

---

# FPGA를 이용한 전자 키 구현

유정근\* · 허창우\*

목원대학교

## Design of Electronic Key Using FPGA

Jeong-gun Yoo\* · Chang-wu Hur\*

Mokwon University

E-mail : [voice666@mokwon.ac.kr](mailto:voice666@mokwon.ac.kr)

### 요 약

최근 키를 가지고 다니는 불편함과 보안성을 고려한 전자 키들이 많이 생산되고 있다. 키의 불편함과 보안성을 보완하는 방법에는 비밀번호 입력, 지문인식, 홍채인식 등의 방법이 이용되고 있는데, 본 논문에서는 비밀번호를 입력하는 방법으로 설계하였다. Altera사의 Software인 MAXPLUS II를 이용하여 설계하였고, Hardware Language인 VHDL을 이용하였다.

### 키워드

CPLD, FPGA, VHDL

## I. 서 론

전자공학의 급격한 발달로 인해 집적회로에서 computer 분야에 이르기까지 눈부신 성장을 이룩하게 되었다. 특히 반도체의 발달로 수만 개의 transistor를 집적할 수 있는 IC(Integrated Circuit), 주문형반도체(ASIC-Application Specific Integrated Circuit)등이 개발되어 그 수요는 날로 증가되고 있는 실정이다.

현재 digital system의 설계와 제작은 자동화 설계 tool을 이용하고 있으며 특히 CPLD(Complex Programmable Logic Device)나 FPGA(Field Programmable Gate Array)에 의한 설계가 많이 이용되고 있다. CPLD와 FPGA에 의한 설계와 제작은 그 자체로써 많은 응용분야가 있으며 ASIC제작 이전에 설계자가 직접 회로를 설계, 시스템을 구현하여 실제의 검증에도 이용한다. 이러한 설계방식은 빠른 설계와 구현이 가능할 뿐만 아니라 설계상의 문제도 쉽게 수정할 수 있다는 장점이 있다.[1]

Digital 설계 자동화를 위해 ALTERA사에서 제공하는 MAX+plus의 software는 다양한 design 방식에 의한 작업을 할 수 있는데 schematic design과 hardware language에 의한 작업을 수행할 수 있으며, 또한 simulation을 실행함으로써 program에 대한 에러를 검증하는데 편리하다.

이러한 이유로 전자 키에 대한 설계 및 제작을 함에 있어서 최근 digital 설계에서 각광 받고 있는 VHDL에 의한 설계방식을 사용하였다.

VHDL(Very high speed integrated circuit Hardware Description Language)은 회로설계에서 상위의 동작 level에서부터 하위의 gate level까지 hardware를 기술하고 설계하도록 하는 language로서 CAD업계 및 IEEE의 표준 language이며 미국 정부가 지원 및 공인한 hardware 설계 language이다. 또한 digital 회로의 성질을 기초로 만들어진 language이다.[2]

또한 MAX+plus의 software를 이용하여 VHDL로 programming한 source를 simulation하여 시계가 정확히 구동을 하는지 확인하고 이를 제작하였다.[3][4][5]

## II. 전자 키 흐름 설명

전자 키의 모드는 크게 문을 열기 위해 비밀번호를 누르는 모드와 비밀번호를 변경하는 모드 두 가지로 나누어진다.

먼저, 사용자가 문을 열기 위해서 비밀번호를 누르는 모드는 저장되어 있는 비밀번호와 비교를 해서 모두 4자리인 비밀번호가 저장되어 있는 비밀번호와 일치하면, 비밀번호 입력 후 ENTER 버튼을 누르면 문

은 열리게 되어 있다. 이 때, 4자리의 비밀번호중 하나라도 맞지 않을 시에는 문은 열리지 않게 된다.

비밀번호를 변경하는 모드에서는 비밀번호 입력 후, SECR 버튼과 RESET 버튼을 누르고 새로운 비밀번호를 입력한 뒤, CLR버튼을 누르면 된다.

### III. 시스템 입·출력 구성도



그림 1. 시스템 입·출력 구성도

[그림1]은 시스템의 전체 입·출력을 나타낸 구성도이다.

데이터를 입력할 수 있는 버튼은 0에서 9까지의 10개의 숫자 버튼과 RESET, ENTER, SECR, CLR 버튼 등, 모두 14개의 버튼으로 이루어져 있다.

RESET 버튼은 비밀번호를 입력할 때, 혹은 비밀번호를 수정할 때, 잘못된 입력이 있을 시에 지금까지 입력했던 데이터를 초기화시킬 때 사용하는 버튼이다.

ENTER 버튼은 비밀번호의 입력이 끝났다는 것을 알리는 신호이다. 이 때, 비밀번호가 맞으면 문은 열리게 되고 그렇지 않을 시에는 문은 열리지 않고 지금까지 입력했던 숫자들은 초기화된다.

SECR 버튼은 비밀번호를 바꿀 시에 사용하는 버튼으로써, 현재 비밀번호를 누르고 SECR 버튼을 누르면 비밀번호 수정 모드로 바뀌게 된다.

CLR 버튼은 SECR 버튼을 누르고 비밀번호 수정 후, 비밀번호 수정 완료알리는 신호를 보낼 때 사용하는 버튼이다.

이렇게 해서 14개의 버튼과 Clock 신호로 모두 15개의 신호로 입력은 이루어져 있다.

출력은 문을 열게 하는 DOOR 신호와 눌러진 버튼을 7segment에 표시하는 SEG\_DISPLAY 버튼으로 이루어져 있다. 아래에 7segment의 상태 표현을 나타낸 FUNCTION 구문을 나타내었다.

```
function dis_seg(x: in integer range 0 to 17)
    return std_logic_vector is
    variable seg_decode: std_logic_vector(6 downto 0);
begin
    case x is
        when 0 => seg_decode := "0111111";
        when 1 => seg_decode := "0000110";
```

```
when 2 => seg_decode := "1011011";
when 3 => seg_decode := "1001111";
when 4 => seg_decode := "1100110";
when 5 => seg_decode := "1101101";
when 6 => seg_decode := "1111101";
when 7 => seg_decode := "0100111";
when 8 => seg_decode := "1111111";
when 9 => seg_decode := "1100111";
when 10 => seg_decode := "1110111";
when 11 => seg_decode := "1011110";
when 12 => seg_decode := "1111011";
when 13 => seg_decode := "0110111";
when 14 => seg_decode := "0110001";
when 15 => seg_decode := "1111110";
when 16 => seg_decode := "1110011";
when others => seg_decode := "0000000";
end case;
return(seg_decode);
end dis_seg;
```

### IV. 시스템 내부 구성도

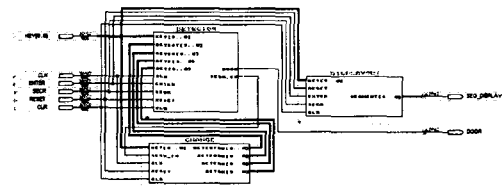


그림 2. 시스템 내부 BLOCK DIAGRAM

[그림2]는 시스템 내부를 나타낸 BLOCK DAIGRAM이다. 그림과 같이 본 논문은 3개의 내부 COMPONENT로 구성되어 있다.

DETECTOR 부분은 사용자가 누르는 비밀번호와 저장되어 있는 비밀번호를 비교하여 DOOR 신호를 보내야 할지 그렇지 않을지를 결정하는 부분이다.

CHANGE 부분은 비밀번호 수정 시에 새로운 비밀번호를 저장하는 곳으로 DETECTOR 부분에 저장되어 있는 비밀번호의 정보를 공유한다.

DISPLAYOUT 부분은 7segment에 사용자가 누르는 버튼을 표시해 주는 기능을 한다.

3개의 COMPONENT를 조합한 MAIN 구문을 아래에 나타내었다.

```
library ieee;
use ieee.std_logic_1164.all;
```

```

use ieee.std_logic_unsigned.all;
use ieee.std_logic_arith;

entity main is
  port(
    clk          : in std_logic;
    reset        : in std_logic;
    enter        : in std_logic;
    secr         : in std_logic;
    clr          : in std_logic;
    key          : in std_logic_vector(9 downto 0);
    door         : out std_logic;
    seg_display  : out std_logic_vector(6 downto 0)
  );
end main;

architecture sample of main is
  signal temp1 : std_logic;
  signal temp2, temp3, temp4, temp5 : std_logic_vector(9 downto 0);

  component detector
    port(
      reset : in std_logic;
      clk   : in std_logic;
      enter : in std_logic;
      secr  : in std_logic;
      clr   : in std_logic;
      detect : in std_logic_vector(9 downto 0);
      detec  : in std_logic_vector(9 downto 0);
      deteok : in std_logic_vector(9 downto 0);
      detok  : in std_logic_vector(9 downto 0);
      detok  : out std_logic_vector(9 downto 0);
      detok  : out std_logic_vector(9 downto 0);
      detok  : out std_logic_vector(9 downto 0)
    );
  end component;

  component displayout
    port(
      key : in std_logic_vector(9 downto 0);
      reset, enter, secr, clr : in std_logic;
      segment : out std_logic_vector(6 downto 0)
    );
  end component;

begin

```

```

u1      : detecto port map(reset, clk,
enter, secr, clr, temp2, temp3, temp4, temp5, key,
door, temp1);

u2      : change port map(reset, clk,
temp1, clr, key, temp2, temp3, temp4, temp5);

u3      : displayout port map(key,
reset, enter, secr, clr, seg_display);

end sample;
    
```

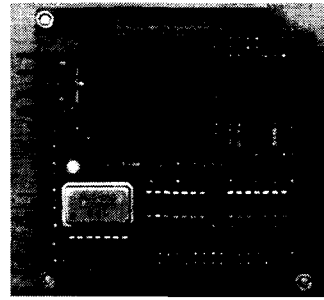


그림 5. FPGA 구현보드

### V. 전자 키 구현

설계한 전자 키를 EEPROM인 'AT17C010A'에 다운로드하였으며, PLD(Programmable Logic Devices)와 FPGA(Field-Programmable Gate Array) 두 종류의 사용자 프로그램 IC(Integrated Circuit) 중 보다 많은 논리 게이트의 구현을 위해서 FPGA 구조를 가지는 ALTERA사의 'EPF10K10LC84-4'를 이용하여 구현하였다.

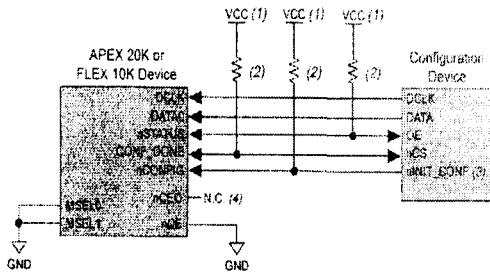


그림 3. Configuration Device Scheme Circuit

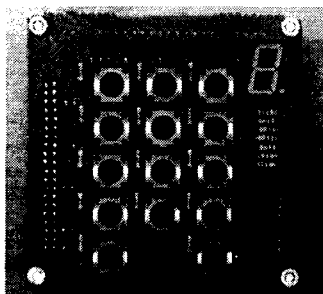


그림 4. 비밀번호 입력을 위한 키패드

[그림3]은 본 논문에 사용된 'EPF10K10LC84-4'와 'AT17C010A' Configuration한 Scheme Circuit을 나타내었다. [그림4]와 [그림5]는 구현된 시스템의 키패드 입력부분과 구현된 보드를 나타내었다.

### VI. 결론

본 논문은 ALTERA사의 MAX+plus II 소프트웨어를 이용하여 Hardware Language인 VHDL로 FPGA 구조를 가지는 ALTERA사의 'EPF10K10LC84-4'에 구현하였다.

보안을 위하여 사용자가 버튼을 누르는 동안에만 7segment에 불이 켜지도록 설계하기 위해 7segment는 하나만을 사용하였다. 그 결과, 사용된 GATE의 수를 4,800개의 GATE로 줄일 수 있었다.

사용자가 문을 열기 위해서는 비밀번호 4자리를 입력하여야 하고, 비밀번호를 바꾸기 위해서는 비밀번호 4자리 입력 후, SECR 버튼을 누르고 새로운 비밀번호 4자리를 입력한 뒤 CLR 버튼을 누르면 된다.

하지만 전원이 꺼졌다가 다시 켜지게 되면 저장되어 있는 비밀번호가 초기화되는 단점이 생기는 것을 알 수 있었다.

향후의 추가적 연구개발은 새로운 비밀번호를 저장할 수 있는 ROM을 외부에 연결하여 전원이 꺼졌다 다시 켜지는 경우라도 비밀번호가 초기화되지 않고 그대로 유지되도록 연구되어야 할 것으로 본다.

### 참고문헌

- [1][2] 박세현, "VHDL기본과 활용", 도서출판 그린, 1998.
- [3] 박용수, 황금주, "VHDL과 디지털 논리회로 설계", 히드텍, 2001.
- [4] 김영철, 정연모, 조중휘, 홍윤식, "디지털 시스템 설계를 위한 VHDL", 홍릉과학출판사, 1998.
- [5] Cohen Ben, "VHDL coding styles methodologies", Kluwer Academic Publishers, 1999.