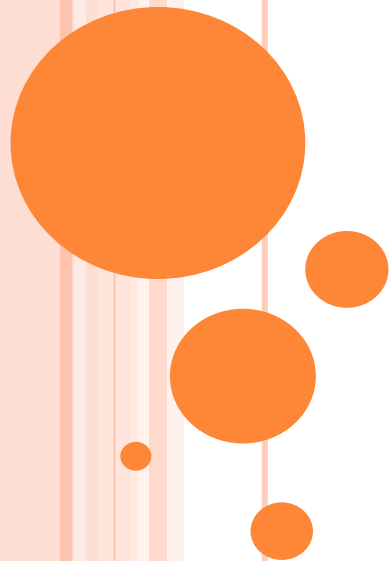


디지털 시계 설계 VHDL

1조

20123261 조이익(조장)

20123263 김한섭



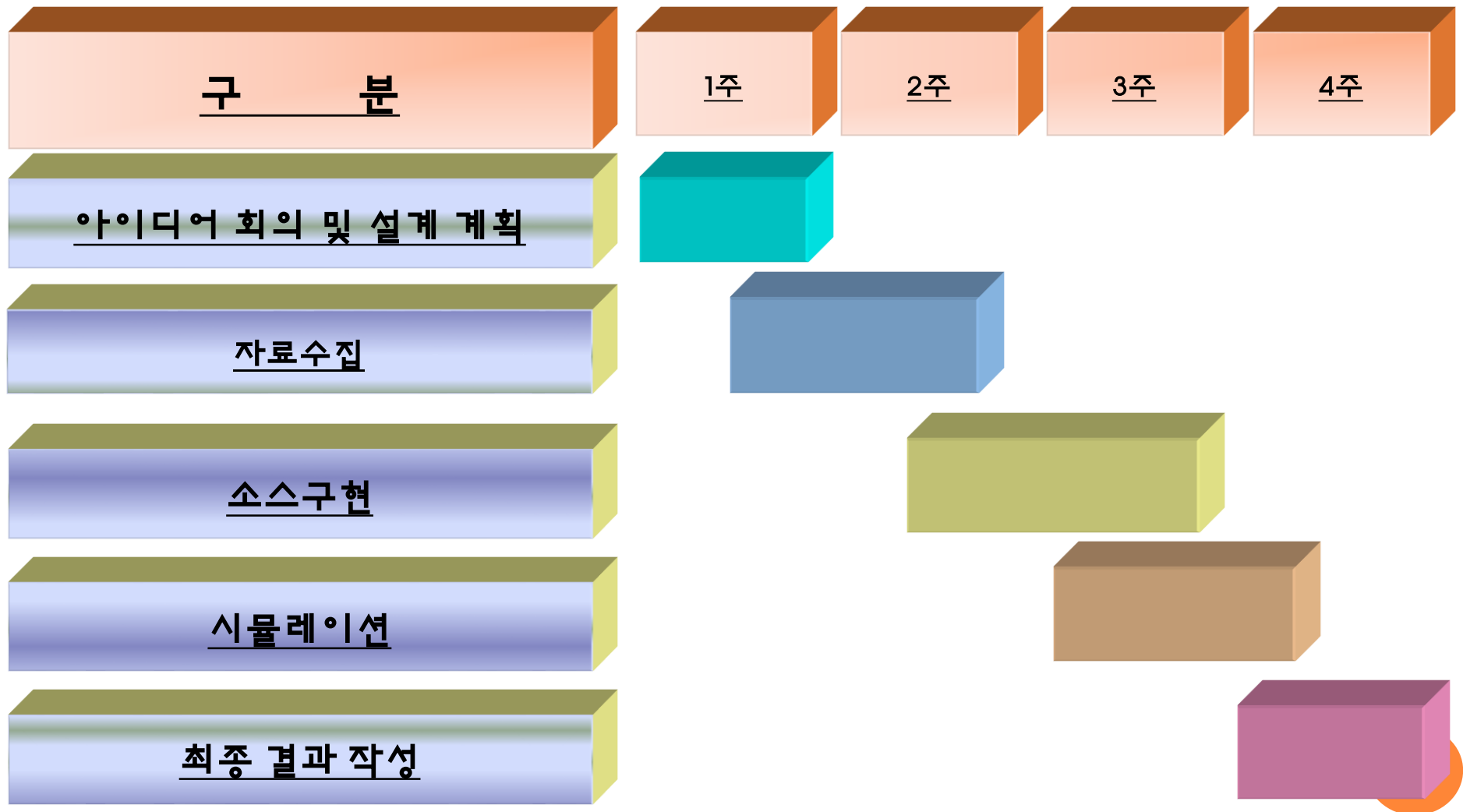
목 차

- 1) 과제 목적
- 2) 추진일정
- 3) 역할분담
- 4) 디지털 시계 구성요소
- 5) 소스 구성
- 6) 시뮬레이션
- 7) Q&A

1. 과제 목적

- 디지털 시계 설계를 위한 알고리즘 구현
- VHDL을 이용하여 디지털 시계 논리회로를 구성

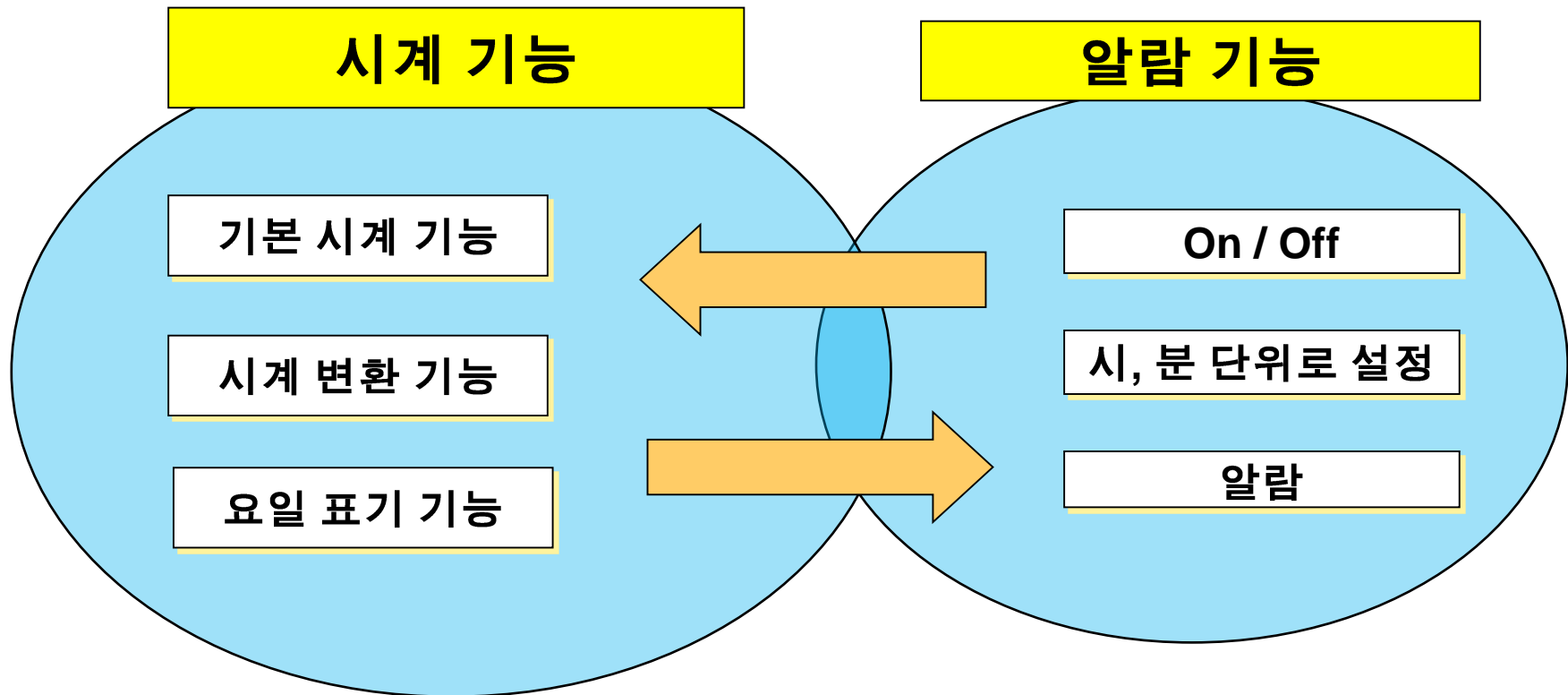
2. 추진 일정



3. 역할 분담

순 번	학과	학년	학번	이름	역할분담
1	정보통신공학과	2	20123261	조이익(조장)	아이디어 및 계획, 소스 구현, 최종 결과작성
2	정보통신공학과	2	20123263	김한섭	아이디어 및 계획, 자료수집, 시뮬레이션

4. 디지털 시계 구성요소



5. 소스 구성

- library ieee;
- use ieee.std_logic_1164.all;
- use ieee.std_logic_unsigned.all;
- entity watchdec is
- port(clk, reset, stop, hour_in, min_in, sec_in, ahour_in, amin_in, asec_in, alarm_set : in std_logic;
- -- 시계 카운터시 사용
- one_min : buffer integer range 0 to 9;
- one_sec : buffer integer range 0 to 9;
- one_hour : buffer integer range 0 to 9;
- ten_hour : buffer integer range 0 to 2;
- ten_sec : buffer integer range 0 to 5;
- ten_min : buffer integer range 0 to 5;
- -- 알람 카운터시 사용
- aone_min : buffer integer range 0 to 9;
- aone_sec : buffer integer range 0 to 9;
- aone_hour : buffer integer range 0 to 9;
- aten_hour : buffer integer range 0 to 2;
- aten_sec : buffer integer range 0 to 5;
- aten_min : buffer integer range 0 to 5;
- bell : out std_logic);
- end watchdec;

5. 소스 구성

- architecture main of watchdec is
- signal flag : std_logic;
- signal alarm : std_logic;
- signal bell : std_logic;
- signal cnt : integer range 0 to 999;
- signal cnts : integer range 0 to 1000;
- signal cc : integer range 0 to 40000000;
- signal c_LED : integer range 0 to 8;
- signal c : integer range 0 to 2;
- signal cnt2 : integer range 0 to 8;
- signal c_LED_C : integer range 0 to 7;
- begin
- process(clk, reset, sec_in, min_in, hour_in)
- begin
- if(reset='1') then
- ten_hour <= 0;
- one_hour <= 0;
- ten_min <= 0;
- one_min <= 0;
- ten_sec <= 0;
- one_sec <= 0;
- aten_hour <= 0;
- aone_hour <= 0;
- aten_min <= 0;
- aone_min <= 0;
- aten_sec <= 0;
- aone_sec <= 0;
- end if;
- end process;
- end;

5. 소스 구성

```
○ flag <= '0';

○ elsif (clk'event and clk='1') then -- 한 클럭당 1초 클럭 발생 , 상승 엣지
○ if( stop='1') then
○ ten_hour<=ten_hour; -- Stop 하면 flag 가 1이 되고 나머지
○ one_hour<=one_hour; -- 나머지 시간은 그 상태로 멈춤
○ ten_min<=ten_min;
○ one_min<=one_min;
○ ten_sec<=ten_sec;
○ one_sec<=one_sec;
○ flag<=not flag;

○ elsif( ten_hour = 2 and one_hour = 3 and ten_min =5 and one_min=9 -- 23시 59분 59초 이면 모든 시간을 0으로 설정
○ and ten_sec=5 and one_sec = 9 and flag = '0') then
○ ten_hour <= 0;
○ one_hour <= 0;
○ ten_min <= 0;
○ one_min <= 0;
○ ten_sec <= 0;
○ one_sec <= 0;
```

5. 소스 구성

```

○ elif(one_hour = 9 and ten_min = 5 and one_min = 9 and -- 9시 59분 59초이면 10시 00분 00초로 설정
○     ten_sec = 5 and one_sec = 9 and flag = '0') then
○         one_hour <= 0;
○         ten_min <= 0;
○         one_min <= 0;
○         ten_sec <= 0;
○         one_sec <= 0;
○         ten_hour <= ten_hour + 1;

○
○ elif(ten_min <= 5 and one_min = 9 and ten_sec = 5 and one_sec = 9 and flag = '0') then
○         ten_min <= 0; -- 59분 59초이면 1시 00분 00초로 설정
○         one_min <= 0;
○         ten_sec <= 0;
○         one_sec <= 0;
○         one_hour <= one_hour + 1;
○
○ elif(one_min = 9 and ten_sec = 5 and one_sec = 9 and flag = '0') then
○         one_min <= 0; -- 9분 59초이면 10분 00초로 설정
○
○         ten_sec <= 0;
○         one_sec <= 0;
○         ten_min <= ten_min + 1;
```

5. 소스 구성

```
elseif(ten_sec = 5 and one_sec = 9 and flag = '0') then
    ten_sec <= 0;          -- 59초 이면 1분으로 설정
    one_sec <= 0;
    one_min <= one_min+1;

elseif(one_sec = 9 and flag = '0') then
    one_sec <= 0;          -- 9초이면 10초로 설정
    ten_sec <= ten_sec+1;

elseif(flag = '0') then   -- 1초는 flag가 0이면 계속 카운트
    one_sec <= one_sec+1;

elseif( sec_in='1' and flag='1') then -- flag가 1이면 초 설정
    if( ten_sec=5 and one_sec=9) then
        ten_sec <= 0;
        one_sec <= 0;
    elseif( one_sec=9) then
        one_sec <= 0;
        ten_sec <= ten_sec+1;
    else
        one_sec <= one_sec+1;
    end if;
```

5. 소스 구성

```

elsif( min_in='1' and flag='1') then -- flag가 1이면 분 설정
    if( ten_min=5 and one_min=9) then
        ten_min <= 0;
        one_min <= 0;

    elsif(one_min=9) then
        one_min <= 0;
        ten_min <= ten_min+1;
    else
        one_min <= one_min+1;
    end if;

elsif( hour_in='1' and flag='1') then -- flag가 1이면 시 설정
    if( ten_hour=2 and one_hour=3) then
        ten_hour <= 0;
        one_hour <= 0;
    elsif( one_hour=9) then
        one_hour <= 0;
        ten_hour <= ten_hour+1;
    else
        one_hour <= one_hour+1;
    end if;

end if;

end process;
```

5. 소스 구성

```
○ process(clk,reset,sec_in,min_in,hour_in,asec_in,amin_in,ahour_in) -- 프로세스 시작
○ begin
○
○         if(reset='1') then                -- reset이 1이면 alarm, bell 초기화
○
○             alarm<='0';
○             bell<='0';
○
○         elsif (clk'event and clk='1') then -- 클럭 발생
○         if( alarm_set ='1') then          -- alarm_set에 1이 들어오면 알람 시간 설정 가능
○             aten_hour<=aten_hour;         1번 누르면 alarm = 1
○             aone_hour<=aone_hour;          2번 누르면 alarm = 0
○             aten_min<=aten_min;
○             aone_min<=aone_min;
○             aten_sec<=aten_sec;
○             aone_sec<=aone_sec;
○             alarm<=not alarm;
○             if(alarm='0') then
○
○                 bell<='0';
○
○             end if;
○         elsif( asec_in='1' and alarm='1') then -- 초 설정
○             if( aten_sec=5 and aone_sec=9) then
○
○                 aten_sec <= 0;
○                 aone_sec <= 0;
```

5. 소스 구성

```
elseif( aone_sec=9) then
    aone_sec <= 0;
    aten_sec <= aten_sec+1;

else
    aone_sec <= aone_sec+1;
end if;

elseif( amin_in='1' and alarm='1') then    -- 분 설정
    if( aten_min=5 and aone_min=9) then
        aten_min <= 0;
        aone_min <= 0;

elseif( aone_min=9) then

        aone_min <= 0;
        aten_min <= aten_min+1;

    else
        aone_min <= aone_min+1;
    end if;

elseif( ahour_in='1' and alarm='1') then    -- 시 설정
    if( aten_hour=2 and aone_hour=3) then
        aten_hour <= 0;
        aone_hour <= 0;

    elseif( aone_hour = 9) then

        aone_hour <= 0;
        aten_hour <= aten_hour+1;

    else

        aone_hour <= aone_hour+1;

    end if;
```

5. 소스 구성

- elsif(alarm ='1' and aten_hour=ten_hour and

-

-

aone_sec=one_sec)then

-

-

-

-

-

end if;

end if;

-

end process;

-

end main;

aone_hour=one_hour and aten_min=ten_min and

aone_min=one_min and aten_sec=ten_sec and

-- 알람 시간과 현재시간이 같으면 벨 울림

bell <='1';

5. 소스 구성

```
○ process(clk)
○     variable ccc : integer range 0 to 43200;
○     variable lcdcount : integer range 0 to 8;
○ begin
○     if (clk'event and clk='1') then
○     case cnt is
○         when 0 =>
○             seg_com <= "01111111";
○             seg_data <= seg1;
○         when 1 =>
○             seg_com <= "10111111";
○             seg_data <= seg2;
○         when 2 =>
○             seg_com <= "11011111";
○             seg_data <= seg3;
○         when 3 =>
○             seg_com <= "11101111";
○             seg_data <= seg4;
○         when 4 =>
○             seg_com <= "11110111";
○             seg_data <= seg5;
○         when 5 =>
○             seg_com <= "11111011";
○             seg_data <= seg6;
○     end case;
○ end if;
○ end process;
```


5. 소스 구성

```

when 6 =>
    seg_com <= "11111101";
    seg_data <= "00000000";
when 7 =>
    seg_com <= "11111110";
    seg_data <= ap(c);

case c_LED is
    when 0 => LED_D <= "0000001";
    when 1 => LED_D <= "0000010";
    when 2 => LED_D <= "0000100";
    when 3 => LED_D <= "0001000";
    when 4 => LED_D <= "0010000";
    when 5 => LED_D <= "0100000";
    when 6 => LED_D <= "1000000";
    when others => null;
end case;

if c_LED_C=8 then
    c_LED_C<=0;
end if;

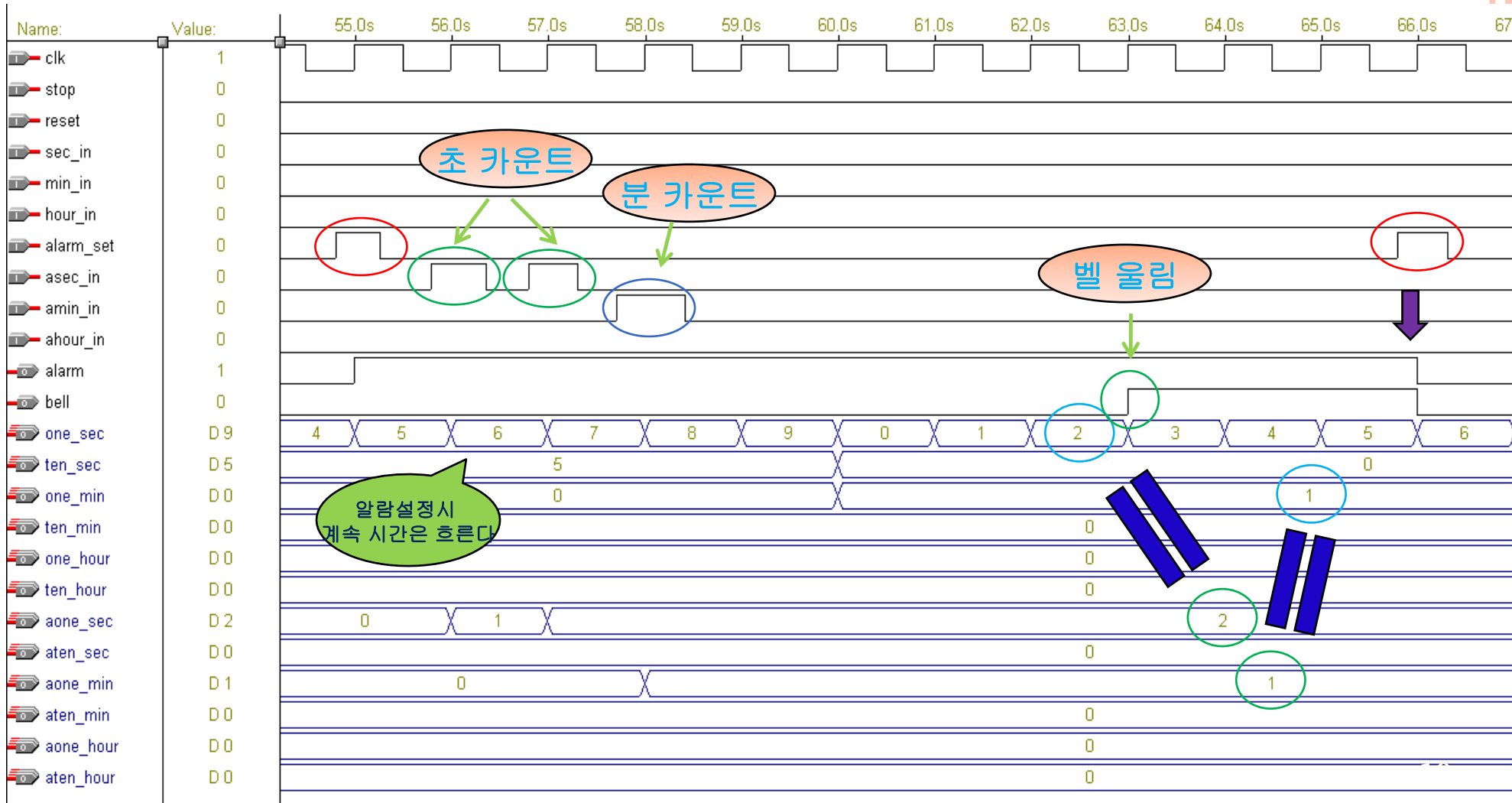
cnt <= cnt+1;
end if;

```

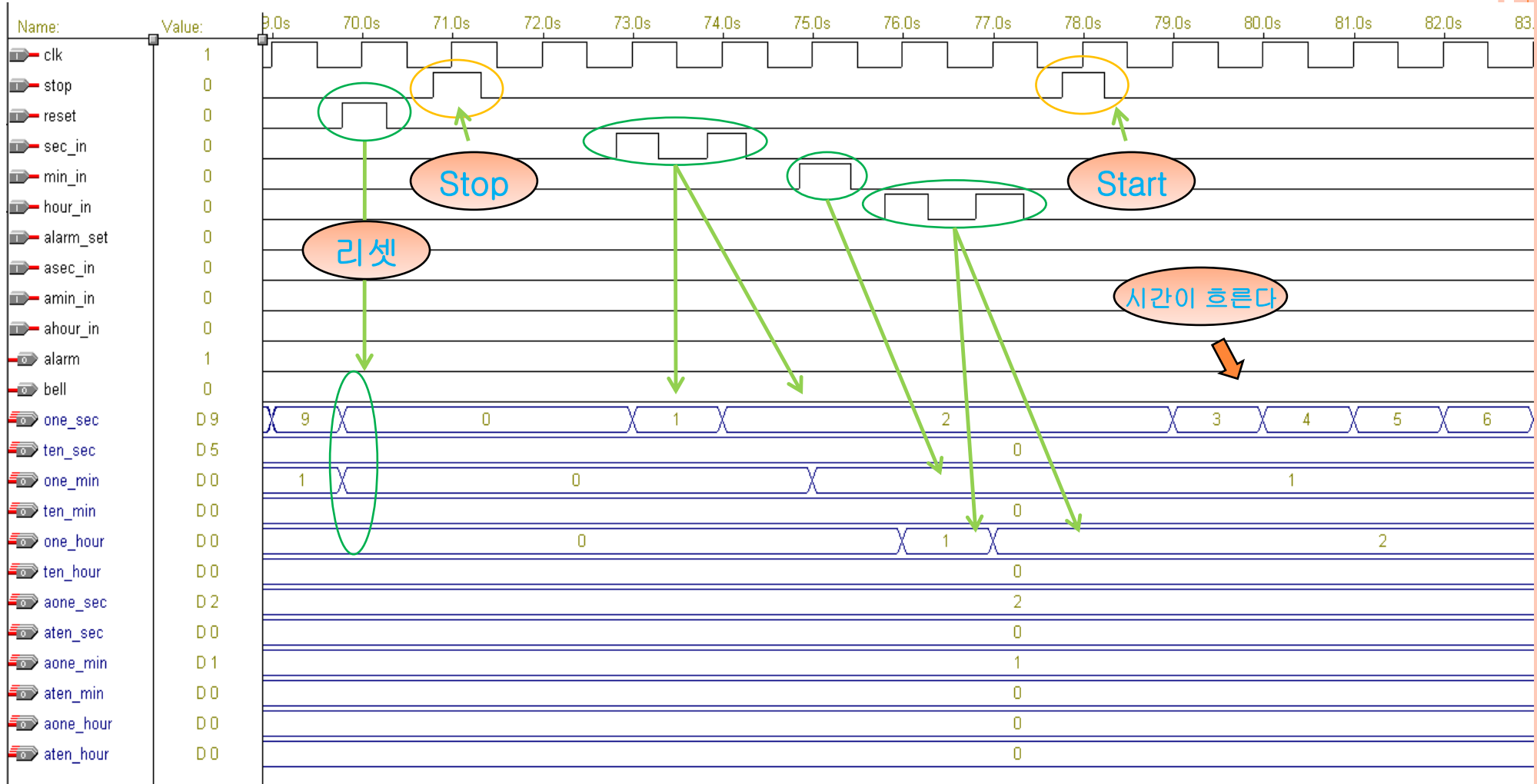
5. 소스 구성

- if (cnt=8) then
- cnt <= 0;
- end if;
- end process;

6. 시뮬레이션



6. 시뮬레이션



7. Q&A

