16에서 시작하여 클럭펄스 주기에 맞춰 1씩 떨어지는 카운터이므로 하향 카운터이다.

## 2) 동기식 카운터

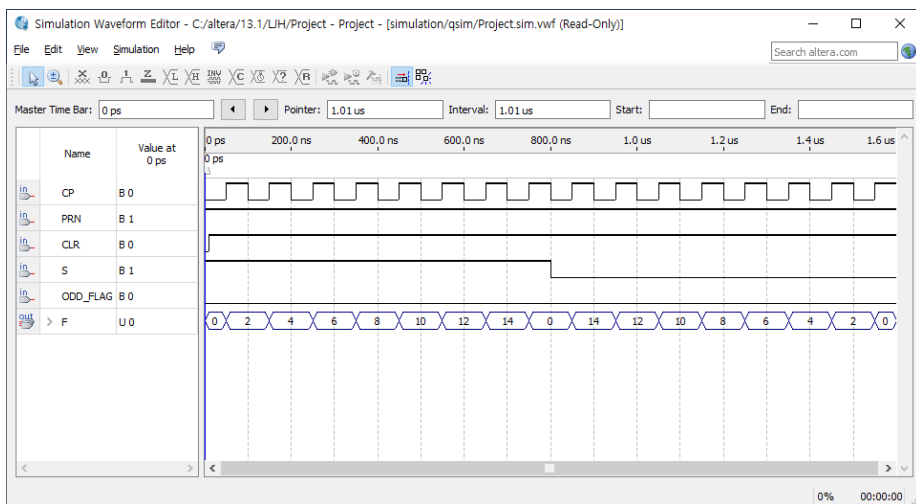
7476과 7408 IC 핀 배치도를 참조하여 아래 그림과 같은 동기식 카운터 회로를 구성한다. 여기서 7476은 2개를, 7408은 1개를 사용한다. 7476의 13번 핀은 접지하고, 5번 핀은 +5V의 전압을 인가한다. 또 7408의 7번 핀은 접지하고, 14번 핀은 +5V의 전압을 인가한다. CLR을 Low에서 High로 하고, 클럭펄스(CP)를 순차적으로 인가하면서 출력 A, B, C의 상태를 표에 기록하여라. 또 그 결과를 타이밍도로 그려보아라



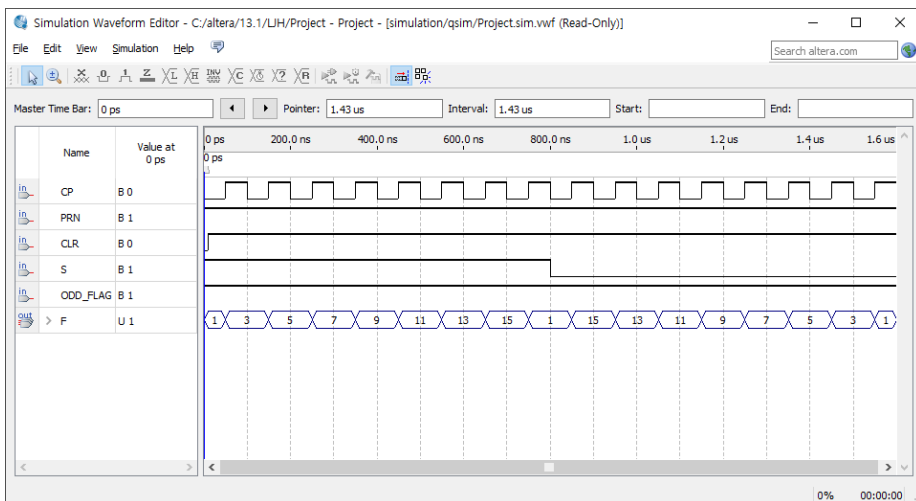




ODD\_FLAG: 홀수 카운터로 작동할지 짝수 카운터로 작동할지 결정  
 S: 1일 때 상향 카운터, 0일 때 하향 카운터로 작동

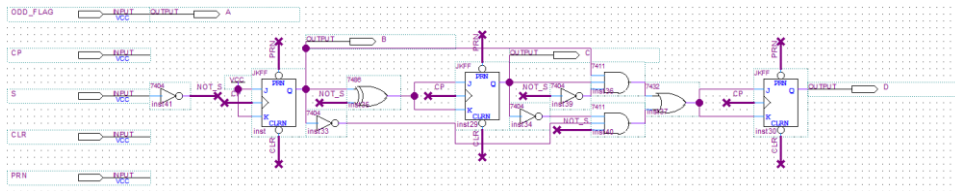


<ODD\_FLAG=0 (짝수), 시뮬레이션>



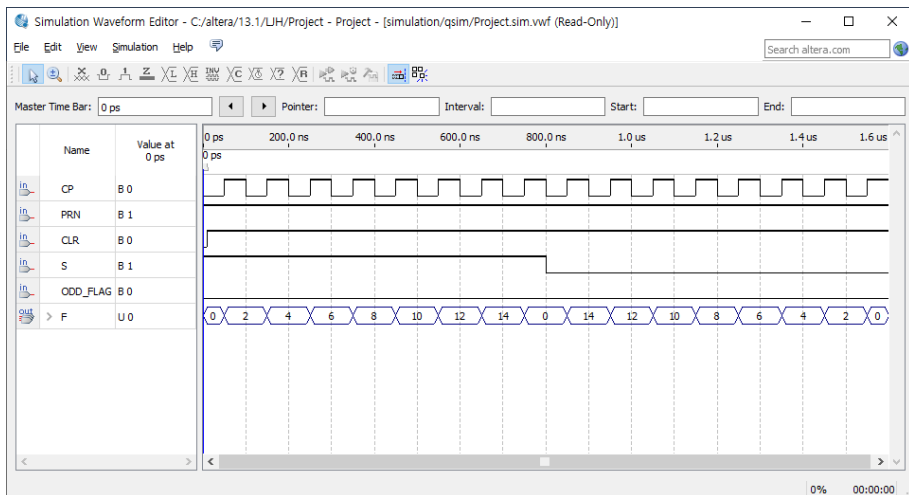
<ODD\_FLAG=1 (홀수), 시뮬레이션>

## 2. 4비트 동기 상향/하향 짝수/홀수 카운터

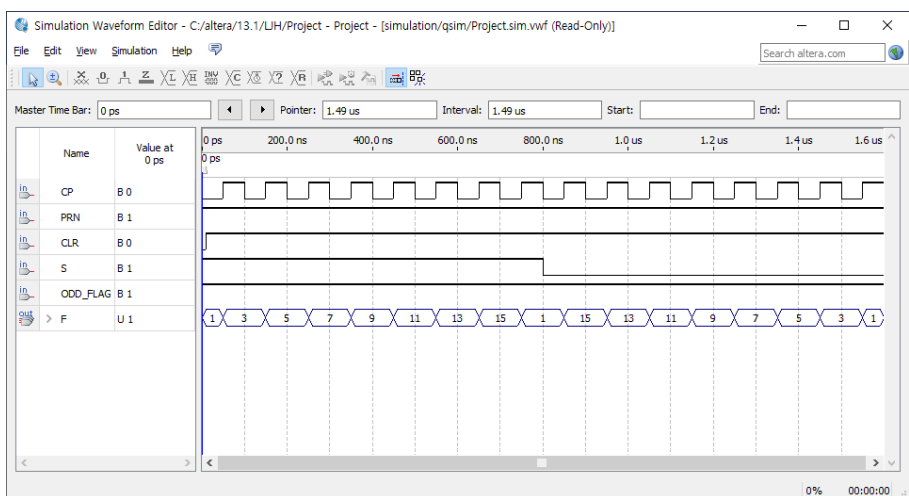


ODD\_FLAG: 홀수 카운터로 작동할지 짝수 카운터로 작동할지 결정

S: 1일 때 상향 카운터, 0일 때 하향 카운터로 작동

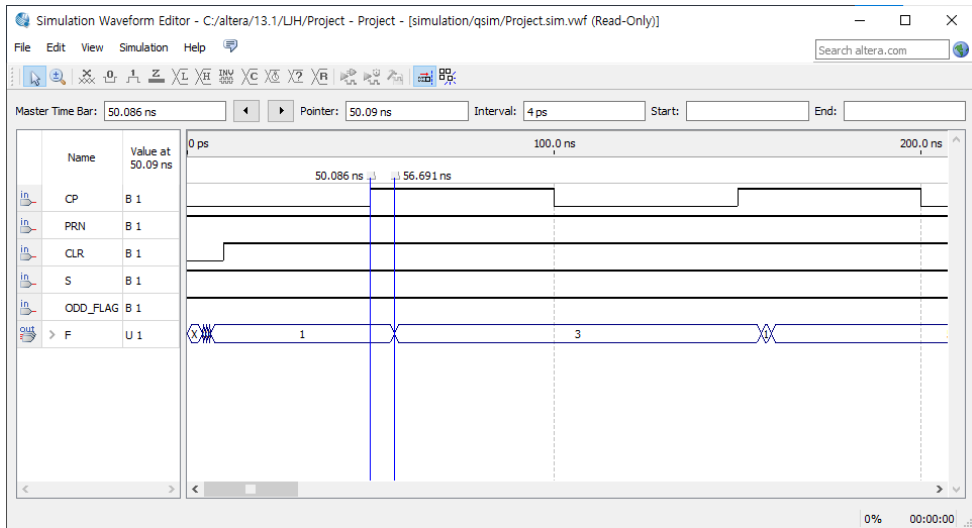


<ODD\_FLAG=0 (짝수), 시뮬레이션>

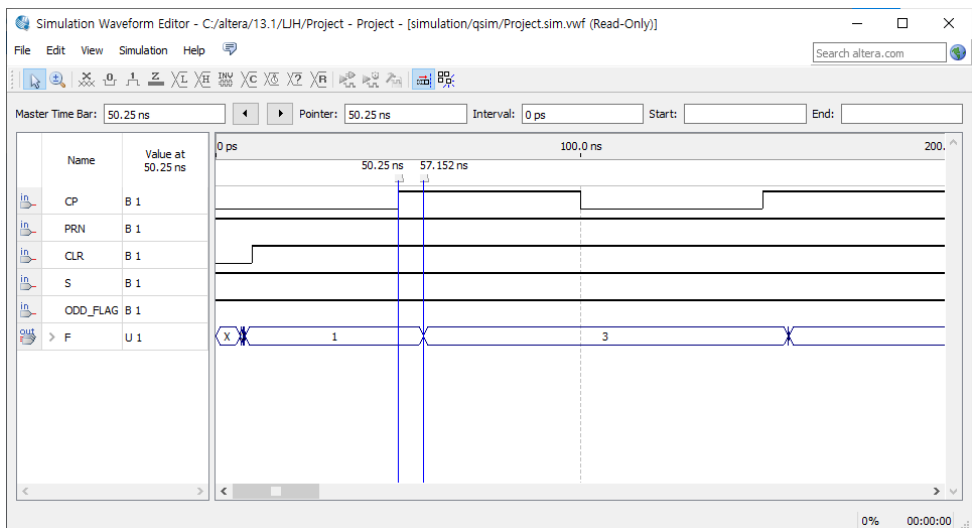


<ODD\_FLAG=1 (홀수), 시뮬레이션>

## 3. 동기과 비동기 카운터 비교 (Timing simulation)



<4비트 비동기 상향/하향 짝수/홀수 카운터 (6.605ns)>



<4비트 동기 상향/하향 짝수/홀수 카운터 (6.902ns)>

비동기식 카운터는 동기식 카운터에 비해 회로가 간단해 진다는 장점이 있으나 전달 지연이 커진다는 단점이 있다. 왜냐하면 이전 플립플롭의 출력에 다음 플립플롭의 클럭펄스가 의존하기 때문이다. Quartus 내부의 최적화 작업으로 인해 시뮬레이션 결과에서 유의미한 지연 차이를 소프트웨어 시뮬레이션으로 확인하기는 어려웠다. 하지만 실제 기판에 동일한 회로로 실험해보면 유의미한 전달 지연을 관찰할 수 있을 것으로 생각된다.

### <느낀 점>

다양한 모듈러스를 가지는 비동기/동기 카운터를 설계해보면서 조금 더 회로에 대해 자세히 이해할 수 있는 시간이 되었다.