IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.021 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 2, February 2025

DOI: 10.17148/IJIREEICE.2025.13209

A novel, time controlled, four digit PIN input system with reset function, for multiple security applications, using VHDL and FPGAs

Dr Evangelos I. Dimitriadis[1], Leonidas Dimitriadis[2]

Department of Computer, Informatics and Telecommunications Engineering, International Hellenic University, End of Magnisias Str, 62124 Serres Greece[1]

Undergraduate student, Department of Information and Electronic Engineering, International Hellenic University, 57400, Sindos Thessaloniki, Greece[2]

Abstract: A novel, time controlled, four digit PIN input system, suitable for many security applications, was programmed and put in use, for the first time, in this work. The DE10-Lite FPGA board was used here. Administrator has the programing ability of creating PIN1 and PIN2 and setting the time needed for user to correctly create one of the above PINs. User handles FPGA's SW switches in conjunction with key buttons, in order to achieve the correct creation of one pin, in a few minutes time, also being able to see his creation in seven segment displays. If PIN input is wrong FA (FALSE) is displayed, while at the opposite case all LEDs light up. In the case that user's time runs out without correct PIN creation, system locks up and administrator must press a third key button connected to FPGA via a pull-down resistance, if he wants to provide user with extra time.

Keywords: PIN input, FPGA, VHDL, time controlled, reset function, security applications.

INTRODUCTION

In the recent years FPGAs have attracted attention of researchers for industrial as well as other applications. [1-11] The combination of software and hardware in FPGAs is their main advantage, since it enables hardware programming for a series of applications. VHDL and Verilog are the most used languages for FPGAs' programing and VHDL is the one used in our work.

We found out that in spite of all work done concerning FPGAs, there is nothing about PIN input security systems. We present here a time controlled, four digit PIN input system, capable of being used in multiple security systems applications. Our work provides a system, which can be implemented in many types of FPGAs and depending on critical time value set by the programer-administrator for user to create the right PIN number, can be used to protect and control the function of many security systems protecting areas, houses, machines, laboratories, etc.

Administrator has also the ability of setting PIN1 and PIN2 of the system and if user fails to create the right PIN within the given time, administrator can decide whether to give extra time by pressing key button acting as reset.

DESIGN OVERVIEW AND OPERATION OF THE PIN INPUT SYSTEM

Figure 1 presents design overview and operational units of our FPGA four digit PIN input system.
The PIN input system starts operating as soon as the VHDL program is sent via USB Blaster interface, to FPGA chip, with simultaneous start of time measurement. It is programed to display 0000EI, where EI means Enter Input, as shown in Figure 2. User has a few minutes time in order to create correctly one of the two PINS (PIN1 or PIN2) set by the administrator. It is obvious that administrator can change the above PINS whenever he wants, as well as the time given to user, in order to create PIN.

DOI: 10.17148/IJIREEICE.2025.13209

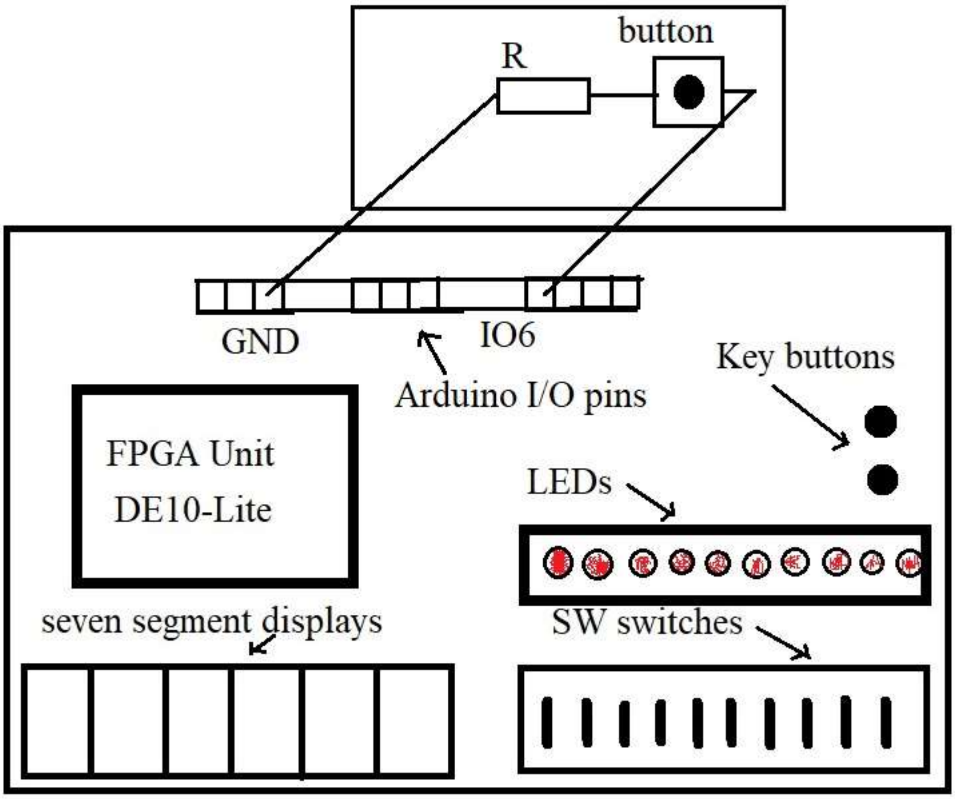

Figure 1: Design overview and operational units of the PIN input system presented in this work.

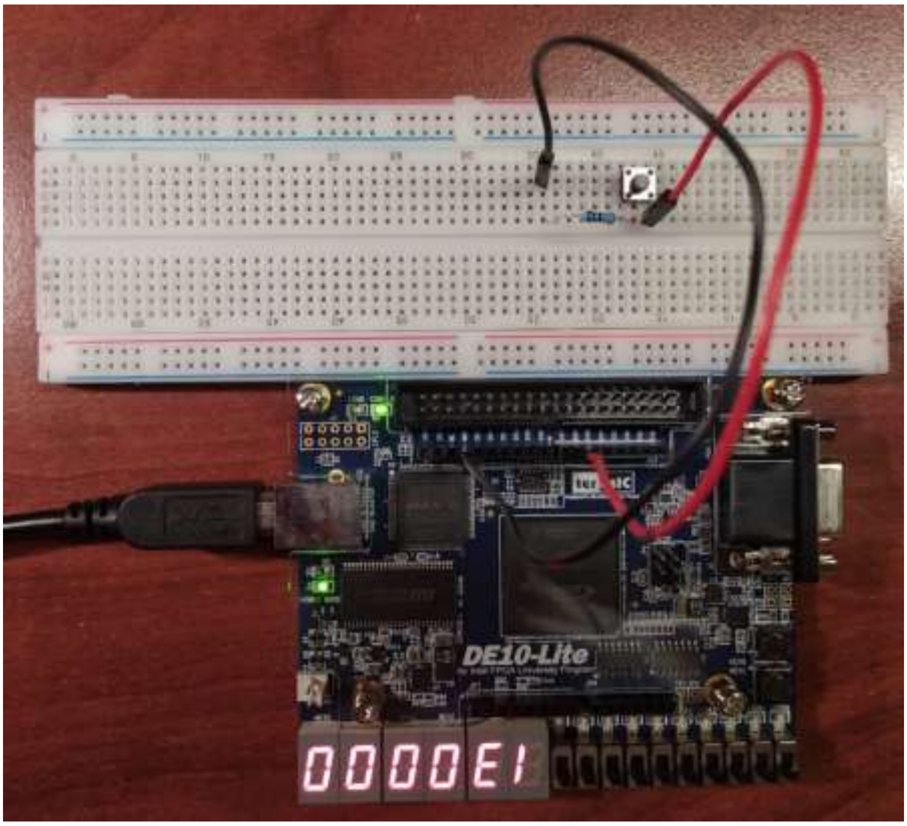

Figure 2: FPGA DE10-Lite used in the PIN input system.

DOI: 10.17148/IJIREEICE.2025.13209

During his attempt user can see the created PIN in seven segment displays with a simultaneous message of – when correct and FA (FALSE) when wrong, as shown in Figure 3.

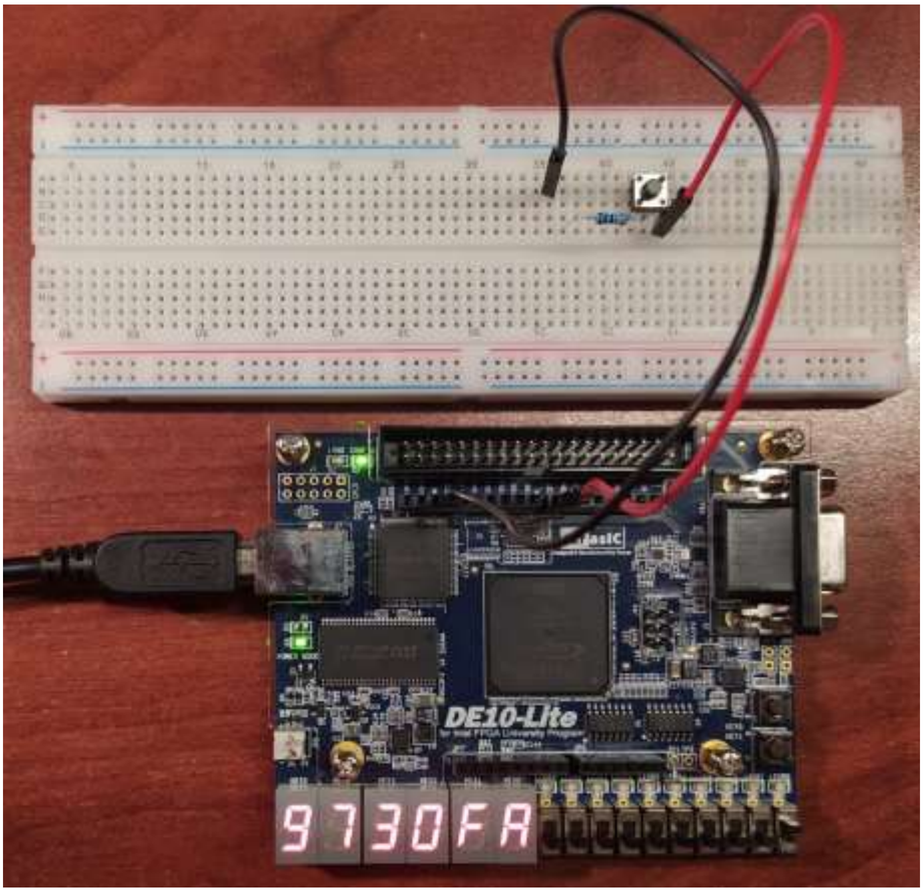

Figure 3: FPGA’s DE10-Lite behavior when PIN input is wrong.

If user succeeds correct PIN creation all LEDs also light up, as presented in Figure 4.
In the case that time runs out and user fails to create the right PIN number, consequently system locks up and user is unable to create another PIN. He must contact the administrator in order to ask for extra time. If the administrator agrees, then he can press external third key button connected via a pull-down resistance of 330 Ohm to FPGA board using Arduino I/O pins, shown in Figure 1.

It must be mentioned that user tries to create PIN number with the sequence presented below in Table1, using key buttons existing at the right of the FPGA board (Figure 1), in conjuction with SW switches of the board. Each of SW switches corresponds to one of
0-9 numbers.

DOI: 10.17148/IJIREEICE.2025.13209

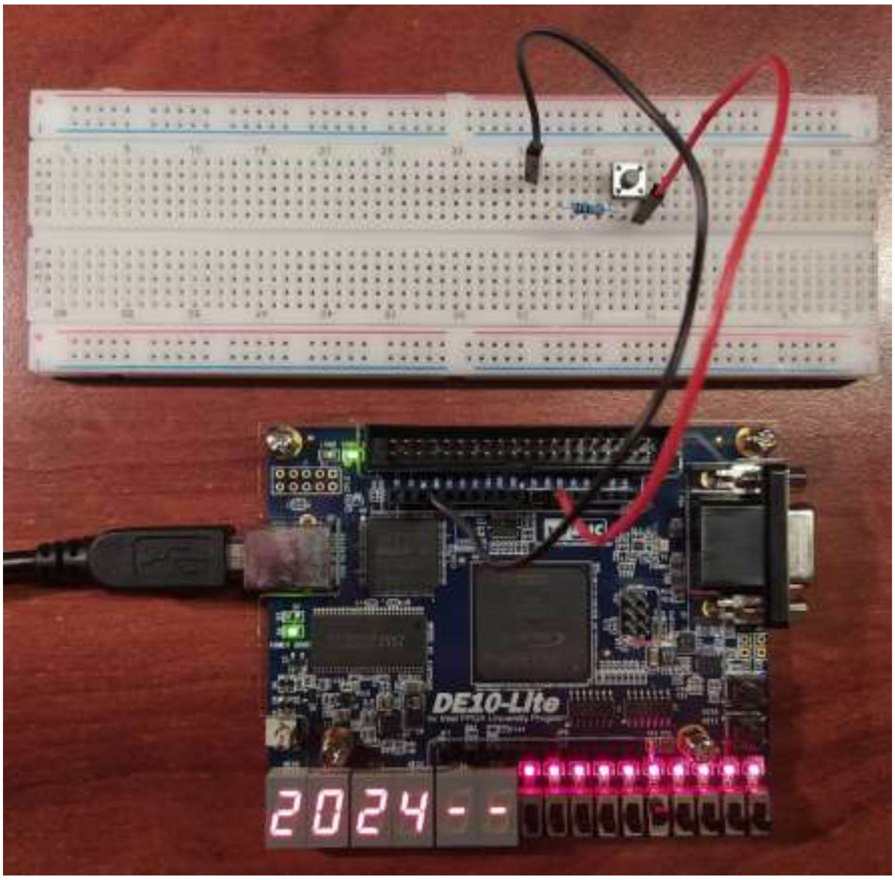

Figure 4: FPGA’s DE10-Lite behavior when PIN input is right.

Table1

key button 0	key button 1	SW switches	PIN digit
1	1	Only one to ON	1st
0	1	Only one to ON	2nd
1	0	Only one to ON	3rd
0	0	Only one to ON	4th

The operation of our four digit PIN input system presented above, is shown in Figure 5 presenting the flowchart related to the functionality of the VHDL program used in this work.

Programing the four digit PIN input system

We used Quartus Prime Lite Edition 21.1.1 for creating the VHDL programs of our four digit PIN input system. SW switches and key buttons act as input for the FPGA chip.

DOI: 10.17148/IJIREEICE.2025.13209

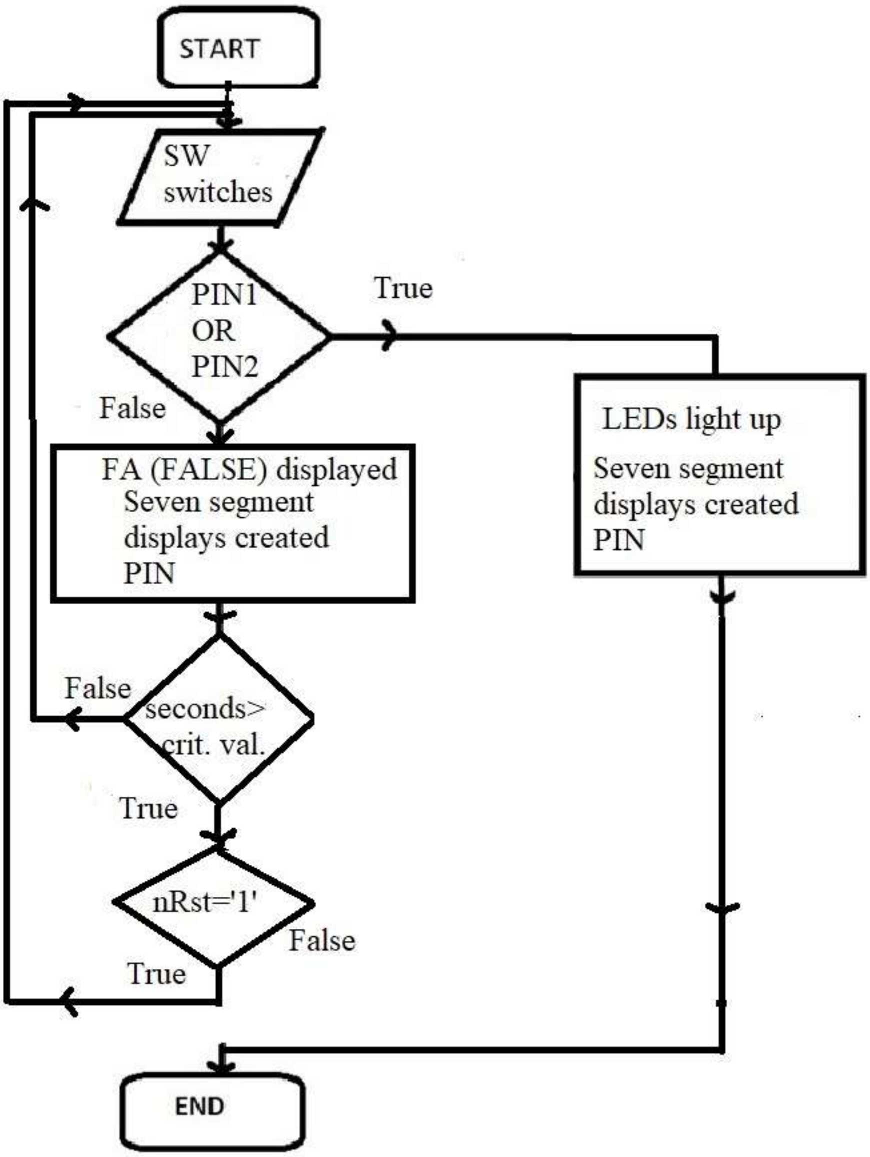

Figure 5: Flowchart related to the functionality of the VHDL program used in this work.

Output units are LEDs and seven segment displays.
The following first part of our program shows all the above mentioned statements.

```
LIBRARY ieee;
USE ieee.std_logic_1164.all;
use ieee.numeric_std.all;
ENTITY Project_pin_system IS
generic(ClockFrequencyHz : integer:=50000000);
PORT(a1: buffer integer range 0 to 9999;--final pin
c: buffer integer range 0 to 9; --first pin number (from right to left)
d: buffer integer range 0 to 9; --second pin number (from right to left)
e: buffer integer range 0 to 9; --third pin number (from right to left)
f: buffer integer range 0 to 9; --fourth pin number (from right to left)
g: buffer std_logic_vector (3 DOWNTO 0);
CLK: IN std_logic;
nRst    : in std_logic; -- reset
Seconds : inout integer;
sw9: IN std_logic;
sw8: IN std_logic;
sw7: IN std_logic;
sw6: IN std_logic;
sw5: IN std_logic;
```

DOI: 10.17148/IJIREEICE.2025.13209

```
sw4: IN std_logic;
sw3: IN std_logic;
sw2: IN std_logic;
sw1: IN std_logic;
sw0: IN std_logic;
key0: IN std_logic;
key1: IN std_logic;
ex1: OUT std_logic_vector(7 DOWNTO 0);
ex2: OUT std_logic_vector(7 DOWNTO 0);
ex3: OUT std_logic_vector(7 DOWNTO 0);
ex4: OUT std_logic_vector(7 DOWNTO 0);
ex5: OUT std_logic_vector(7 DOWNTO 0);
ex6: OUT std_logic_vector(7 DOWNTO 0);
led1: out std_logic;
led2: out std_logic;
led3: out std_logic;
led4: out std_logic;
led5: out std_logic;
led6: out std_logic;
led7: out std_logic;
led8: out std_logic;
led9: out std_logic;
led10: out std_logic
);
END Project_pin_system;
```

The second part of the program contains the signals declared in the architecture. Signal state_a1 corresponds to user created four digit PIN.

```
ARCHITECTURE behaviour OF Project pin_system IS
signal state_LED: std_logic;
signal state_e: std_logic_vector (3 DOWNTO 0);
signal state_g: std_logic_vector (3 DOWNTO 0);
signal state_a1: integer range 0 to 9999;
constant pin1: integer range 0 to 9999:=2024;
constant pin2: integer range 0 to 9999:=2023;
signal pin1_state: integer range 0 to 9999;
signal pin2_state: integer range 0 to 9999;
-- Signal for counting clock periods
   signal Ticks : integer;
```

The third part of the program presents the main functioning of the timer used in this work and Seconds acts as the main parameter controlling the PIN input system.

```
Begin
 process(CLK) is
   begin
      if rising_edge(CLK) then
         -- If the negative reset signal is active
         if nRst = '0' then
            Ticks   <= 0;
            Seconds <= 0;
                     else
            -- True once every second
            if Ticks = ClockFrequencyHz - 1 then
               Ticks <= 0;
                  Seconds <= Seconds + 1;
            else
               Ticks <= Ticks + 1;
            end if;
         end if;
      end if;
   end process;
```

The fourth part of the program contains four similar processes, one for each digit, in order to help user creating input PIN.

```
process (nRst,Seconds,CLK,c,key0,key1,sw9,sw8,sw7,sw6,sw5,sw4,sw3,sw2,sw1,sw0)
begin
IF nRst='1'THEN
IF CLK'EVENT AND CLK='1' AND  (Seconds>=0 and Seconds<=30)   THEN
IF (key0='1' and key1='1') THEN
IF (sw9='1'and sw8='0'and sw7='0'and sw6='0'and sw5='0'and sw4='0'
and sw3='0'and sw2='0' and sw1='0' and sw0='0') then c<=9;
ELSIF (sw9='0' and sw8='1'and sw7='0'and sw6='0'and sw5='0'and sw4='0'
 and sw3='0'and sw2='0'and sw1='0' and sw0='0') then c<=8;
ELSIF (sw9='0' and sw8='0'and sw7='1'and sw6='0'and sw5='0'and sw4='0'
 and sw3='0'and sw2='0'and sw1='0' and sw0='0') then c<=7;
ELSIF (sw9='0' and sw8='0'and sw7='0'and sw6='1'and sw5='0'and sw4='0'
 and sw3='0'and sw2='0'and sw1='0' and sw0='0') then c<=6;
ELSIF (sw9='0' and sw8='0'and sw7='0'and sw6='0'and sw5='1'and sw4='0'
 and sw3='0'and sw2='0'and sw1='0' and sw0='0') then c<=5;
ELSIF (sw9='0' and sw8='0'and sw7='0'and sw6='0'and sw5='0'and sw4='1'
 and sw3='0'and sw2='0'and sw1='0' and sw0='0') then c<=4;
ELSIF (sw9='0' and sw8='0'and sw7='0'and sw6='0'and sw5='0'and sw4='0'
 and sw3='1'and sw2='0'and sw1='0' and sw0='0') then c<=3;
ELSIF (sw9='0' and sw8='0'and sw7='0'and sw6='0'and sw5='0'and sw4='0'
 and sw3='0'and sw2='1'and sw1='0' and sw0='0') then c<=2;
ELSIF (sw9='0' and sw8='0'and sw7='0'and sw6='0'and sw5='0'and sw4='0'
 and sw3='0'and sw2='0'and sw1='1' and sw0='0') then c<=1;
ELSIF (sw9='0' and sw8='0'and sw7='0'and sw6='0'and sw5='0'and sw4='0'
 and sw3='0'and sw2='0'and sw1='0' and sw0='1') then c<=0;
end if;
end if;
end if;
end if;
end process;
process (nRst,Seconds,CLK,d,key0,key1,sw9,sw8,sw7,sw6,sw5,sw4,sw3,sw2,sw1,sw0)
begin
IF nRst='1' THEN
IF CLK'EVENT AND CLK='1' AND (Seconds>=0 and Seconds<=30)  THEN
IF (key0='0' and key1='1') THEN
IF (sw9='1' and sw8='0'and sw7='0'and sw6='0'and sw5='0'and sw4='0'
 and sw3='0'and sw2='0'and sw1='0' and sw0='0') then d<=9;
ELSIF  (sw9='0' and sw8='1'and sw7='0'and sw6='0'and sw5='0'and sw4='0'
 and sw3='0'and sw2='0'and sw1='0' and sw0='0') then d<=8;
ELSIF (sw9='0' and sw8='0'and sw7='1'and sw6='0'and sw5='0'and sw4='0'
 and sw3='0'and sw2='0'and sw1='0' and sw0='0') then d<=7;
ELSIF (sw9='0' and sw8='0'and sw7='0'and sw6='1'and sw5='0'and sw4='0'
 and sw3='0'and sw2='0'and sw1='0' and sw0='0') then d<=6;
ELSIF (sw9='0' and sw8='0'and sw7='0'and sw6='0'and sw5='1'and sw4='0'
 and sw3='0'and sw2='0'and sw1='0' and sw0='0') then d<=5;
ELSIF (sw9='0' and sw8='0'and sw7='0'and sw6='0'and sw5='0'and sw4='1'
 and sw3='0'and sw2='0'and sw1='0' and sw0='0') then d<=4;
ELSIF (sw9='0' and sw8='0'and sw7='0'and sw6='0'and sw5='0'and sw4='0'
 and sw3='1'and sw2='0'and sw1='0' and sw0='0') then d<=3;
ELSIF (sw9='0' and sw8='0'and sw7='0'and sw6='0'and sw5='0'and sw4='0'
 and sw3='0'and sw2='1'and sw1='0' and sw0='0') then d<=2;
ELSIF (sw9='0' and sw8='0'and sw7='0'and sw6='0'and sw5='0'and sw4='0'
 and sw3='0'and sw2='0'and sw1='1' and sw0='0') then d<=1;
ELSIF (sw9='0' and sw8='0'and sw7='0'and sw6='0'and sw5='0'and sw4='0'
 and sw3='0'and sw2='0'and sw1='0' and sw0='1') then d<=0;
end if;
end if;
end if;
end if;
end process;
process (nRst,Seconds,CLK,e,key0,key1,sw9,sw8,sw7,sw6,sw5,sw4,sw3,sw2,sw1,sw0)
begin
IF nRst='1' THEN
IF CLK'EVENT AND CLK='1' AND (Seconds>=0 and Seconds<=30)  THEN
IF (key0='1' and key1='0') THEN
IF (sw9='1' and sw8='0'and sw7='0'and sw6='0'and sw5='0'and sw4='0'
 and sw3='0'and sw2='0'and sw1='0' and sw0='0') then e<=9;
ELSIF  (sw9='0' and sw8='1'and sw7='0'and sw6='0'and sw5='0'and sw4='0'
 and sw3='0'and sw2='0'and sw1='0' and sw0='0') then e<=8;
ELSIF (sw9='0' and sw8='0'and sw7='1'and sw6='0'and sw5='0'and sw4='0'
 and sw3='0'and sw2='0'and sw1='0' and sw0='0') then e<=7;
ELSIF (sw9='0' and sw8='0'and sw7='0'and sw6='1'and sw5='0'and sw4='0'
 and sw3='0'and sw2='0'and sw1='0' and sw0='0') then e<=6;
ELSIF (sw9='0' and sw8='0'and sw7='0'and sw6='0'and sw5='1'and sw4='0'
 and sw3='0'and sw2='0'and sw1='0' and sw0='0') then e<=5;
ELSIF (sw9='0' and sw8='0'and sw7='0'and sw6='0'and sw5='0'and sw4='1'
 and sw3='0'and sw2='0'and sw1='0' and sw0='0') then e<=4;
ELSIF (sw9='0' and sw8='0'and sw7='0'and sw6='0'and sw5='0'and sw4='0'
 and sw3='1'and sw2='0'and sw1='0' and sw0='0') then e<=3;
ELSIF (sw9='0' and sw8='0'and sw7='0'and sw6='0'and sw5='0'and sw4='0'
 and sw3='0'and sw2='1'and sw1='0' and sw0='0') then e<=2;
ELSIF (sw9='0' and sw8='0'and sw7='0'and sw6='0'and sw5='0'and sw4='0'
 and sw3='0'and sw2='0'and sw1='1' and sw0='0') then e<=1;
```

DOI: 10.17148/IJIREEICE.2025.13209

```
ELSIF (sw9='0' and sw8='0'and sw7='0'and sw6='0'and sw5='0'and sw4='0'
 and sw3='0'and sw2='0'and sw1='0' and sw0='1') then e<=0;
end if;
end if;
end if;
end if;
end process;
process (nRst,Seconds,CLK,f,key0,key1,sw9,sw8,sw7,sw6,sw5,sw4,sw3,sw2,sw1,sw0)
begin
IF nRst='1' THEN
IF CLK'EVENT AND CLK='1' AND (Seconds>=0 and Seconds<=30)  THEN
IF (key0='0' and key1='0') THEN
IF (sw9='1' and sw8='0'and sw7='0'and sw6='0'and sw5='0'and sw4='0'
 and sw3='0'and sw2='0'and sw1='0' and sw0='0') then f<=9;
ELSIF  (sw9='0' and sw8='1'and sw7='0'and sw6='0'and sw5='0'and sw4='0'
 and sw3='0'and sw2='0'and sw1='0' and sw0='0') then f<=8;
ELSIF (sw9='0' and sw8='0'and sw7='1'and sw6='0'and sw5='0'and sw4='0'
 and sw3='0'and sw2='0'and sw1='0' and sw0='0') then f<=7;
ELSIF (sw9='0' and sw8='0'and sw7='0'and sw6='1'and sw5='0'and sw4='0'
 and sw3='0'and sw2='0'and sw1='0' and sw0='0') then f<=6;
ELSIF (sw9='0' and sw8='0'and sw7='0'and sw6='0'and sw5='1'and sw4='0'
 and sw3='0'and sw2='0'and sw1='0' and sw0='0') then f<=5;
ELSIF (sw9='0' and sw8='0'and sw7='0'and sw6='0'and sw5='0'and sw4='1'
 and sw3='0'and sw2='0'and sw1='0' and sw0='0') then f<=4;
ELSIF (sw9='0' and sw8='0'and sw7='0'and sw6='0'and sw5='0'and sw4='0'
 and sw3='1'and sw2='0'and sw1='0' and sw0='0') then f<=3;
ELSIF (sw9='0' and sw8='0'and sw7='0'and sw6='0'and sw5='0'and sw4='0'
 and sw3='0'and sw2='1'and sw1='0' and sw0='0') then f<=2;
ELSIF (sw9='0' and sw8='0'and sw7='0'and sw6='0'and sw5='0'and sw4='0'
 and sw3='0'and sw2='0'and sw1='1' and sw0='0') then f<=1;
ELSIF (sw9='0' and sw8='0'and sw7='0'and sw6='0'and sw5='0'and sw4='0'
 and sw3='0'and sw2='0'and sw1='0' and sw0='1') then f<=0;
end if;
end if;
end if;
end if;
end process;
```

Fifth part of our program shows the codes for presenting the results to seven segment displays and LEDs and also calculating in decimal system, the user created PIN.

```
state_a1<=(f*10**3)+(e*10**2)+(d*10**1)+(c*10**0);
a1<= state_a1;
WITH c SELECT
ex4<= "11000000" when 0, -- 0, active low i.e. 0: display & 1: no display
    "11111001" when 1, -- 1
    "10100100" when 2, -- 2
    "10110000" when 3, -- 3
    "10011001" when 4, -- 4
    "10010010" when 5, -- 5
    "10000010" when 6, -- 6
    "11111000" when 7, -- 7
    "10000000" when 8, -- 8
    "10010000" when 9, -- 9
    "11111111" when others;

WITH d SELECT
ex3<= "11000000" when 0, -- 0, active low i.e. 0: display & 1: no display
    "11111001" when 1, -- 1
    "10100100" when 2, -- 2
    "10110000" when 3, -- 3
    "10011001" when 4, -- 4
    "10010010" when 5, -- 5
    "10000010" when 6, -- 6
    "11111000" when 7, -- 7
    "10000000" when 8, -- 8
    "10010000" when 9, -- 9
    "11111111" when others;

WITH e SELECT
ex2<= "11000000" when 0, -- 0, active low i.e. 0: display & 1: no display
    "11111001" when 1, -- 1
    "10100100" when 2, -- 2
    "10110000" when 3, -- 3
    "10011001" when 4, -- 4
    "10010010" when 5, -- 5
    "10000010" when 6, -- 6
    "11111000" when 7, -- 7
    "10000000" when 8, -- 8
    "10010000" when 9, -- 9
    "11111111" when others;

WITH f SELECT
ex1<= "11000000" when 0, -- 0, active low i.e. 0: display & 1: no display
    "11111001" when 1, -- 1
    "10100100" when 2, -- 2
    "10110000" when 3, -- 3
    "10011001" when 4, -- 4
    "10010010" when 5, -- 5
    "10000010" when 6, -- 6
    "11111000" when 7, -- 7
    "10000000" when 8, -- 8
    "10010000" when 9, -- 9
    "11111111" when others;

pin1_state<=pin1;
pin2_state<=pin2;

process (a1,pin1_state, pin2_state, nRst)
begin
IF (a1=0) THEN
  state_LED <= '0';
  state_g <="0011";
end if;
IF (a1/=0) AND (a1=pin1_state) then
  state_LED <= '1';
  state_g <="0001";
 else
  IF (a1/=0) AND (a1/=pin1_state) then
  state_LED <= '0';
  state_g <="0000";
  IF (a1=pin2_state) then
  state_LED <= '1';
  state_g <="0001";
 else
  state_LED <= '0';
  state_g <="0000";
  end if;
  end if;
  end if;
 end process;
led1 <= state_LED;
led2 <= state_LED;
led3 <= state_LED;
led4 <= state_LED;
led5 <= state_LED;
led6 <= state_LED;
led7 <= state_LED;
led8 <= state_LED;
led9 <= state_LED;
led10 <= state_LED;
g <= state_g;

WITH g SELECT
ex5<= "10111111" when "0001",
    "10001110" when "0000",
                    "10000110" when "0011",
              "11111111" when others;

WITH g SELECT
ex6<= "10111111" when "0001",
    "10001000" when "0000",
                    "11001111" when "0011",
              "11111111" when others;
END behaviour;
```

DOI: 10.17148/IJIREEICE.2025.13209

CONCLUSION

We presented here for the first time, a four digit, time controlled, PIN input system based on the DE10-Lite FPGA board. Our system allows user to create in a few minutes given time, one of the two PINS set by the programmer. Created PIN is presented in seven segment displays and user attempts are time limited. System locks up if time runs out without a successful PIN creation and administrator has the ability of providing extra time by using key button connected to FPGA via a pull-down resistance. All LEDs light up in case of user correct PIN creation, otherwise FA (FALSE) is shown in seven segment displays. Administrator has the ability of changing time limitation period, as well as PIN1 and PIN2, making our system suitable for a variety of security applications.

REFERENCES

[1]. M. Yussup, M.M. Ibrahim, L. Lombigit, N.A.A. Rahman and M.R.M. Zin, “Implementation of data acquisition interface using on-board field-programmable gate array (FPGA) universal serial bus (USB) link”, Advanc. in Nuclear Research and Energy Development, AIP Conf. Proc. 1584, 69-72 (2014), doi: 10.1063/1.4866106

[2]. A. Gujar, International Journal of Computer Science and Information Technologies, “Image Encryption using AES Algorithm based on FPGA”, vol 5, (5), 2014, 6853-6859

[3]. S. Singh, A.K. Saini, R. Saini, I.J. Image, Graphics and Signal Processing, “Interfacing the Analog Camera with FPGA Board for Real-time Video Acquisition” 2014, 4, 32-38, DOI: 10.5815/ijigsp.2014.04.04

[4]. S. Muthukrishnan and R. Priyadharsini, International Journal of Computer Science and Mobile Computing, “32-Bit RISC and DSP System Design in an FPGA” vol 3, issue 12, Dec. 2014, pg. 361-368

[5]. L. Chen, Y. Chang, L. Yan, IEEE Transactions on Geoscience and Remote Sensing, “On-orbit real-time variational image destriping: FPGA architecture and implementation”, 10.1109/tgrs.2022.3140428, 2022, pp. 1-1

[6]. S. Yarlagadda, S. Kaza, A. Tummala, E. Babu, R. Prabhakar, Information Technology in Industry, “The reduction of Crosstalk in VLSI due to parallel bus structure using Data Compression Bus Encoding technique implemented on Artix 7 FPGA Architecture”, 10.17762/itii.v9i1.151, 2021, Vol 9 (1), pp. 456-460

[7]. C. Du, Y. Yamaguchi, Electronics, “High-Level Synthesis Design for Stencil Computations on FPGA with High Bandwidth Memory”, 2020, 9(8), 1275; https://doi.org/10.3390/electronics9081275

[8]. X. Hao, C. Lin and Q. Wu, Electronics, “A Parallel Timing Synchronization Structure in Real-Time High Transmission Capacity Wireless Communication Systems”, 2020, 9(4), 652; https://doi.org/10.3390/electronics9040652

[9]. P. A. Bawiskar, R.K. Agrawal, International Journal of Innovative Research in Science, Engineering and Technology, “FPGA Based Home Security System” vol.4, issue 12, Dec. 2015, p. 12865-12869, DOI: 10.15680/IJIRSET.2015.0412139

[10]. K. Saroch, A. Sharma, IOSR Journal of Electronics and Communication Engineering, “FPGA Based System Login Security Lock Design Using Finite State Machine” vol 5, issue 3, Mar.-Apr. 2013, pp 70-75

[11]. R.S. Parikh, Int. Journal of Engineering Research and Application, “Alarm System Implementation on Field Programmable Gate Array” vol 8, issue 1, Jan. 2018, pp 01-04.