

Project_Clock

Block Diagram

한지윤

목차

1. Clock

1-1. Clock 모드 Controller

1-2. Clock 모드 DataPath

2. Stop_Watch

2-1. Stop_Watch 모드 Controller

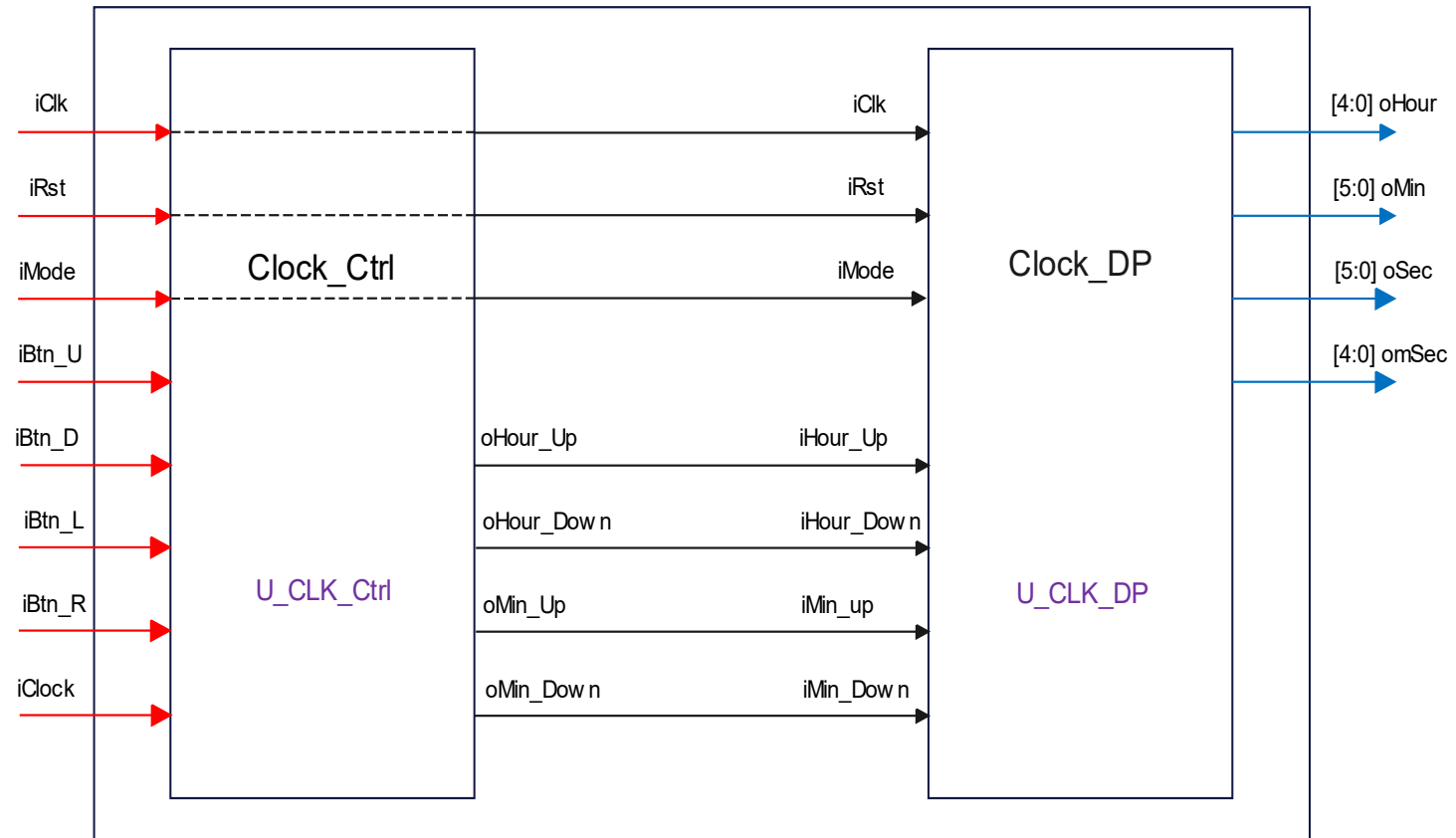
2-2. Stop_Watch 모드 DataPath

3. Timer

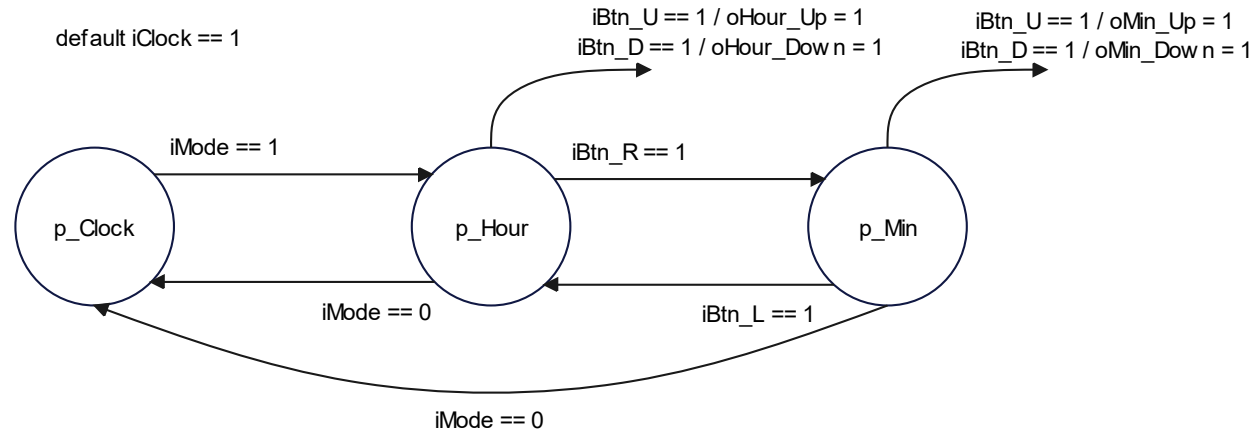
3-1. Timer 모드 Controller

4. Fnd_Controller

Clock



Clock_Ctrl



iMode[1] == 0 일 때 Clock State로 시계 모드로 작동한다.

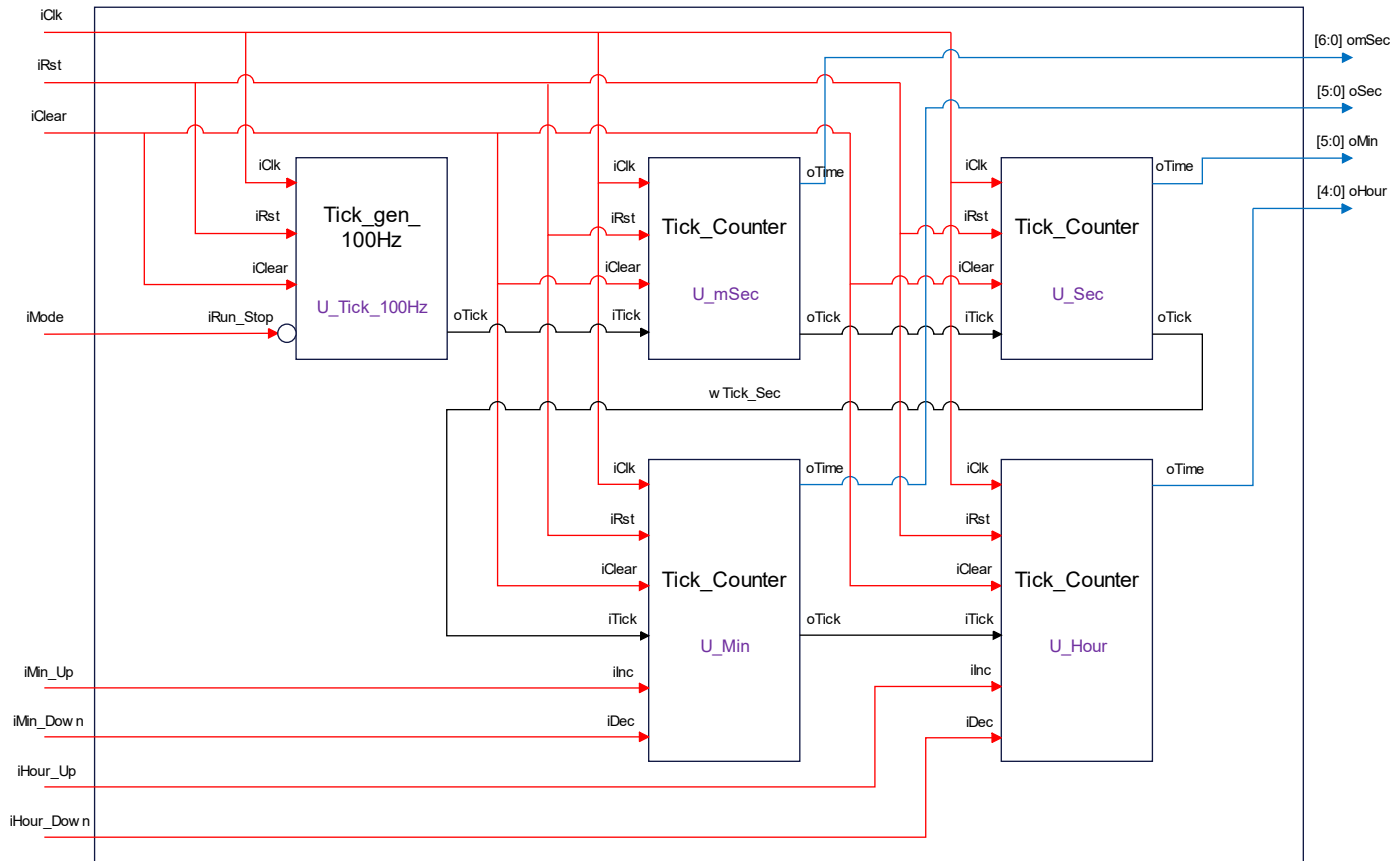
iMode[2] == 1 일 때 시간 세팅 모드로 시간을 조절할 수 있다.

오른쪽 버튼을 누르면 분 세팅 모드로 들어간다.

분 세팅 모드에서는 왼쪽 버튼을 눌러 시간 세팅 모드로 들어간다.

각각 시간 조절은 위, 아래 버튼을 사용하여 조작한다.

Clock_DP



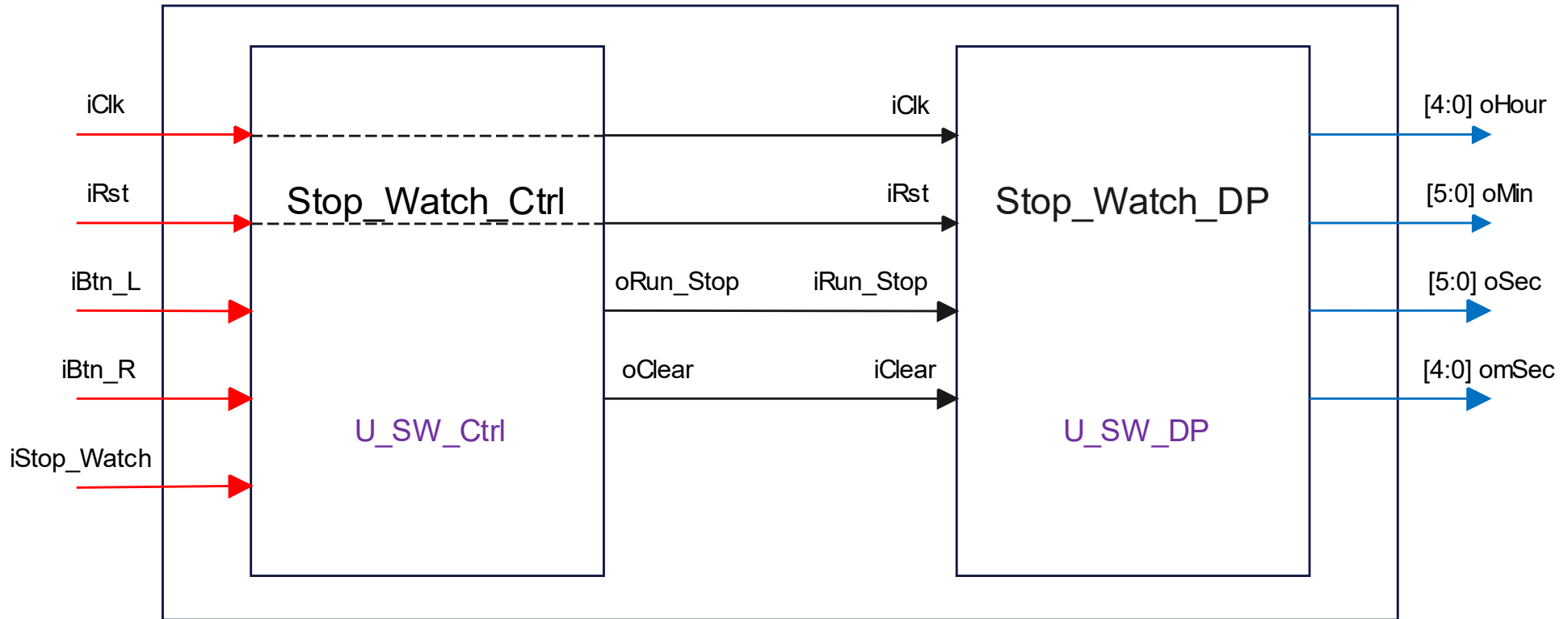
Tick_gen_100Hz 모듈을 사용하여 0.01s 시간이 들어온다.

각 Tick_Counter 모드에서 원하는 만큼 시간이 도달하면 다음 Counter의 값에 영향을 주어
시간, 분, 초, 밀리 초를 나타낼 수 있다.

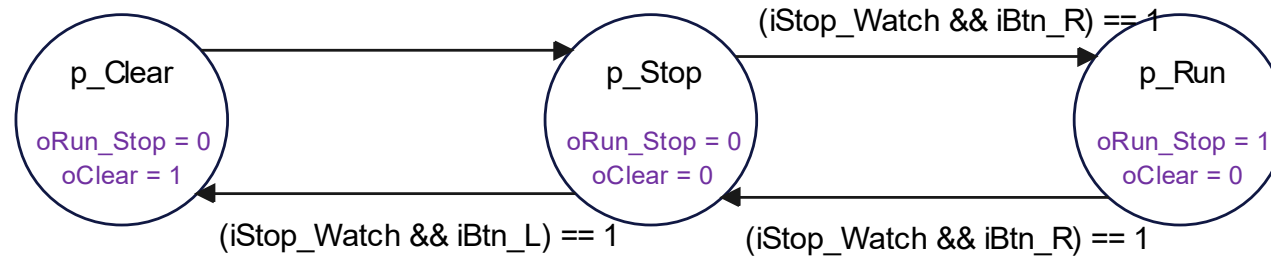
iMode [2] == 1 일 경우 시간을
조절해야 하기 때문에 Clock을 멈추기
위해 100Hz Clock과 연결시켜 주었다.

시, 분을 담당하는 Counter에는 시간을
조절해야 하기 때문에
수동 Up/Down기능을 추가해 주었다.

Stop_Watch



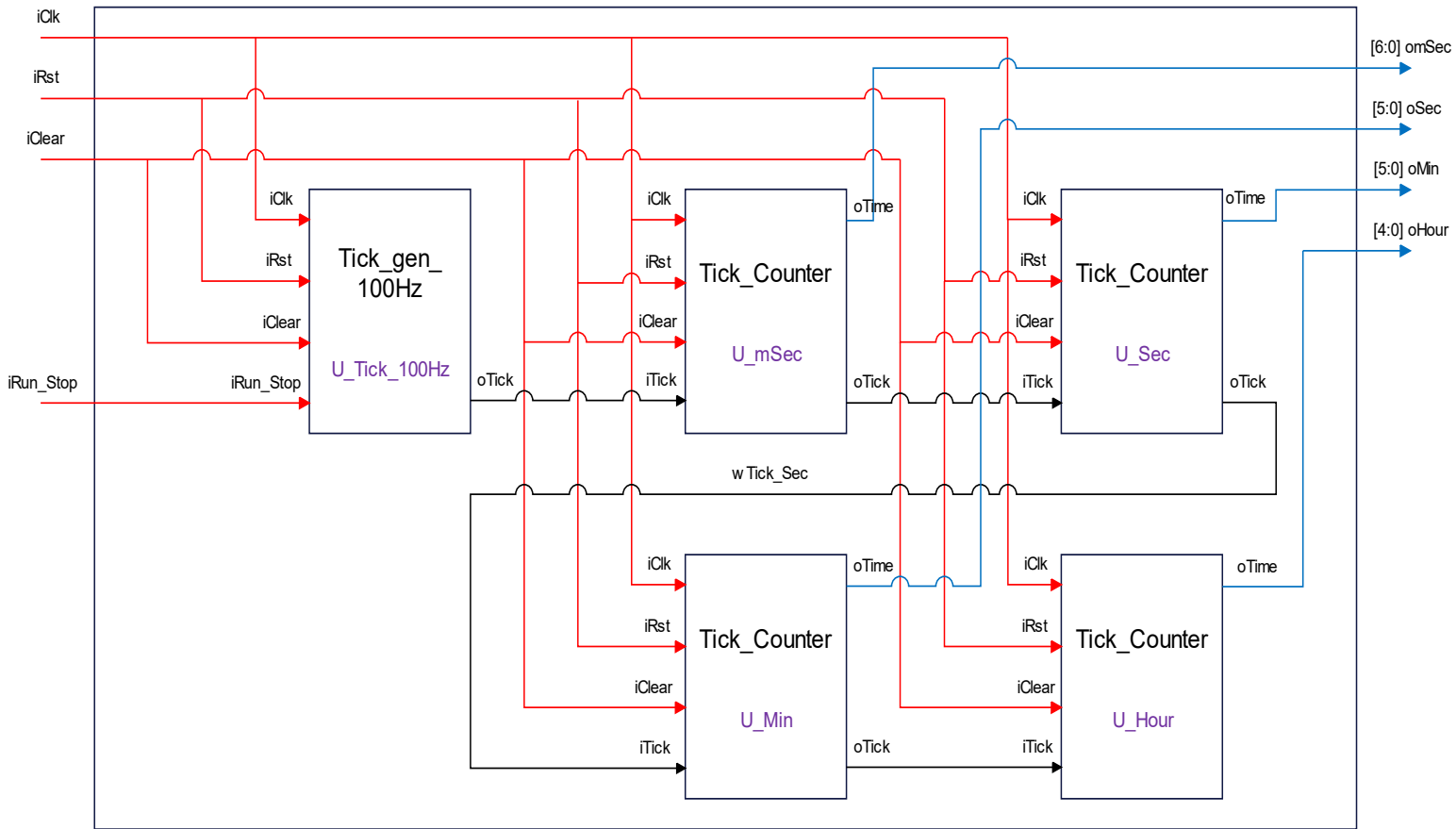
Stop_Watch_Ctrl



iMode [1] == 1 일 때 Stop_Watch State로 스톱워치 모드로 작동한다.

오른쪽 버튼으로 시작/멈춤을 조작할 수 있고,
정지 상태에서 왼쪽 버튼을 통해 초기화를 할 수 있다.
이때, 리셋이랑 다른 점은 스톱워치만 초기화 된다는 점이다.

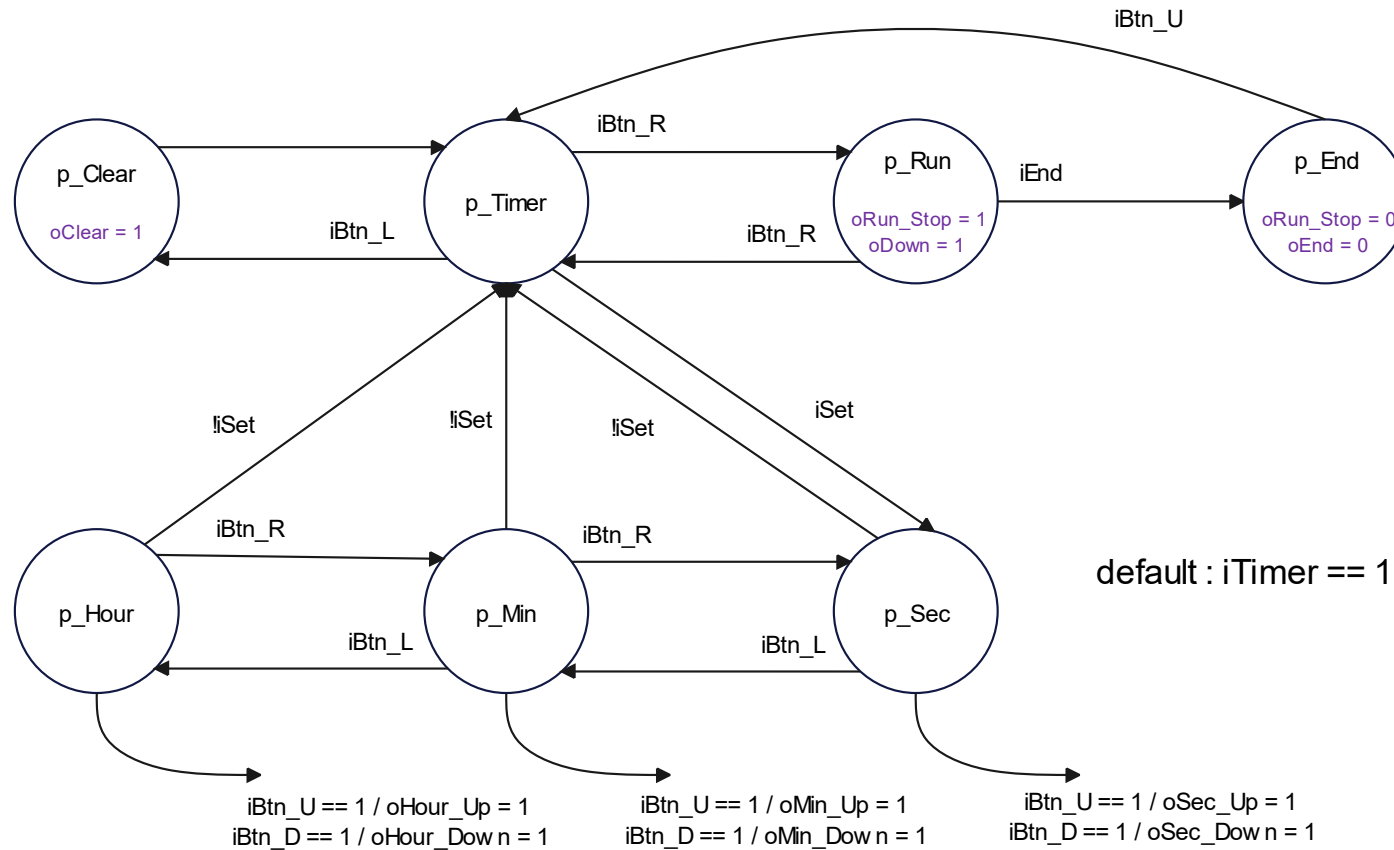
Stop_Watch_DP



Tick_gen_100Hz 모듈을 사용하여
0.01s 시간이 들어온다.

각 Tick_Counter 모드에서 원하는 만큼 시간이 도달하면 다음 Counter의 값에 영향을 주어 시간, 분, 초, 밀리 초를 나타낼 수 있다.

Timer_Ctrl

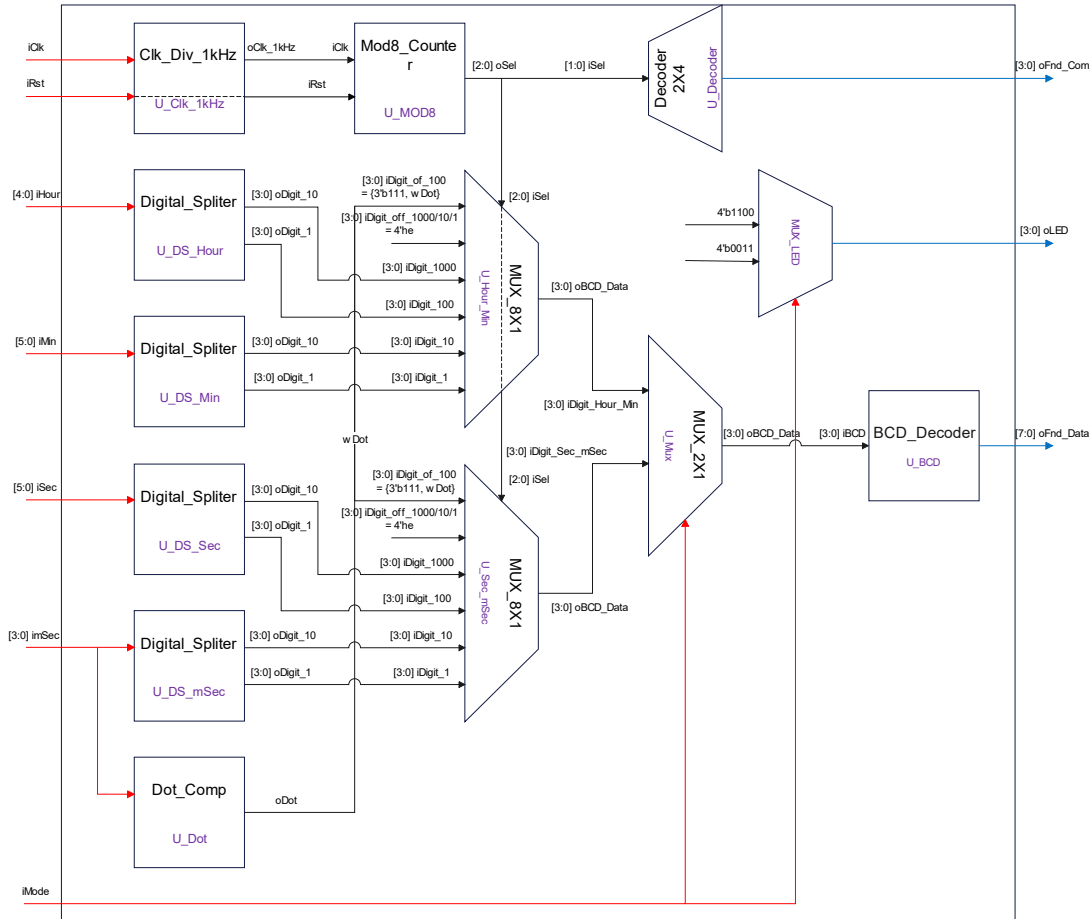


Top, DP 모듈은 위와
비슷하여 생략.

작동은 Clock과
Stop_Watch를 합쳐놓은
것과 유사함

단, Up_Counter가 아닌
Down_Counter로 작동함

Fnd_Controller



1kHz Clk과 Mod8_Counter를 통해 125H의 해상도를 가진 7Segment 디스플레이를 작동시킬 수 있다.

Digital_Splitter를 통해 시간을 각각 10진수로 변환시키고, "."을 0.5s 마다 나타내기 위해 8X1 MUX를 사용하여 동작을 추가해주었다.

iMode [0]을 통해 시,분 / 초,밀리 초 화면을 전환시킬 수 있고, LED를 추가하여 구별하기 쉽게 해주었다.

이후 뒷단에 Mux만 추가하여 Timer, Sensor 등 기능을 추가할 때 용이하도록 설계.