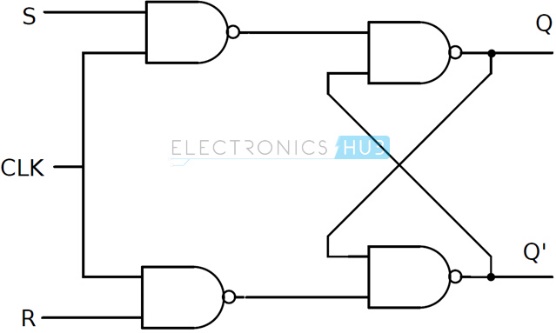
11주차 결과보고서

전공: 아트엔테크놀로지 학년: 3학년 학번: 20191172 이름: 함승우

1. RS Flip-Flop의 결과 및 Simulation 과정

우선, Flip Flop은 1비트의 정보를 기억할 수 있는 최소의 기억 소자이다. 그 중 RS 플립플롭은 Reset 단자와 Set 단자 두 가지를 가지고 있고, Flip Flop의 특성상 Clock을 가지고 있다. 따라서 Reset, Set, Clock 세 가지의 입력을 갖고, q의 원래 상태와(이전 상태), nextq(이후 상태)를 출력 값으로 갖는다.

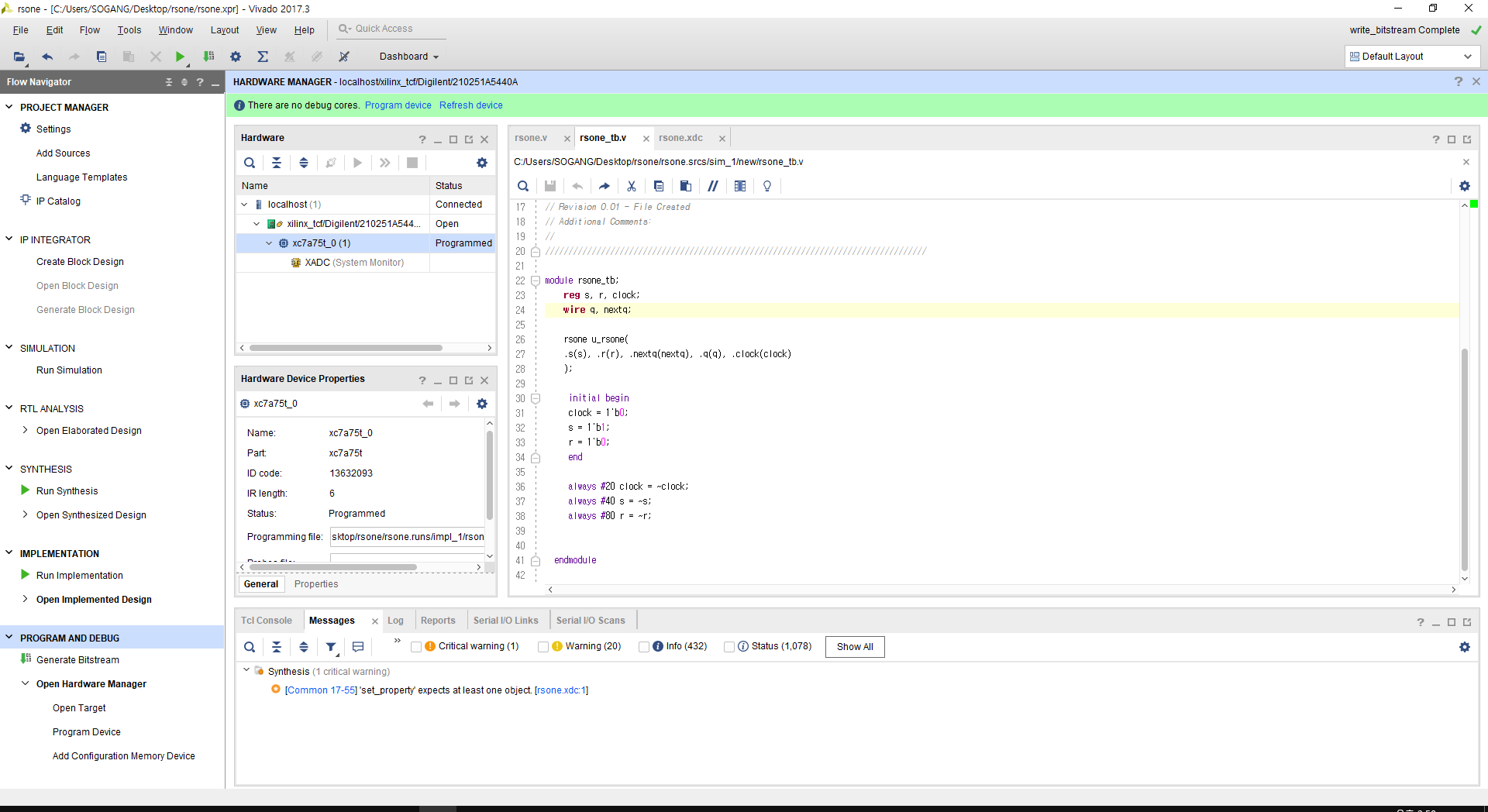
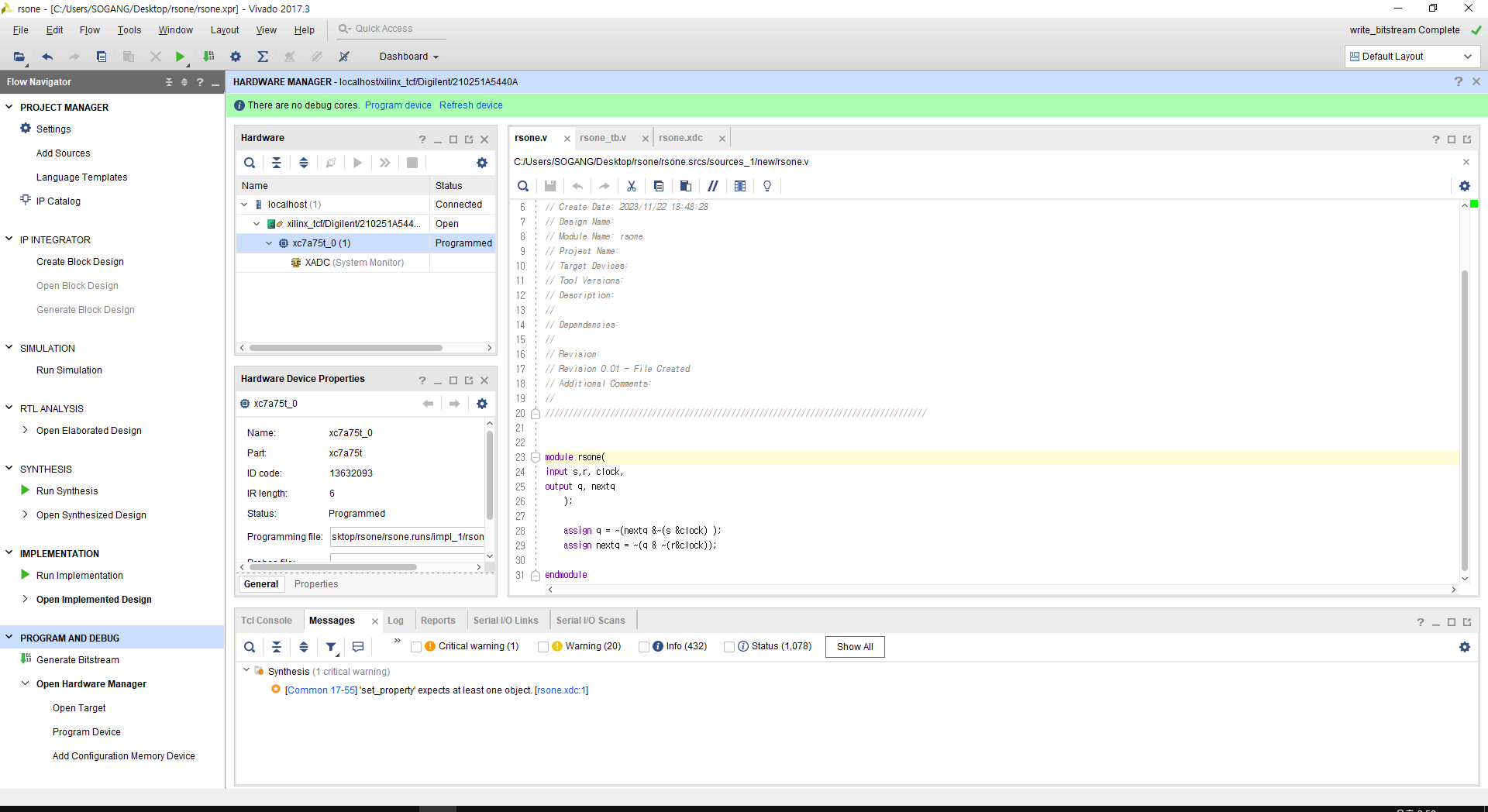
* NAND gate 사용



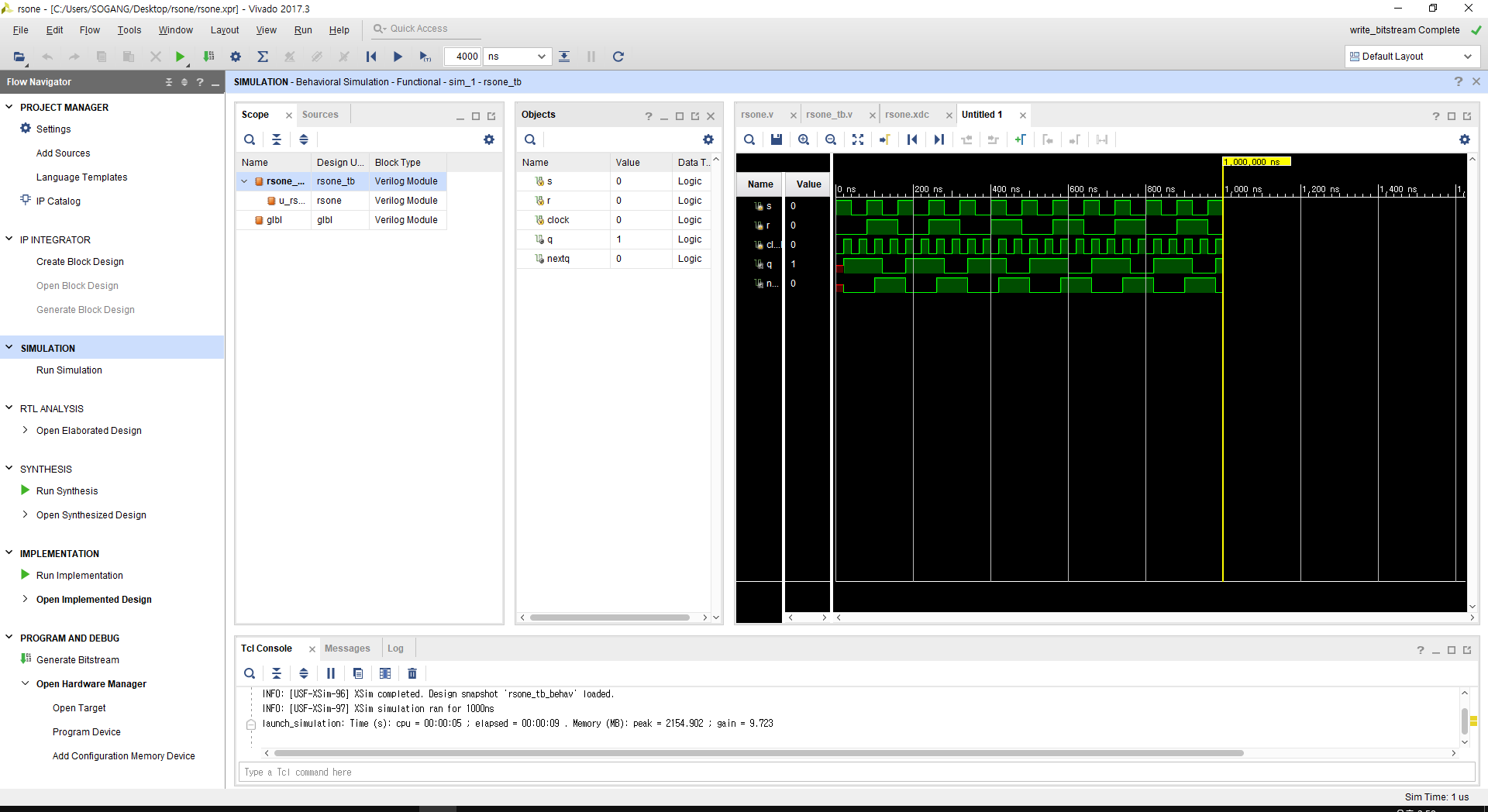
위 논리도로 식을 세워 보자면

Q = ~(Q(t+1) & ~(S & clk))

Q(t+1) = ~(Q(t) & ~(R & clk)이다.

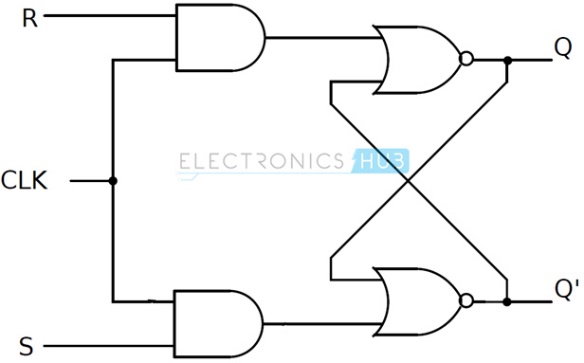


-Simulation



q와 nextq의 앞 부분은 빨간색으로 나타나는데, 이는 설정되지 않아서 나타난다. clock과 q와 nextq의 관계를 살펴보면, clock이 0에서 1이될 때나 1에서 0이 될 때 q나 nextq가 변하는 것을 알 수 있다. 또한, s이 0이고 r이 0인 경우를 잘 살펴보면, q와 nextq의 차이가 없어지는 것을 확인 할 수 있고, s가 1이고, r이 0인 경우에는, q값에 상관없이 nextq가 1, s가 0이고, r이 1인 경우에, nextq가 0이되는 것을 확인 할 수 있다.

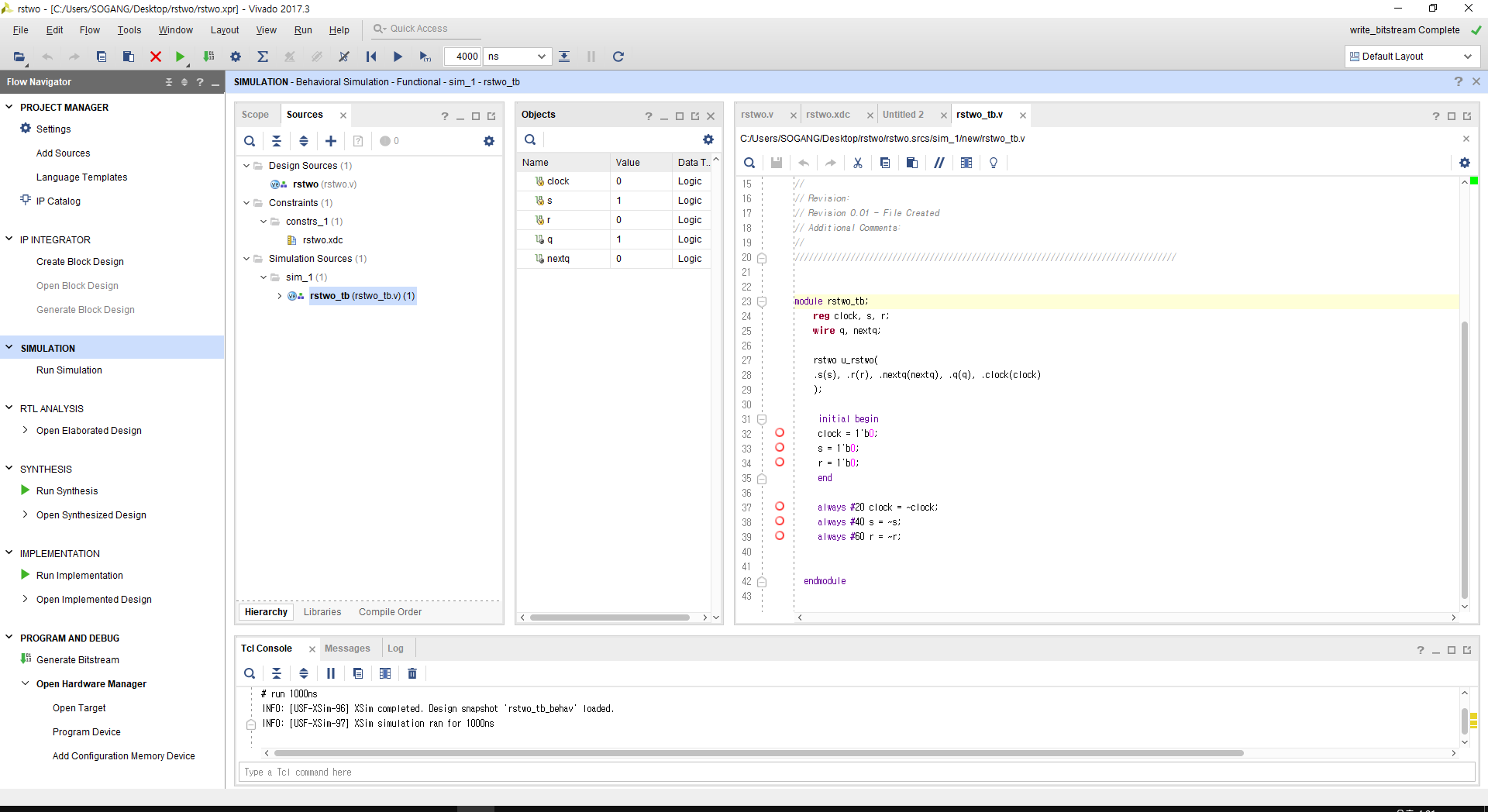
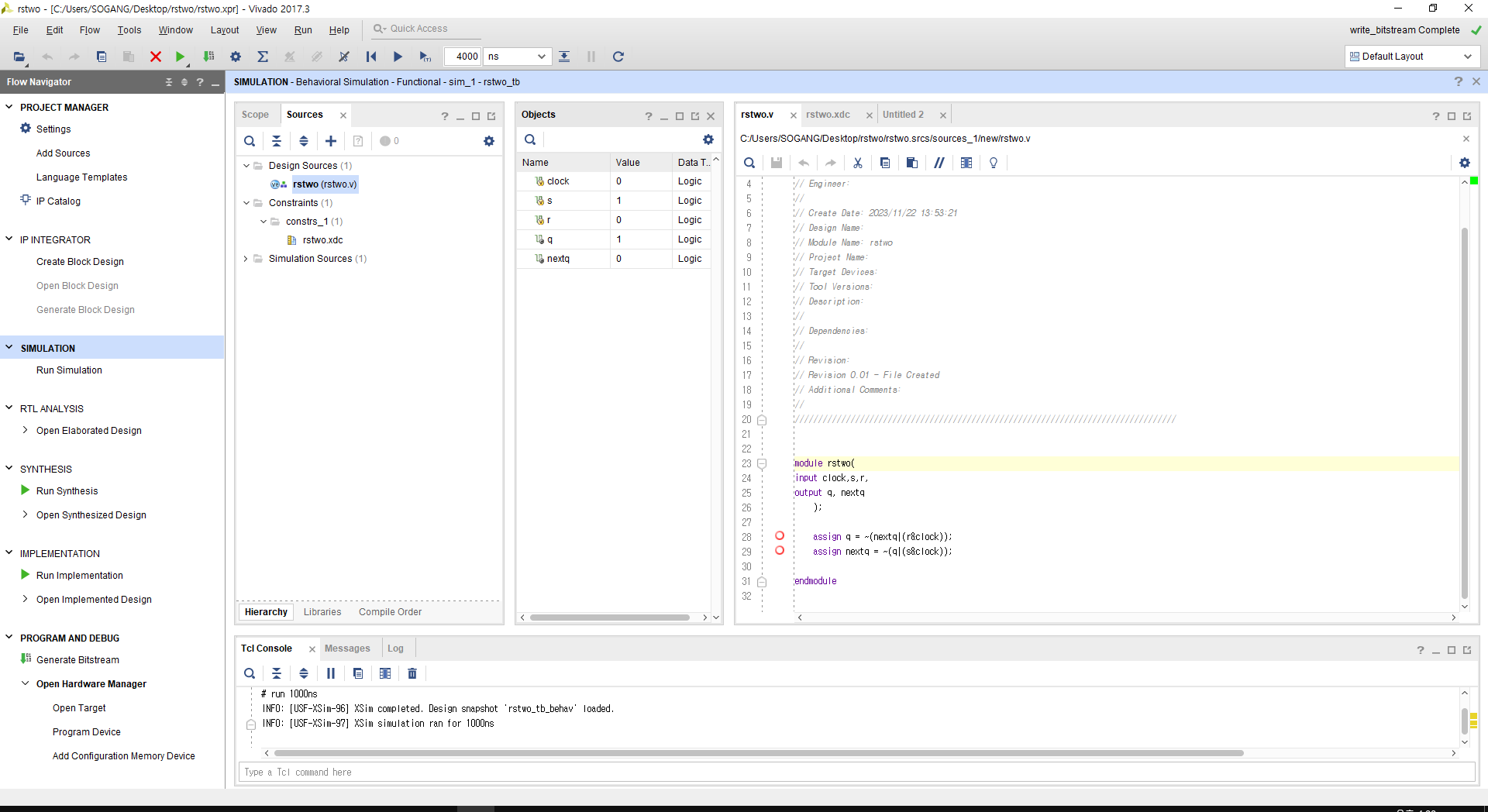
* NOR gate 사용



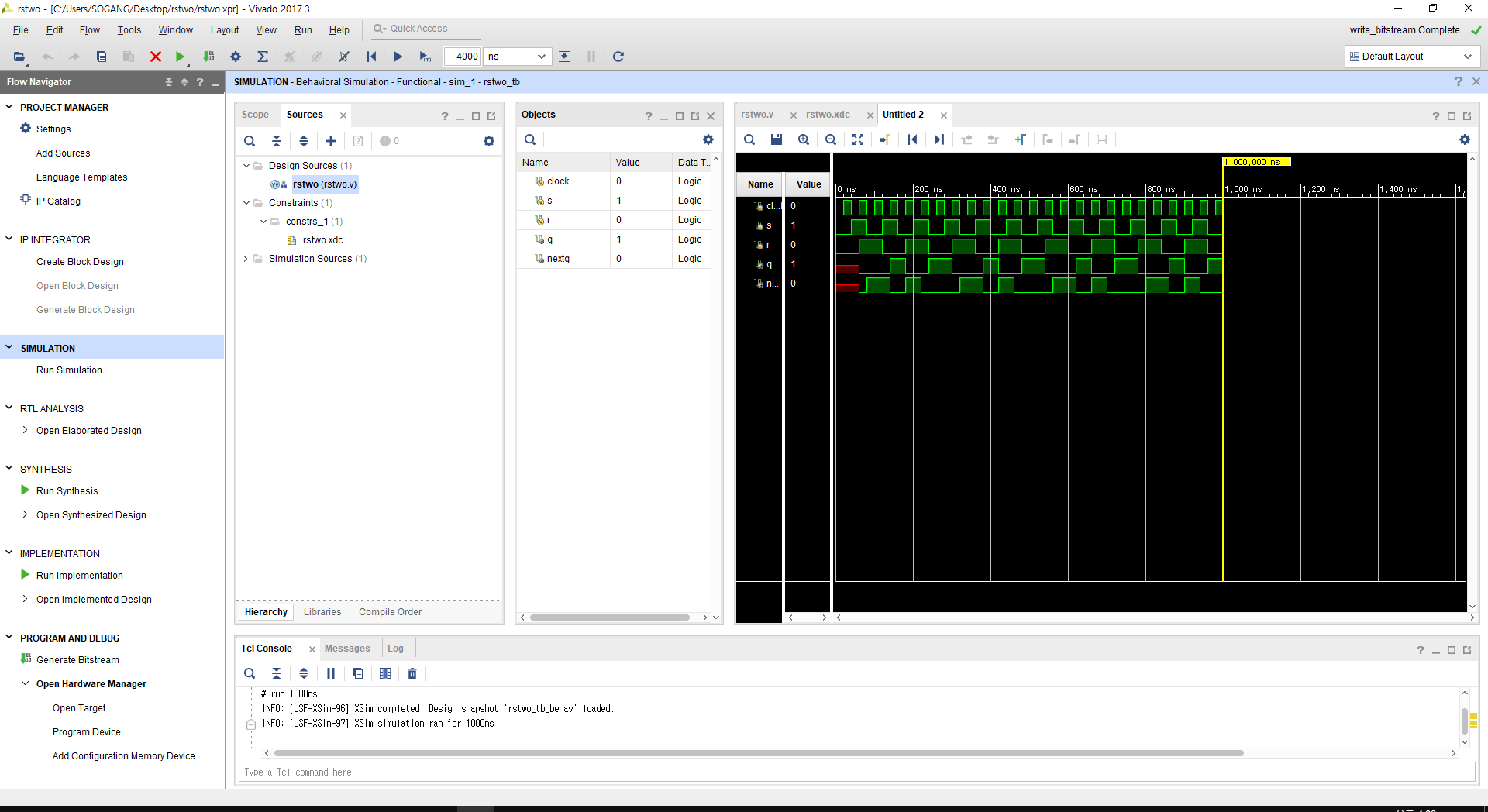
논리도에 따라서,

Q(t) = ~(Q(t+1) | (R & clk))

Q(t+1) = ~(Q(t) ! (S & clk))



* Simulation



tb 파일에서 initial begin 부분이 위 NAND 부분과는 다르가 때문에 앞 부분과 순서에는 차이가 있다. 하지만, 결과는 위 Simulation과 동일하기 때문에 설명은 생략하겠다.

다음은, RS flip flop의 truth table과 excitation table을 작성하면,

|  |  |  |  |
| --- | --- | --- | --- |
| S | R | Q(t) | Q(t+1) |
| 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 |
| 1 | 0 | 0 | 1 |
| 1 | 1 | 0 |  |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 1 |  |

S와 R이 모두 0인경우에는 이전 상태를 유지하고, S가 1일 때 무조건 이후 상태는 1이된다. R이 1이 될 때에는, 무조건 이후 상태는 0이 되고, S와 R이 모두 1일 때는 작동하지 않는다.

아래는 excitation table이다.

|  |  |  |  |
| --- | --- | --- | --- |
| Q(t) | Q(t+1) | S | R |
| 0 | 0 | 0 | X |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 |
| 1 | 1 | X | 0 |

위 이전상태(Q(t))와 이후상태(Q(t+1))의 값을 통해서 S와 R 값을 채운다고 생각하면 된다.

이전 상태와, 이후 상태가 모두 0 0인 경우에는, 상태 불변일 수도 있고, Reset이 켜 있을 수도 있다. 따라서, S는 0이지만, R은 0과 1 모두 될 수 있다. 이전상태가 0, 이후상태가 1인 경우에는, Set이 켜 있는 경우밖에 없다. 따라서, S는 1, R은 0이다. 마찬가지로, 이전 상태는 1, 이후상태는 0일 때, Reset이 켜 있는 상황밖에 없기 때문에, S는 0, R은 1이다. 마지막으로, 이전 상태가 1, 이후 상태가 1인 경우에는, Set일 수 있고, 상태 불변일 수 있기 때문에, Set은 0과 1 모두 될 수 있고, R은 0이다.

위 truth table을 통해서 Q(t+1)의 Equation을 구해보자면,

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SR  Q(t) | 00 | 01 | 11 | 10 |
| 0 |  |  | X | 1 |
| 1 | 1 |  | X | 1 |

따라서 Q(t+1) = nextq = S + QR’이다.

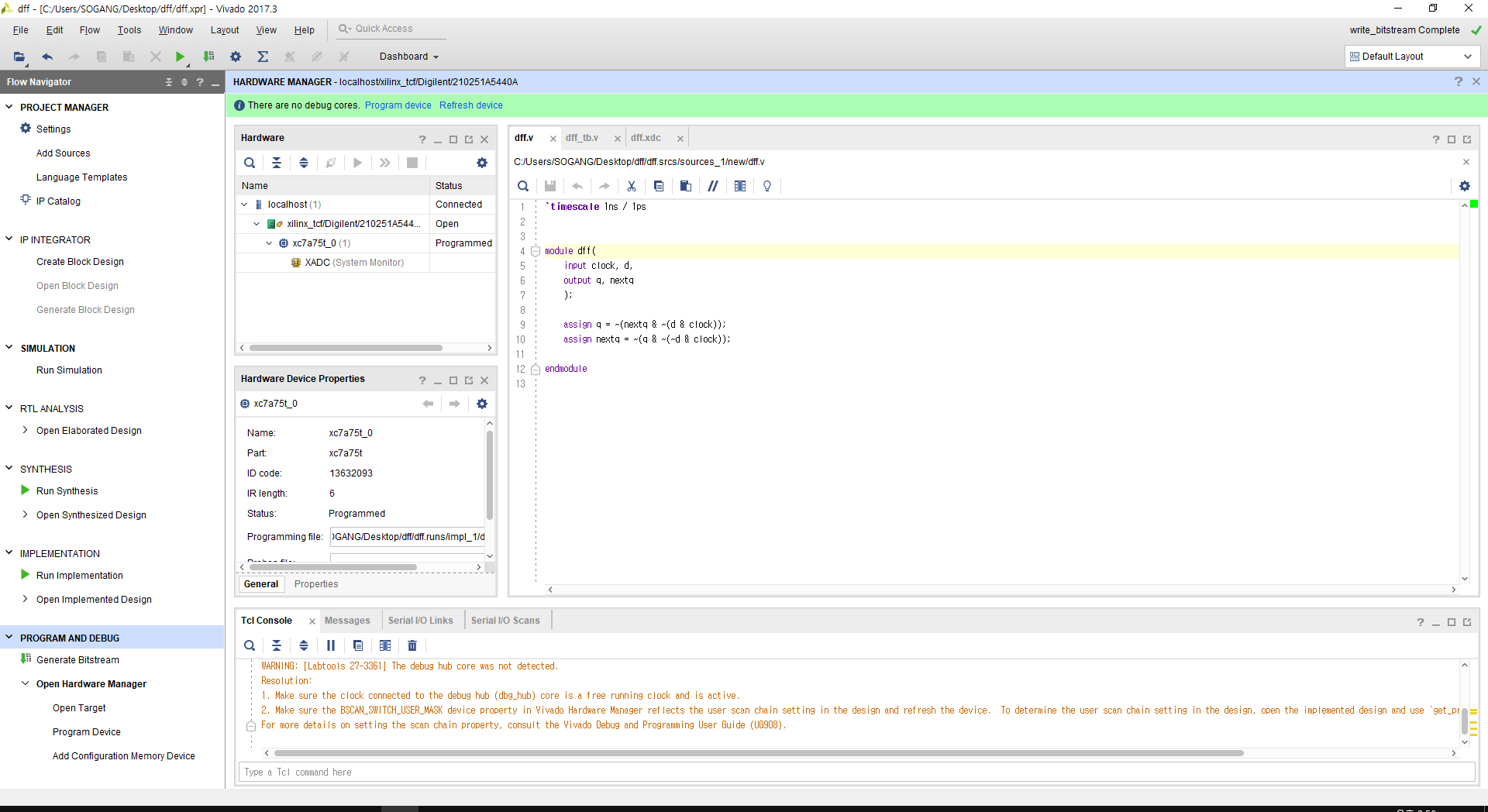
2. D Flip-Flop의 결과 및 Simulation 과정

도표, 스케치, 라인, 그림이(가) 표시된 사진

자동 생성된 설명

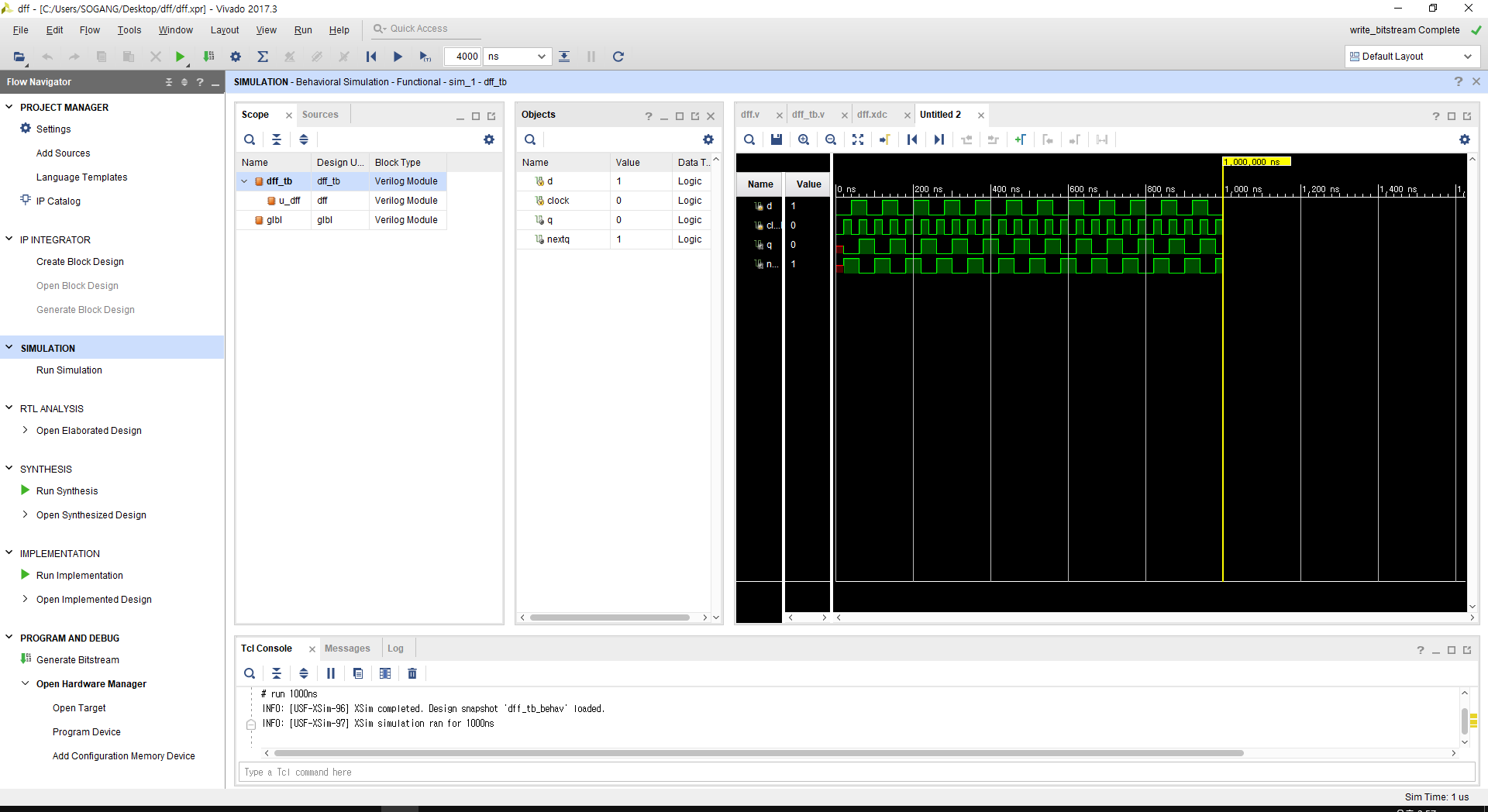
D flip flop은 input 값이 clock을 제외하고 하나라는 것에서 RS flip flop과 차이가 있다. 하지만, 구조를 보면 비슷함을 알 수 있다. D flip flop은 입력받은 값을 Delay를 갖게 한 뒤, 그대로 출력하는 flip flop이다. RS flip flop에 inverter를 추가한 구조로, 위 논리도를 통해서 논리식을 세울 수 있다.

* Verilog 코딩

텍스트, 스크린샷, 디스플레이, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

* Simulation



맨 처음에 앞의 예시들과 동일하게 clock값이 0이 부분에서는 q와 nextq가 동일하게 unknown으로 나타난다. 이후, clock이 1 될 때를 설명하겠다. 이 때 d 값이 1이 된다면, nextq값은 1을 갖게 된다. 또한 d 값이 0이라면, nextq값이 0을 갖게 되는 것을 확인할 수 있다. 즉, d값과 nextq값은 동일하게, clock이 켜져 있을 때 q는 항상 nextq값을 갖는 다는 것을 확인할 수 있다. 또한, q값과 nextq는 동일한 조건에서 항상 반대의 값을 갖는다.

D flip flop의 truth table을 작성하자면,

|  |  |  |  |
| --- | --- | --- | --- |
| **clock** | **D** | **q** | **nextq** |
| 0 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 |

이 truth table을 통해서 excitation table을 작성하자면,

|  |  |  |
| --- | --- | --- |
| q(t) | q(t+1) | D |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

즉, Q(t+1) = D(t)임을 알 수 있다.

3. 결과 검토 및 논의 사항.

RS flip flop과 D flip flop을 구현해보았는데, 3가지 simulation에서는 q와 nextq가 설정되지 않아 unknown으로 나타나는 경우가 모두 발생했다. 또한, flip flop의 논리식을 설정하는 데 있어서 어려움이 있었지만, 논리도를 기반으로 논리식을 세울 수 있었다. Clock이 flip flop에서의 역할을 분석하고, latch와 다르게 0에서 1, 1에서 0으로 변화할 때에 달라지는 점을 확인할 수 있었다.

4. 추가 이론 조사 및 작성.

플립플롭이 정상적으로 동작하기 위해서는 플립플롭의 data 입력은 clock 활성 전후로 일정 기간 동안 그 값을 유지해야 한다. Set – up time은 활성 엣지 직전 data 값을 분변으로 유지해야 하는 최소 시간을 의미하고, Hold time은 활성 엣지 이후 data 값을 불변으로 유지해야 하는 최소 시간을 의미한다. 쉽게 이야기하자면, setup time은 상승(하강) 엣지 전, 입력으로 받아들이는데 필요한 최소 시간으로, Switching이 일어나기 전까지 입력이 정확히 인식되는데 필요한 최조 유지 시간이다. Hold time은 상승(하강) 엣지 후, 출력으로 유지하기 위해 필요한 최소 시간으로, Switching이 일어난 후 상태의 변화가 정확히 인식되도록 필요한 최소 시간을 의미한다.

도표, 라인, 스크린샷, 그래프이(가) 표시된 사진

자동 생성된 설명