

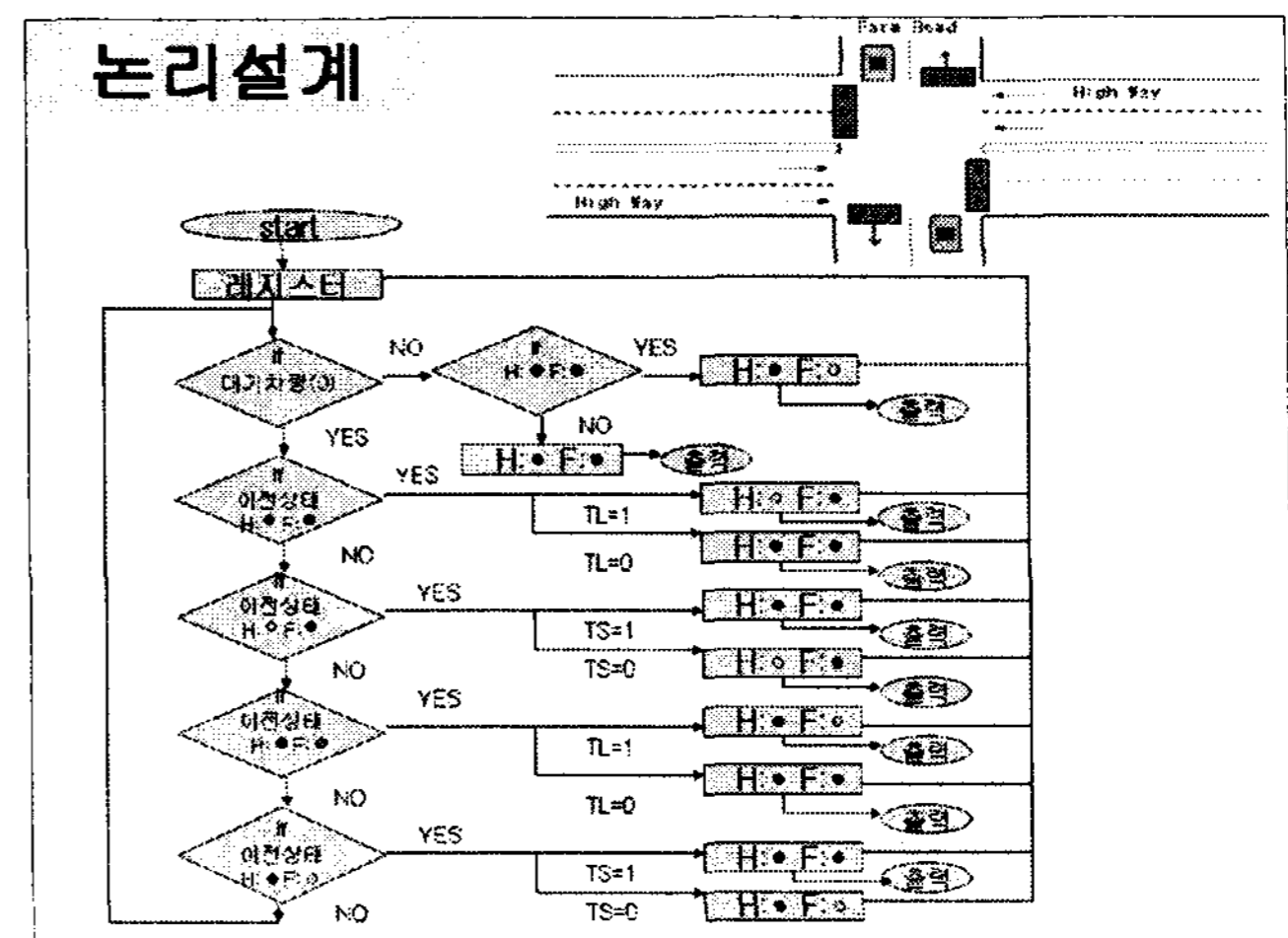
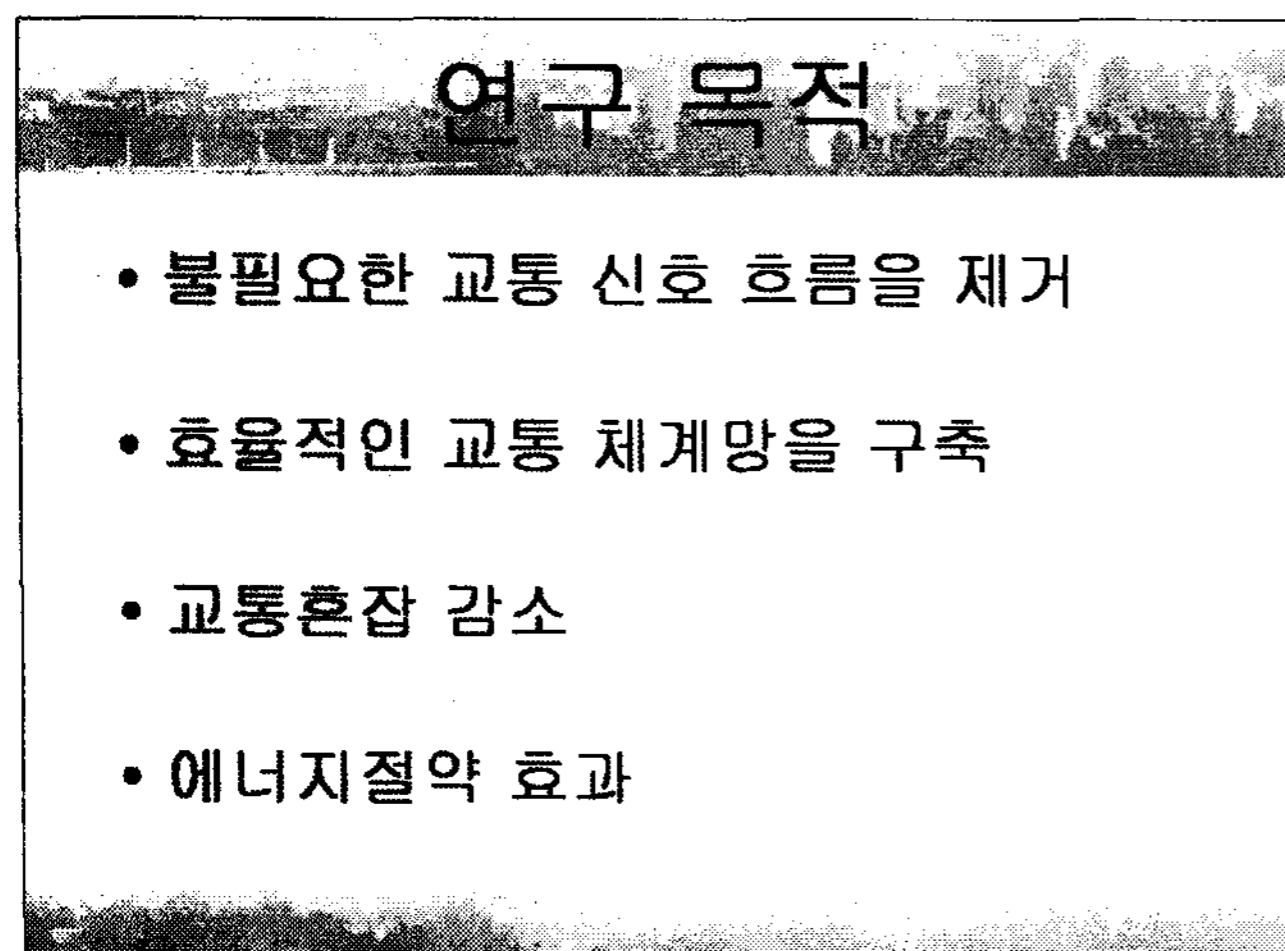
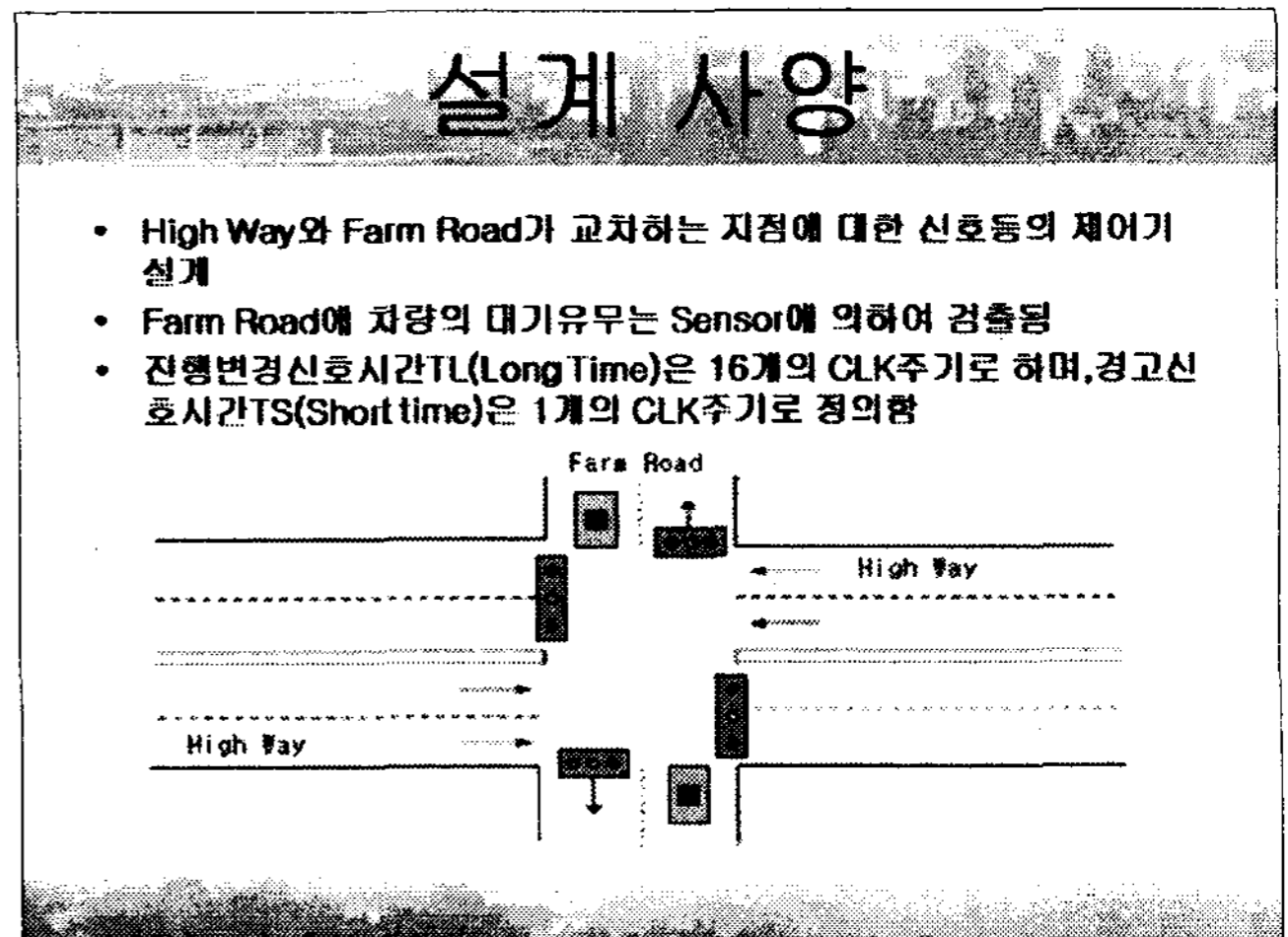
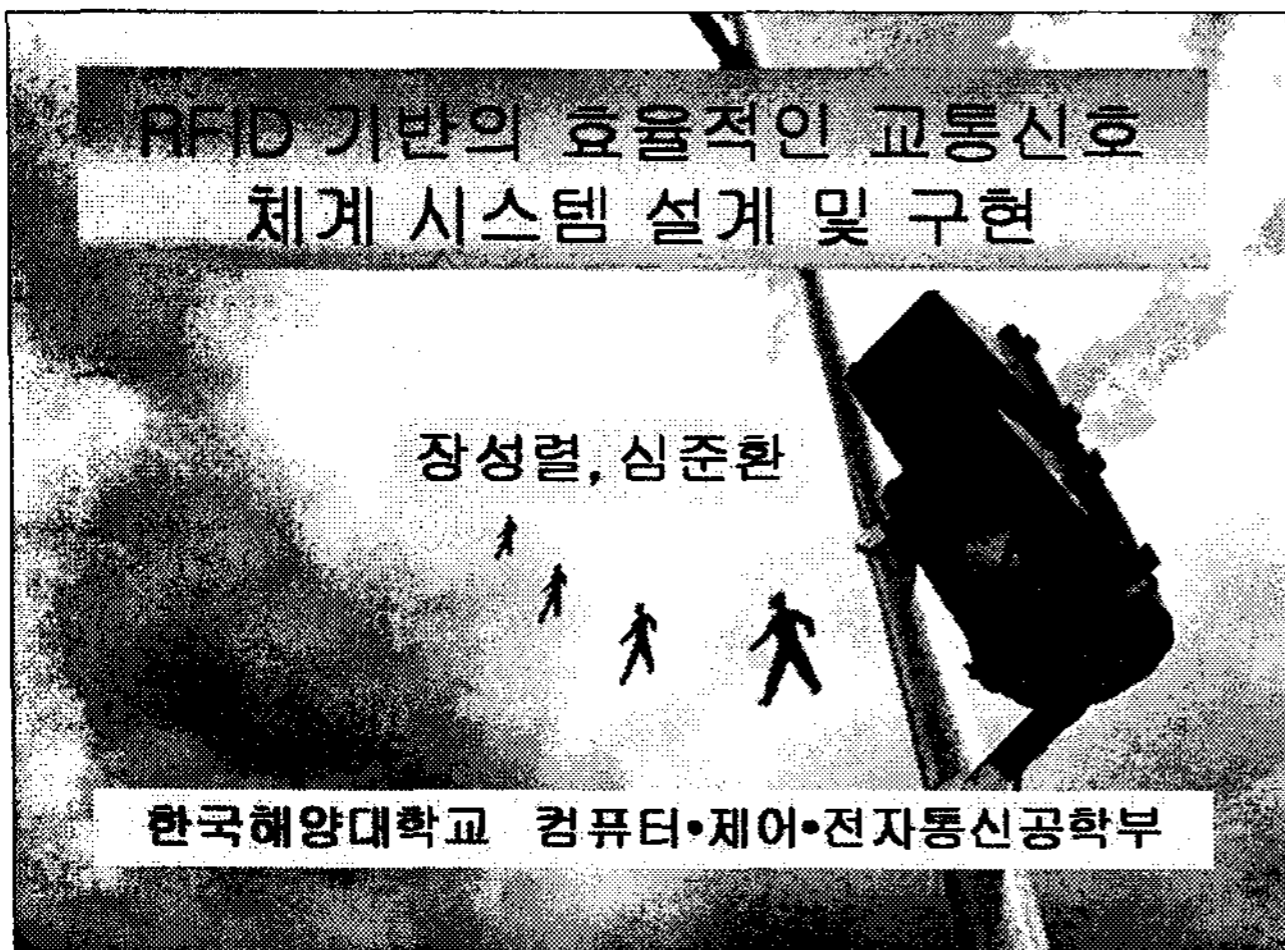
# RFID 기반의 효율적인 교통신호체계 시스템 설계 및 구현

장성렬\*, † 심준환\*

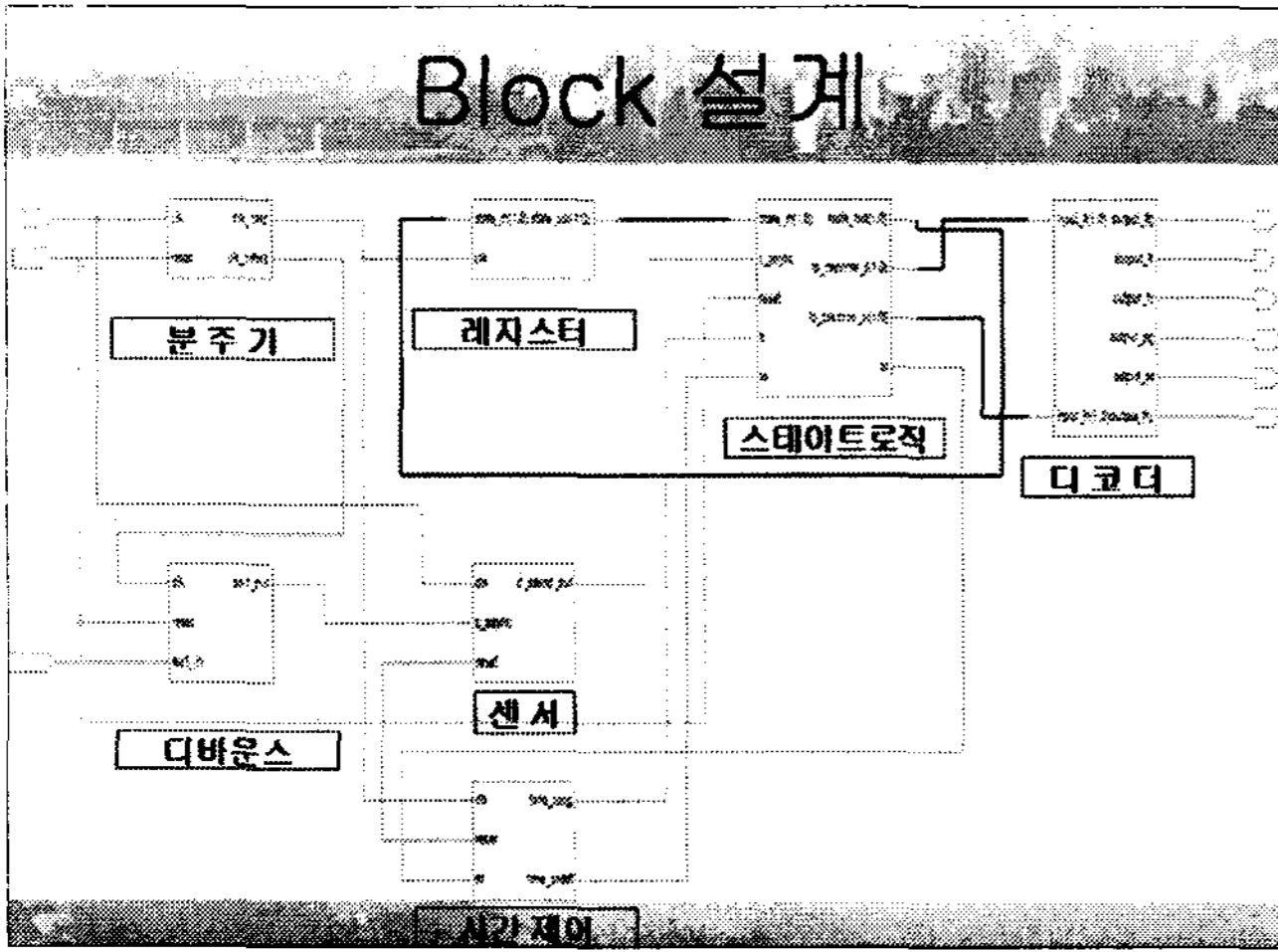
\*한국해양대학교 컴퓨터·제어·전자통신공학부

**요약** : 본 논문은 RFID를 이용한 효율적인 교통신호 체계를 구축하기 위해서 연구했다. 이 시스템은 4차선의 주차로와 2차선의 소방로로 이루어진 교차로에서 주차로의 교통량이 많을 경우, 차로에서는 녹색불이 지속되어 교통의 흐름을 원활하게 해주며, 소방로에서 차량이 일정량 증가하게 되면, 주차로의 신호 등이 적색등으로 바뀌어 소방로의 차량이 소통되도록 하는 지능형 교통체계 시스템을 설계하였다. 본 시스템의 구성을 위해서 RFID 태그를 차량에 설치하고, 리더기를 도로에 설치하여 차량의 유무를 판별할 수 있도록 하였다.

**핵심용어** : 항만보안, 영상보안시스템, IP 카메라, 프로그래시브 카메라, PDA, MPEG4



† 교신저자 : 심준환(정회원) jhsim@hhu.ac.kr

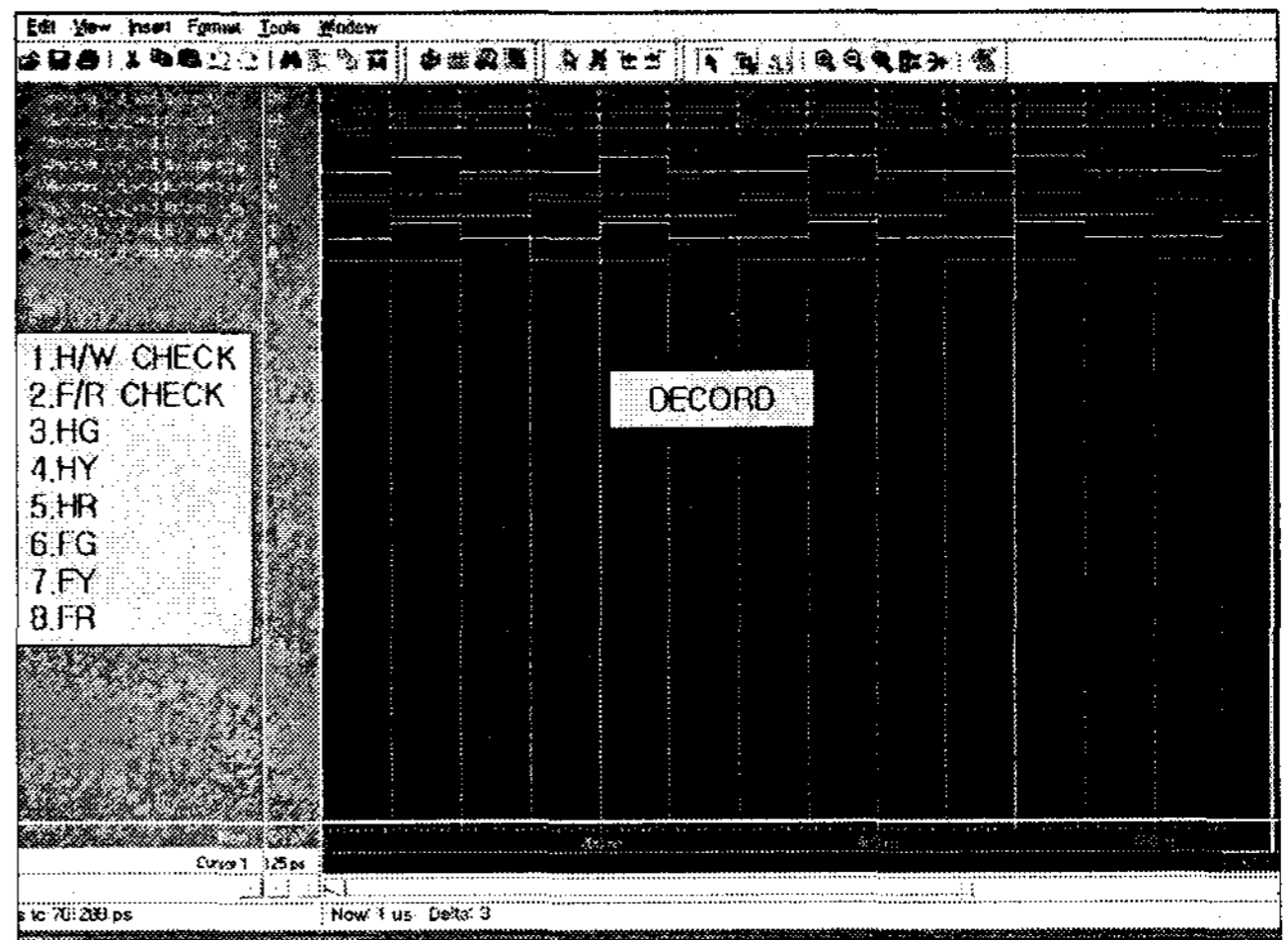
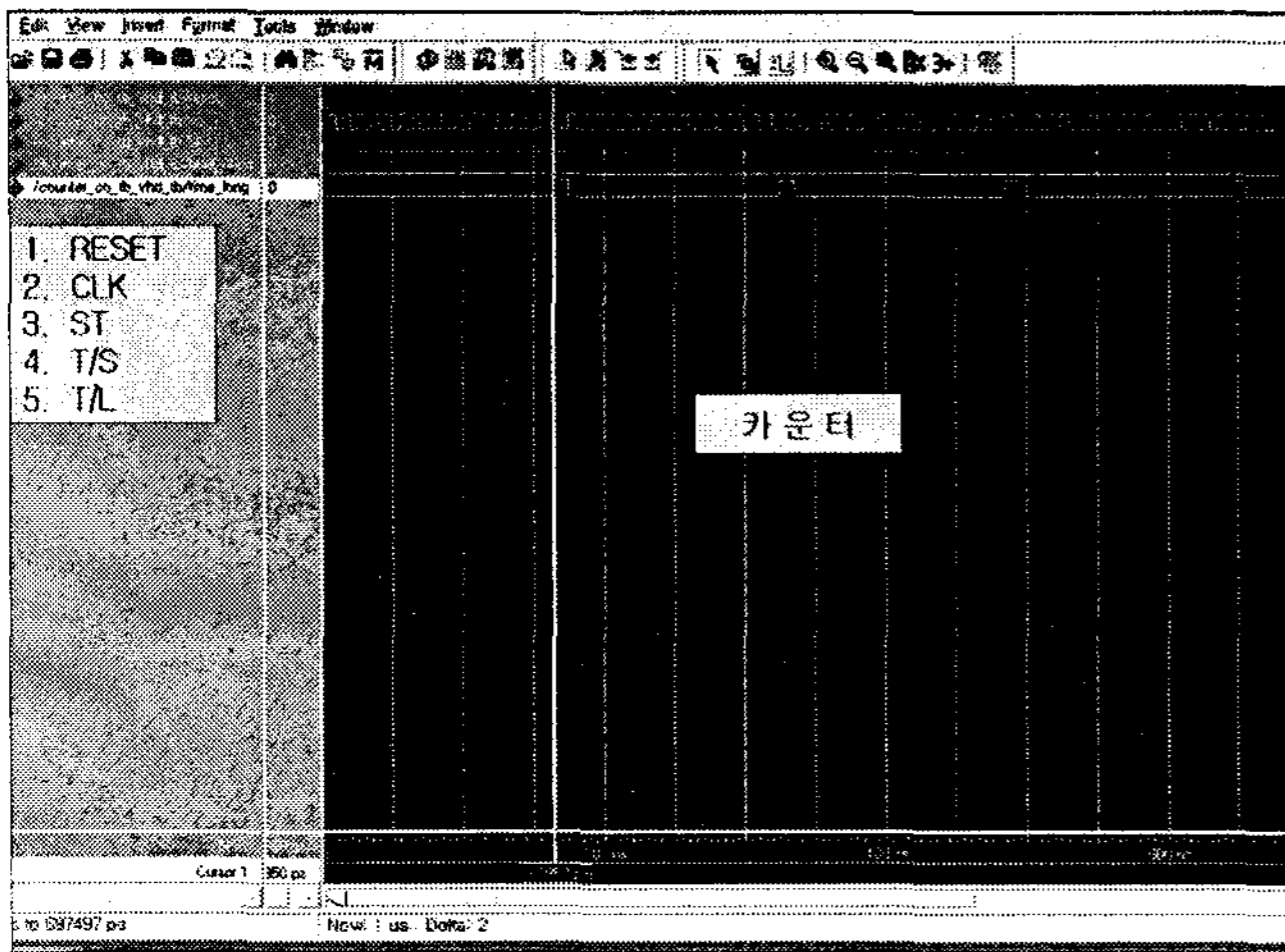


```

-- Green: 혹은 Red 상태인 경우
else
  -- TL 시간이 경과
  if count = "1111" then
    time_long <= '0';
    time_short <= '0';
    count <= "0000";
  else
    time_long <= '0';
    time_short <= '0';

    case count is
      when "0000" => count <= "0001";
      when "0001" => count <= "0010";
      when "0010" => count <= "0011";
      when "0011" => count <= "0100";
      when "0100" => count <= "0101";
      when "0101" => count <= "0110";
      when "0110" => count <= "0111";
      when "0111" => count <= "1000";
      when "1000" => count <= "1001";
      when "1001" => count <= "1010";
      when "1010" => count <= "1011";
      when "1011" => count <= "1100";
      when "1100" => count <= "1101";
      when "1101" => count <= "1110";
      when "1110" => count <= "1111";
      when "1111" => count <= "0000";
      when others => count <= "0000";
    end case;
  end if;
end if;
end if;
end process;
end main;

```



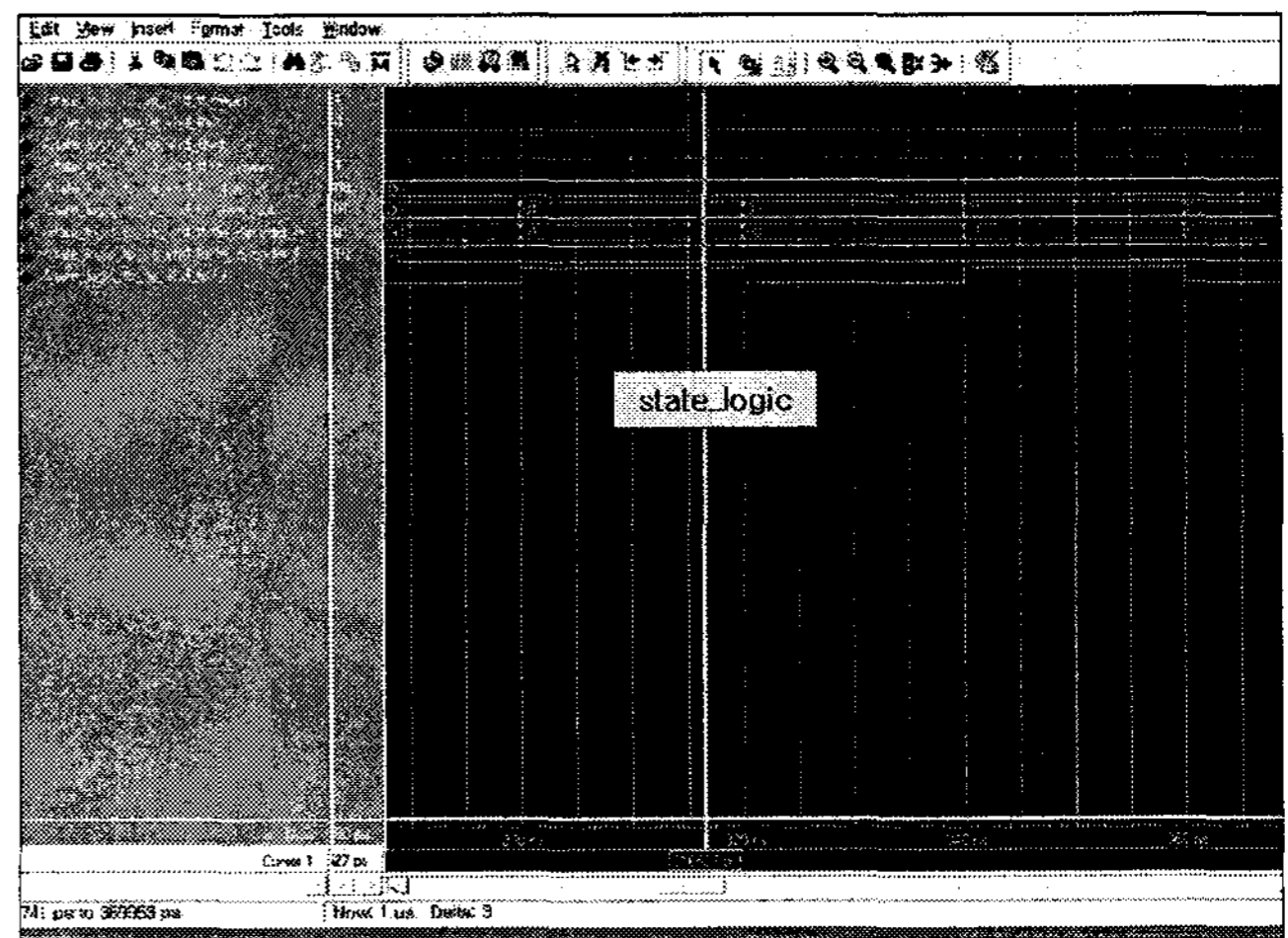
```

-- 파일명 : counter.vhd
-- 설명 : TL 시간과 TS 시간을 출력하는 카운터
library ieee;
use ieee.std_logic_1164.all;

entity counter is
  port(
    reset : in std_logic;
    clk : in std_logic;
    st : in std_logic;
    time_short : out std_logic;
    time_long : out std_logic;
  );
end counter;

architecture main of counter is
  signal count : std_logic_vector(3 downto 0);
begin
  process (clk, reset, st)
  begin
    -- reset(초기화) 상태
    if reset = '0' then
      time_short <= '0';
      time_long <= '0';
      count <= "0000";
    -- 일반적인 상태
    else
      if (clk'event) and (clk='1') then
        -- Yellow 상태인 경우
        if st = '1' then
          count <= "0000";
          time_long <= '0';
          time_short <= '1';
        end if;
      end if;
    end if;
  end process;
end main;

```





```

--00: High way Green, farmway Red      01: Green
--01: High way Yellow, farmway Red     01: Yellow
--10: High way Red, farmway Green     10: Red
--11: High way Red, farmway Yellow

#reset = '1' then

-- HG, FR
if state_in = "00" then
  -- T1 시간이 경과하고 차량이 대기중
  if # = '1' and classno = '1' then
    to_decoder_h <= "01";
    to_decoder_l <= "10";
    state_out <= "01";
    st <= '1';
  -- T1 시간이 경과하지 않거나, T1 시간이 경과하고 차량을 없애는 경우
  else
    to_decoder_h <= "00";
    to_decoder_l <= "10";
    state_out <= "00";
    st <= '0';
  end if;
-- HY, FR
elsif state_in = "01" then
  -- T2 시간이 경과하면
  if ts = '1' then
    to_decoder_h <= "10";
    to_decoder_l <= "00";
    state_out <= "10";
    st <= '0';
  -- T2 시간이 경과하지 않으면
  else
    to_decoder_h <= "01";
    to_decoder_l <= "10";
    state_out <= "01";
    st <= '0';
  end if;
end if;

```

