



**HO CHI MINH UNIVERSITY OF TECHNOLOGY
FACULTY OF COMPUTER SCIENCE AND ENGINEERING
DEPARTMENT OF MECHATRONICS**



LOGIC DESIGN WITH HDL LAB

STUDENT REPORT

Class: CC01

Group: 04

Student's name: Trịnh Sơn Lâm 1852502

Nguyễn Trọng Tính 1752545

Lê Bá Thành 1852739

Hoàng Nhật Quang 1852691

Lecturer Ph.D Phạm Quốc Cường

APRIL 1, 2022

WEEK 2 Combinational Logic Circuit

Exercise:

Exercise 1

Design a circuit that has one 4-bit input and 4-bit output with functions as follow (all outputs are active HIGH):

Output 0: active when there are even number of bit 1 in the input.

Output 1: active when there are only 1 bit 1 in the input.

Output 2: active when there is no bit 1 in the input.

Output 3: active when all bit in the input are 1.

Write Verilog HDL RTL code and test bench for the design.

We have file code is **two_bit_reduction_operator.v** for idea to create **reduction_operator.v**(four_bit_reduction_operator)

And then we have RTL schematic circuit:

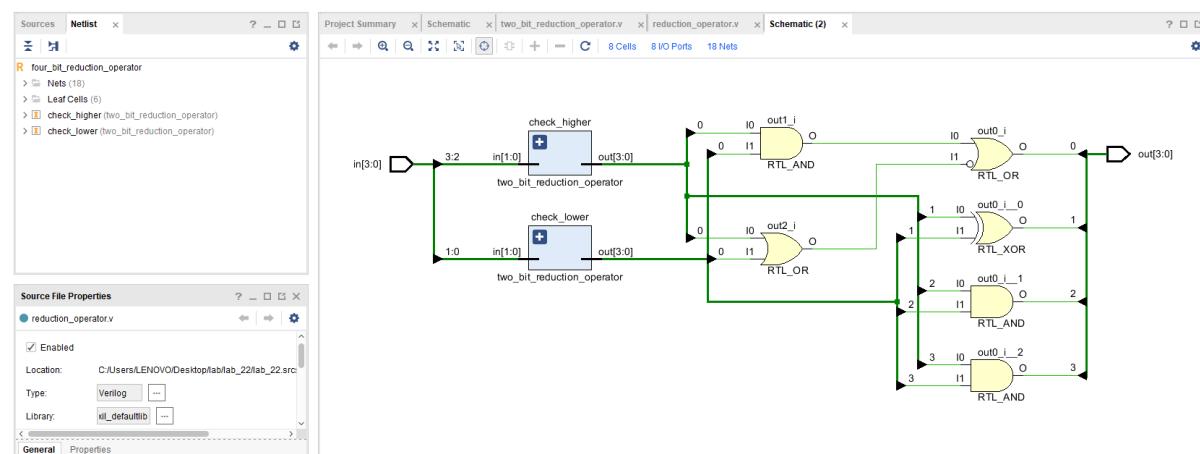


Figure 1: 4bit_reduction_operator in RTL schematic

We have file code for test bench of reduction operator is **four_bit_reduction_operator_tb.v**

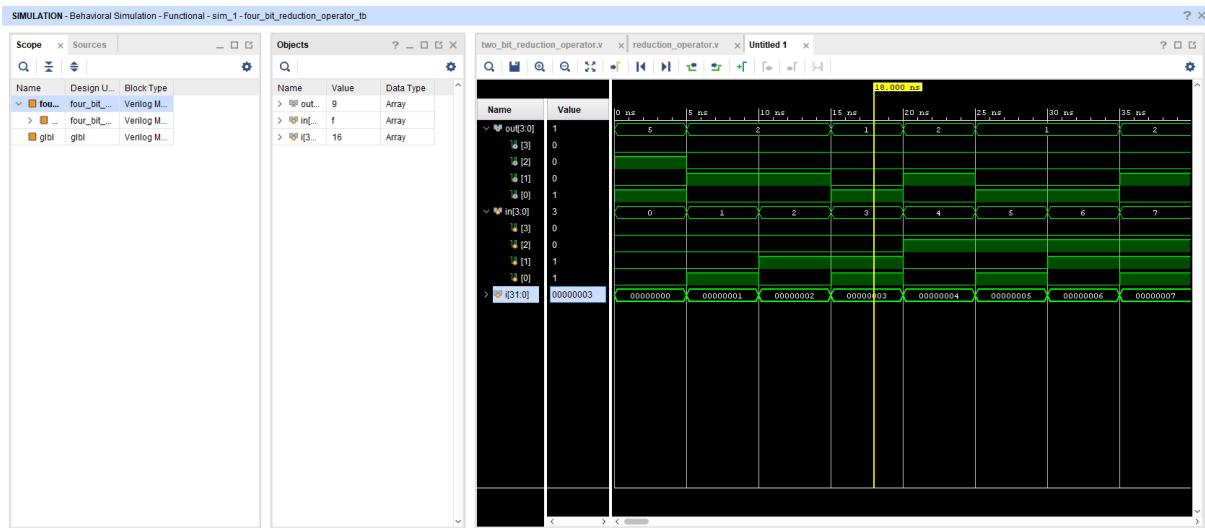


Figure 2: The waveform of test bench for reduction operator

Test the circuit on FPGA using buttons and LEDs.

We have some changes in constraint as follow:

```

22 : ## LEDs
23 : set_property -dict { PACKAGE_PIN R14 IOSTANDARD LVCMS33 } [get_ports { out[0] }]; #IO_L6N_TO_VREF_34 Sch=LED0
24 : set_property -dict { PACKAGE_PIN P14 IOSTANDARD LVCMS33 } [get_ports { out[1] }]; #IO_L6P_TO_34 Sch=LED1
25 : set_property -dict { PACKAGE_PIN M16 IOSTANDARD LVCMS33 } [get_ports { out[2] }]; #IO_L21N_T3_DQS_AD14N_35 Sch=LED2
26 : set_property -dict { PACKAGE_PIN M14 IOSTANDARD LVCMS33 } [get_ports { out[3] }]; #IO_L23P_T3_35 Sch=LED3
27 :
28 : ## Buttons
29 : set_property -dict { PACKAGE_PIN D19 IOSTANDARD LVCMS33 } [get_ports { in[0] }]; #IO_L4P_TO_35 Sch=BTN0
30 : set_property -dict { PACKAGE_PIN D20 IOSTANDARD LVCMS33 } [get_ports { in[1] }]; #IO_L4N_TO_35 Sch=BTN1
31 : set_property -dict { PACKAGE_PIN L20 IOSTANDARD LVCMS33 } [get_ports { in[2] }]; #IO_L9N_T1_DQS_AD3N_35 Sch=BTN2
32 : set_property -dict { PACKAGE_PIN L19 IOSTANDARD LVCMS33 } [get_ports { in[3] }]; #IO_L9P_T1_DQS_AD3P_35 Sch=BTN3
33 :

```

Figure 3: Assign pin for Arty-Z27 before running for FPGA

Then we have the generated bitstream file is **four_bit_reduction_operator.bit**

And we have some picture about led in board as follow:

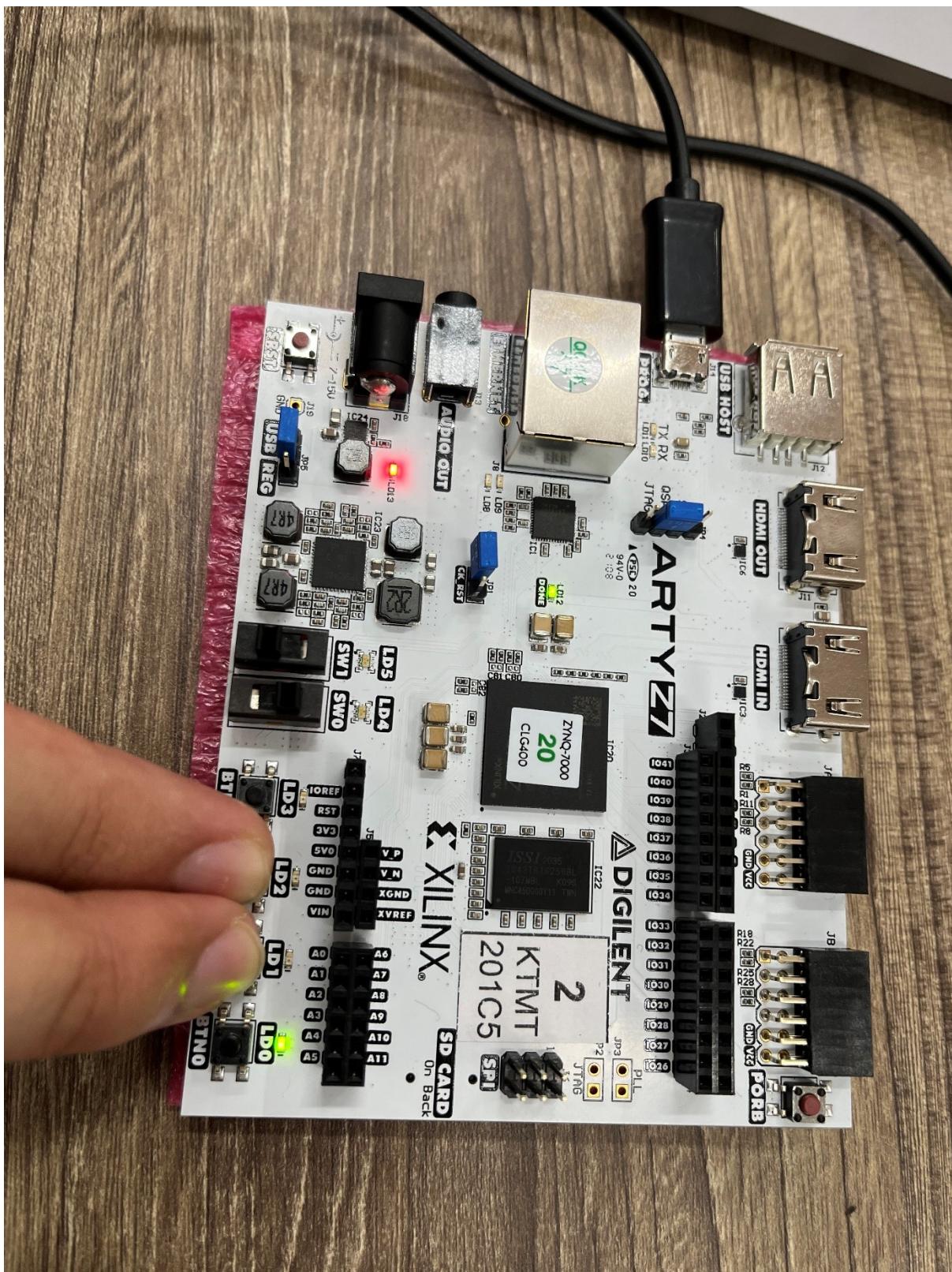


Figure 4: Case ouput0 active when input is 0110

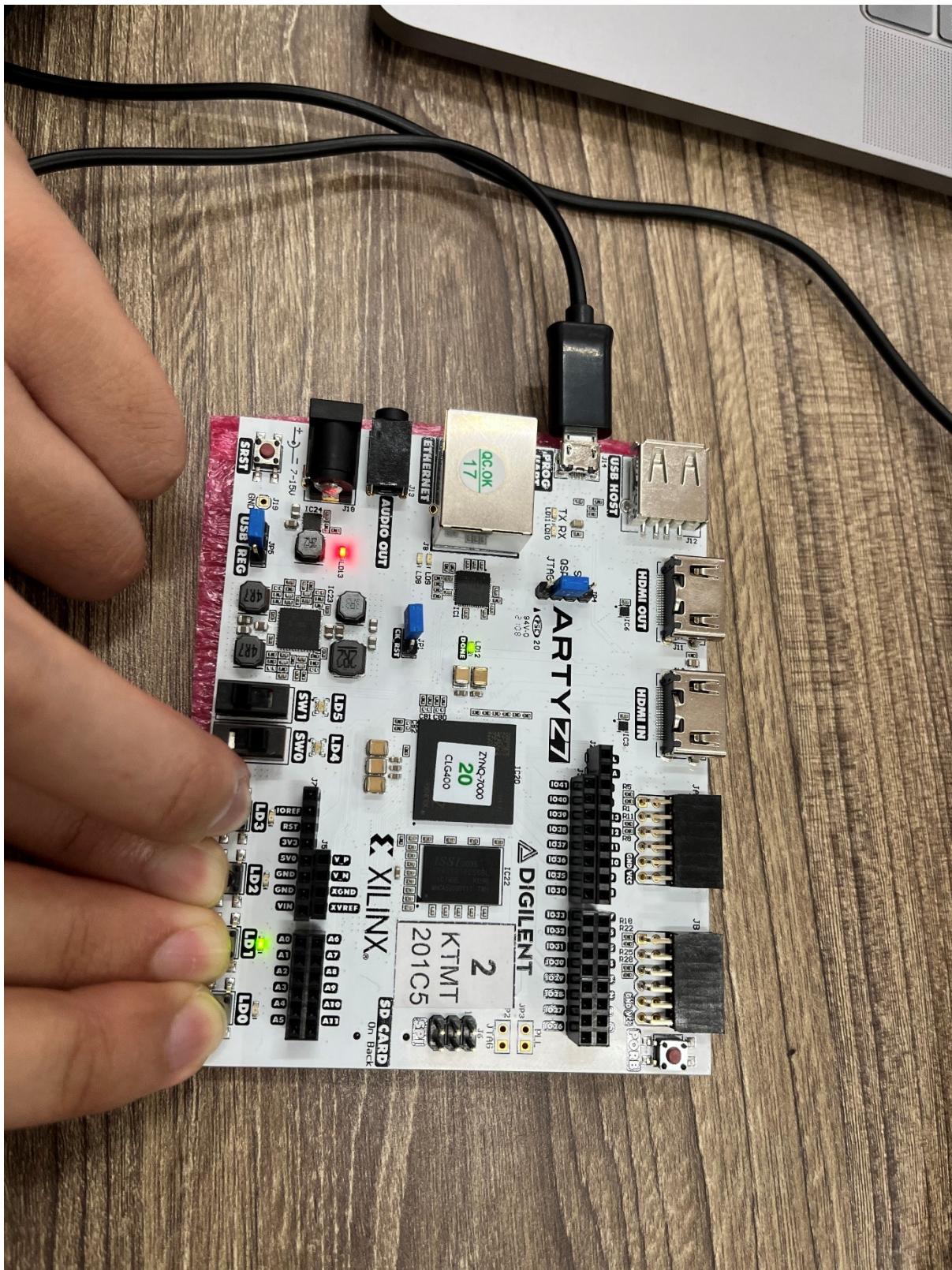


Figure 5: Case output1 active when input is 0010

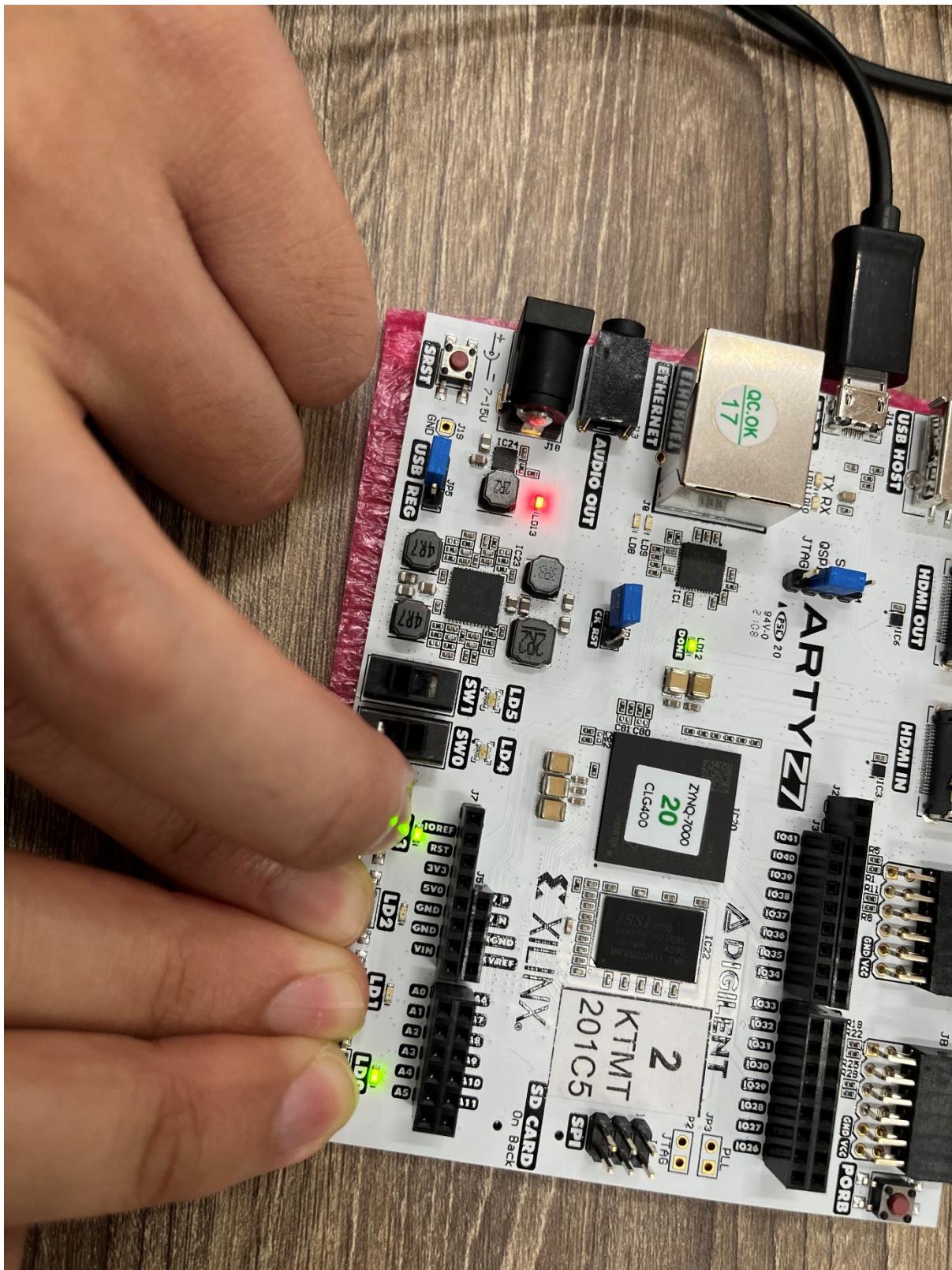


Figure 6: Case ouput0 and ouput 2 active when input is 1111

Exercise 2

Design a 7-segment LED decoder which will accept a 4-bit input and generate a 7-bit output.

Write a test bench to simulate the implemented circuit

We have file code is **bin2led7.v**

And then we have RTL schematic circuit:

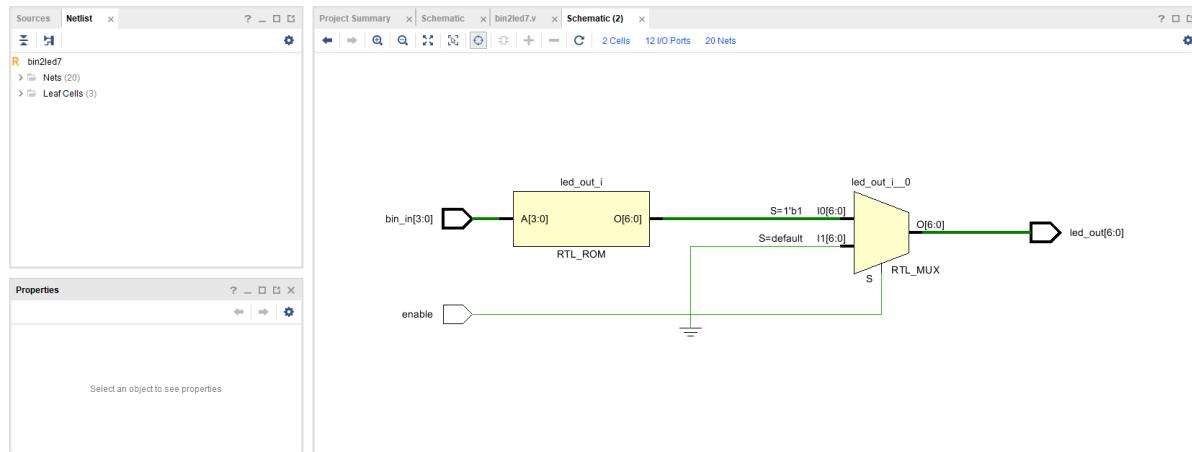


Figure 7: 7-segment LED in RTL schematic

And we have code file of test bench for 7-segment LED is **bin2led7_tb.v**

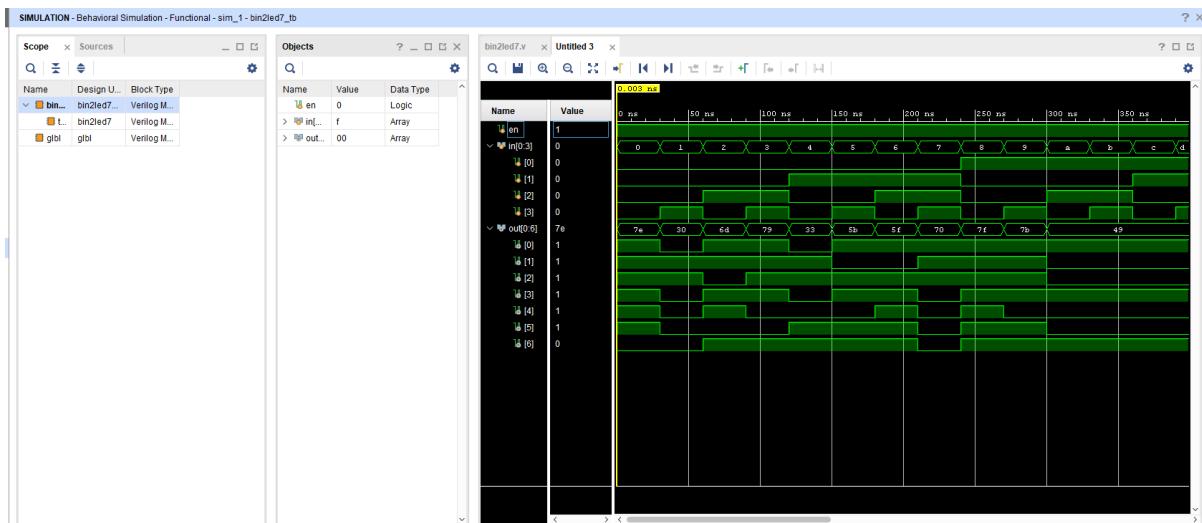


Figure 8: The waveform of test bench for 7-segment LED

Test the circuit on Arty-Z7 FPGA board using switches/buttons and external 7-seg LEDs.

We have some changes in constraint as follow:

```

7 #_set_property -dict { PACKAGE_PIN H16   IOSTANDARD LVCMS33 } [get_ports { clk }]; #IO_L13P_T2_MRCC_35 Sch=SYSCLK
8 #create_clock -add -name sys_clk_pin -period 8.00 -waveform { 0 4 } [get_ports { clk }];
9
10 ## Switches
11 #_set_property -dict { PACKAGE_PIN M20   IOSTANDARD LVCMS33 } [get_ports { enable }]; #IO_L7N_T1_AD2N_35 Sch=SW0
12 #_set_property -dict { PACKAGE_PIN M19   IOSTANDARD LVCMS33 } [get_ports { sv[1] }]; #IO_L7P_T1_AD2P_35 Sch=SW1
13
14 ## RGB LEDs
15 #_set_property -dict { PACKAGE_PIN L15   IOSTANDARD LVCMS33 } [get_ports { led4_b }]; #IO_L22N_T3_AD7P_35 Sch=LED4_B
16 #_set_property -dict { PACKAGE_PIN G17   IOSTANDARD LVCMS33 } [get_ports { led4_g }]; #IO_L16P_T2_35 Sch=LED4_G
17 #_set_property -dict { PACKAGE_PIN N15   IOSTANDARD LVCMS33 } [get_ports { led6_r }]; #IO_L21P_T3_DOS_AD1P_35 Sch=LED4_R
18 #_set_property -dict { PACKAGE_PIN G14   IOSTANDARD LVCMS33 } [get_ports { led6_b }]; #IO_L0_35 Sch=LED5_B
19 #_set_property -dict { PACKAGE_PIN L14   IOSTANDARD LVCMS33 } [get_ports { led6_g }]; #IO_L22P_T3_AD7P_35 Sch=LED5_G
20 #_set_property -dict { PACKAGE_PIN M15   IOSTANDARD LVCMS33 } [get_ports { leds_r }]; #IO_L23N_T3_35 Sch=LED5_R
21
22 ## LEDs
23 #_set_property -dict { PACKAGE_PIN R14   IOSTANDARD LVCMS33 } [get_ports { out[0] }]; #IO_L6N_TO_VREF_34 Sch=LED0
24 #_set_property -dict { PACKAGE_PIN P14   IOSTANDARD LVCMS33 } [get_ports { out[1] }]; #IO_L6P_TO_34 Sch=LED1
25 #_set_property -dict { PACKAGE_PIN N16   IOSTANDARD LVCMS33 } [get_ports { out[2] }]; #IO_L21N_T3_DOS_AD1N_35 Sch=LED2
26 #_set_property -dict { PACKAGE_PIN M14   IOSTANDARD LVCMS33 } [get_ports { out[3] }]; #IO_L23P_T3_35 Sch=LED3
27
28 ## Buttons
29 #_set_property -dict { PACKAGE_PIN D19   IOSTANDARD LVCMS33 } [get_ports { bin_in[0] }]; #IO_L4P_T0_35 Sch=BTN0
30 #_set_property -dict { PACKAGE_PIN D20   IOSTANDARD LVCMS33 } [get_ports { bin_in[1] }]; #IO_L4N_T0_35 Sch=BTN1
31 #_set_property -dict { PACKAGE_PIN L20   IOSTANDARD LVCMS33 } [get_ports { bin_in[2] }]; #IO_L8N_T1_DOS_AD3N_35 Sch=BTN2
32 #_set_property -dict { PACKAGE_PIN L19   IOSTANDARD LVCMS33 } [get_ports { bin_in[3] }]; #IO_L9P_T1_DQS_AD3P_35 Sch=BTN3
33
34 ## Pmod Header JA
35
36
#_ChipKit Outer Digital Header
set_property -dict { PACKAGE_PIN T14   IOSTANDARD LVCMS33 } [get_ports { led_out[0] }]; #IO_L5P_TO_34 Sch=CK_I00
set_property -dict { PACKAGE_PIN U12   IOSTANDARD LVCMS33 } [get_ports { led_out[1] }]; #IO_LN_TO_34 Sch=CK_I01
set_property -dict { PACKAGE_PIN U13   IOSTANDARD LVCMS33 } [get_ports { led_out[2] }]; #IO_L3P_TO_DOS_PUDC_B_34 Sch=CK_I02
set_property -dict { PACKAGE_PIN V13   IOSTANDARD LVCMS33 } [get_ports { led_out[3] }]; #IO_L8N_TO_DQS_34 Sch=CK_I03
set_property -dict { PACKAGE_PIN V15   IOSTANDARD LVCMS33 } [get_ports { led_out[4] }]; #IO_L10P_TO_34 Sch=CK_I04
set_property -dict { PACKAGE_PIN T15   IOSTANDARD LVCMS33 } [get_ports { led_out[5] }]; #IO_L5P_TO_34 Sch=CK_I05
set_property -dict { PACKAGE_PIN R16   IOSTANDARD LVCMS33 } [get_ports { led_out[6] }]; #IO_L9P_T1_TO_34 Sch=CK_I06
set_property -dict { PACKAGE_PIN U17   IOSTANDARD LVCMS33 } [get_ports { led_out[7] }]; #IO_L8N_T1_DQS_34 Sch=CK_I07
#set_property -dict { PACKAGE_PIN V17   IOSTANDARD LVCMS33 } [get_ports { ck_i08 }]; #IO_L21P_T3_DQS_34 Sch=CK_I08
#set_property -dict { PACKAGE_PIN V18   IOSTANDARD LVCMS33 } [get_ports { ck_i09 }]; #IO_L21N_T3_DQS_34 Sch=CK_I09
set_property -dict { PACKAGE_PIN T16   IOSTANDARD LVCMS33 } [get_ports { ck_i10 }]; #IO_L8P_TO_DQS_34 Sch=CK_I10
#set_property -dict { PACKAGE_PIN R17   IOSTANDARD LVCMS33 } [get_ports { ck_i11 }]; #IO_L19N_T3_VREF_34 Sch=CK_I011
#set_property -dict { PACKAGE_PIN P18   IOSTANDARD LVCMS33 } [get_ports { ck_i12 }]; #IO_L23N_T3_34 Sch=CK_I012
#set_property -dict { PACKAGE_PIN N17   IOSTANDARD LVCMS33 } [get_ports { ck_i13 }]; #IO_L23P_T3_34 Sch=CK_I013

```

Figure 9: Assign pin for Arty-Z27 before running for FPGA

Then we have the generated bitstream file is **bin2led7.bit**

And we have some pictures about led in board as follow:

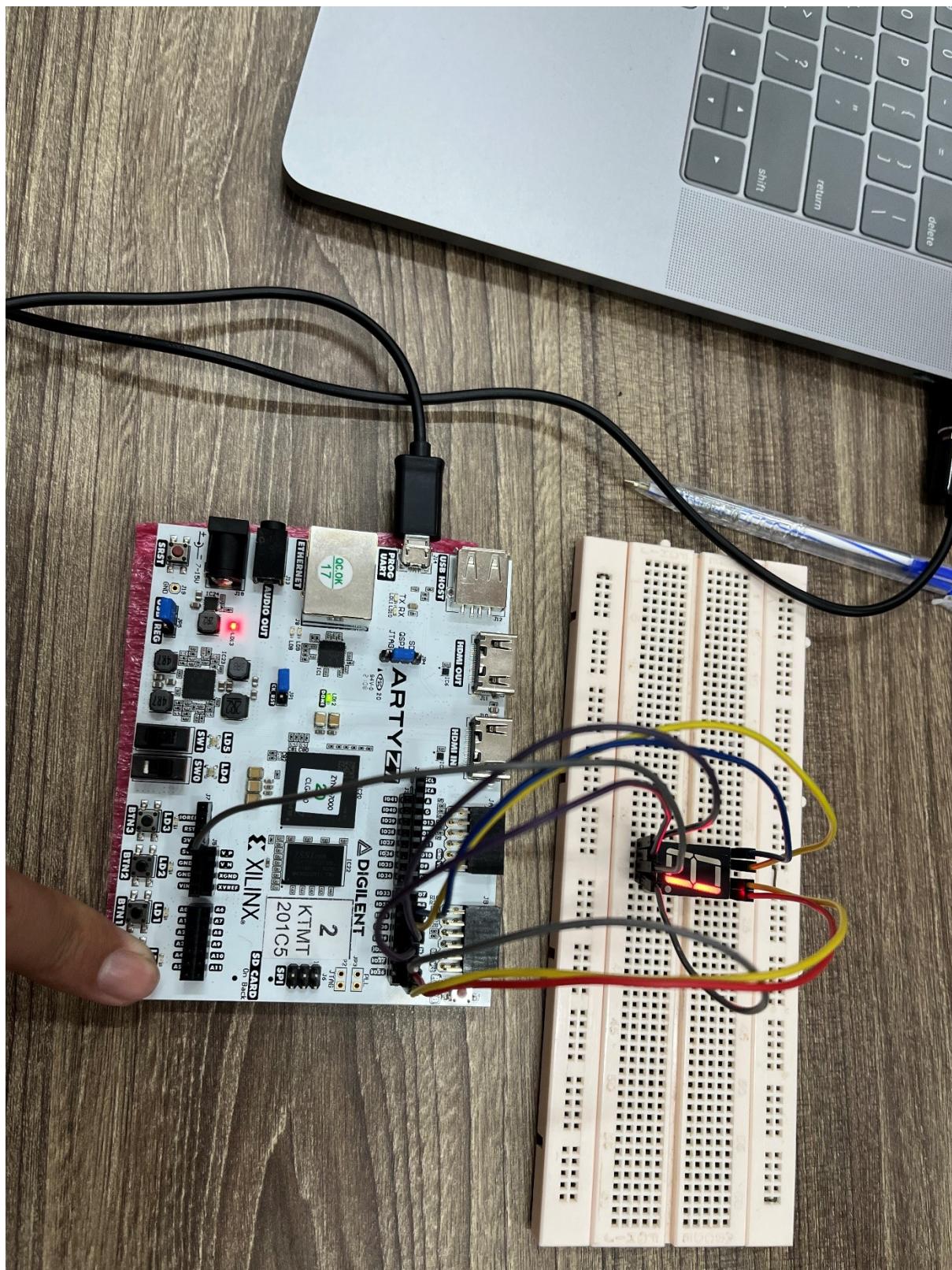


Figure 10: Ouput is 1

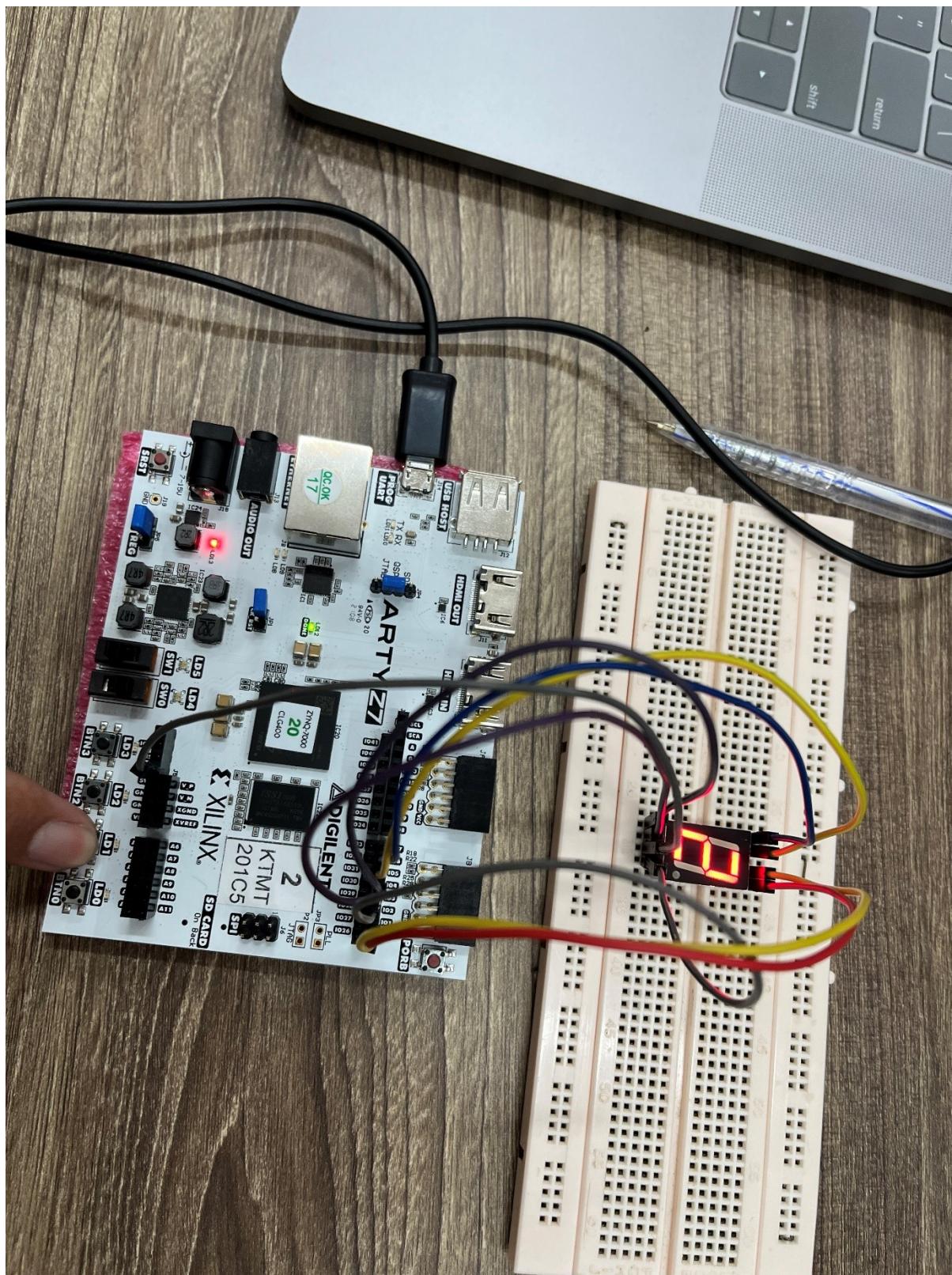


Figure 11: Ouput is 2

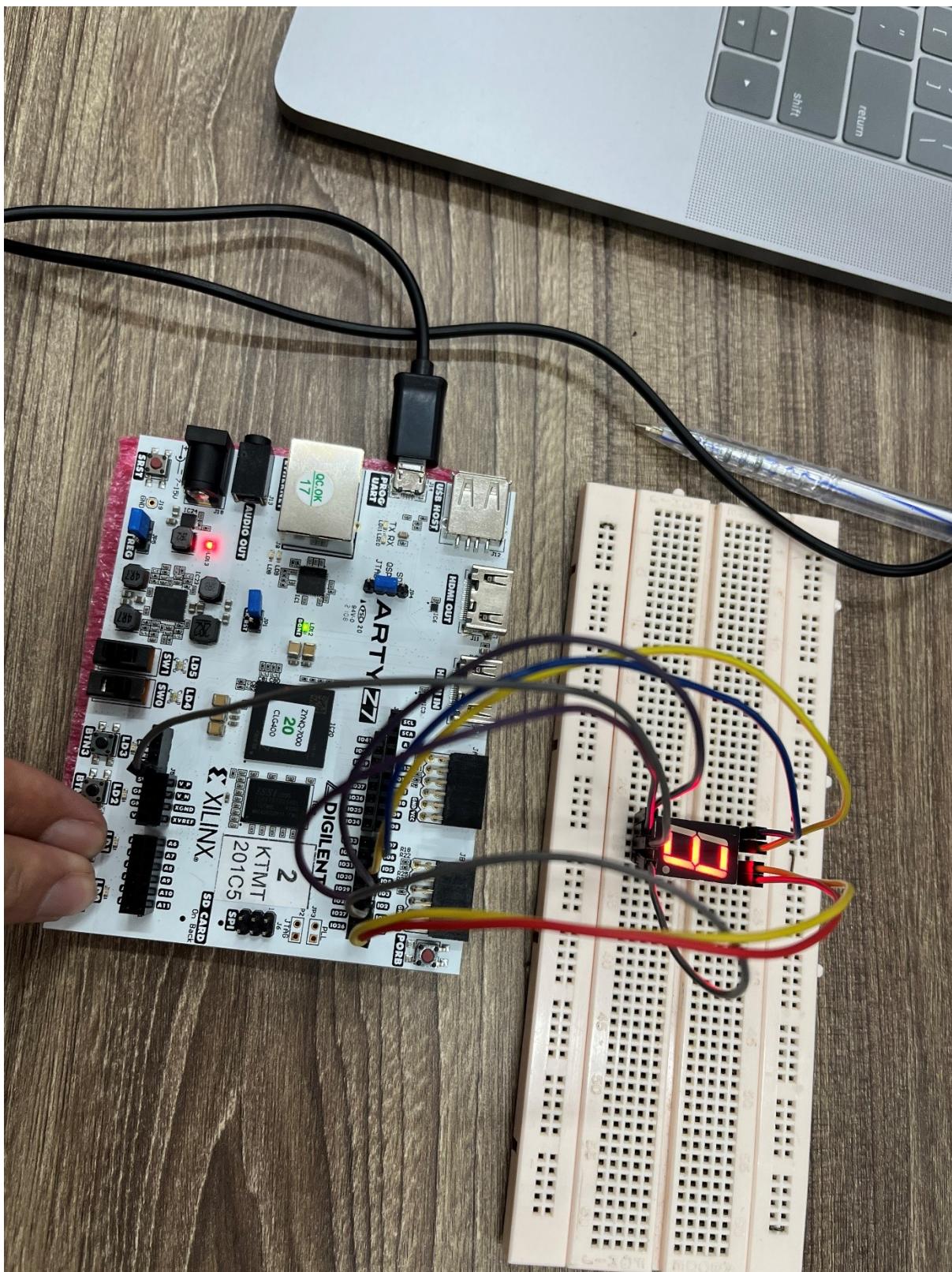


Figure 12: Output is 3

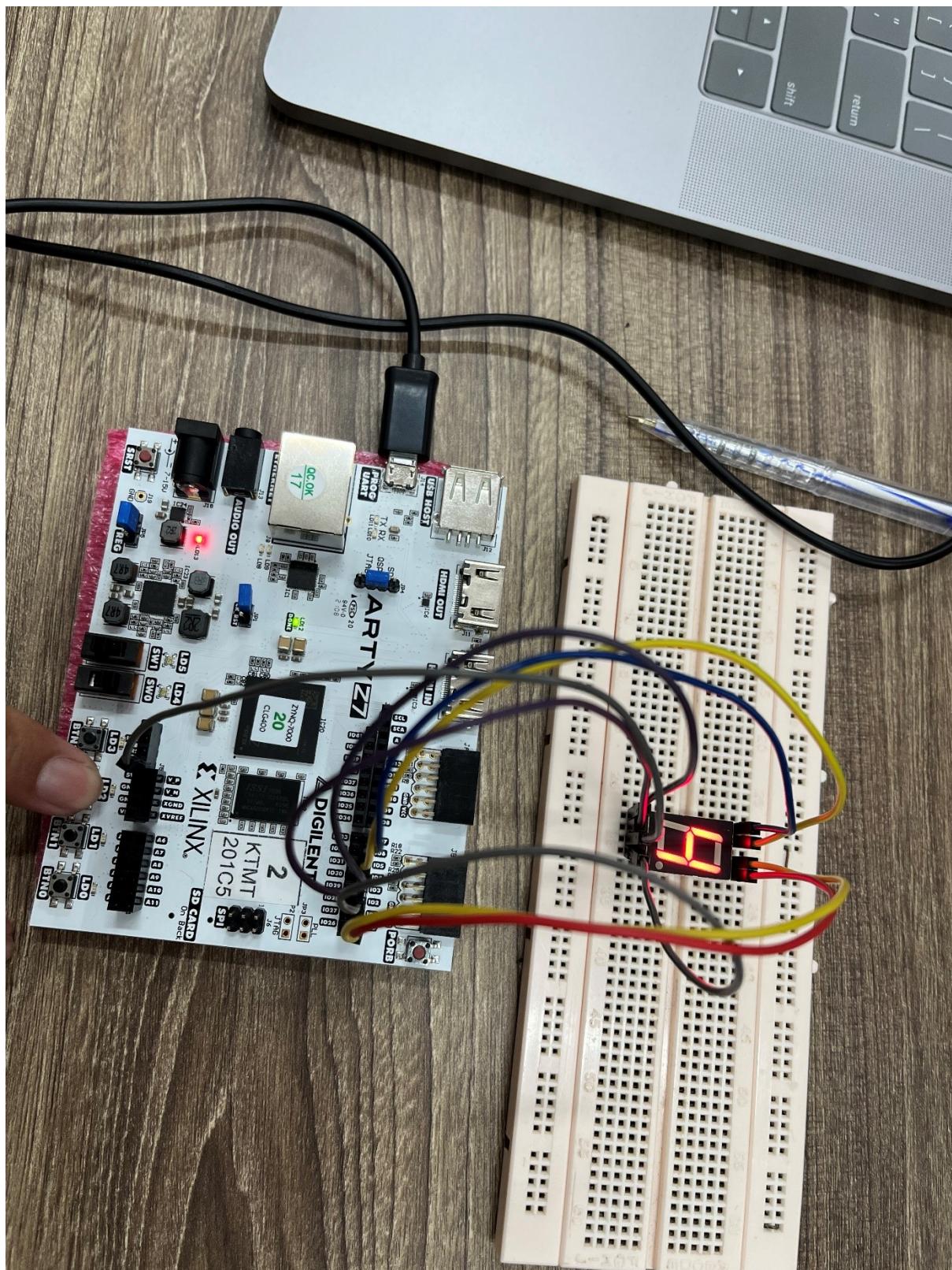


Figure 13: Ouput is 4

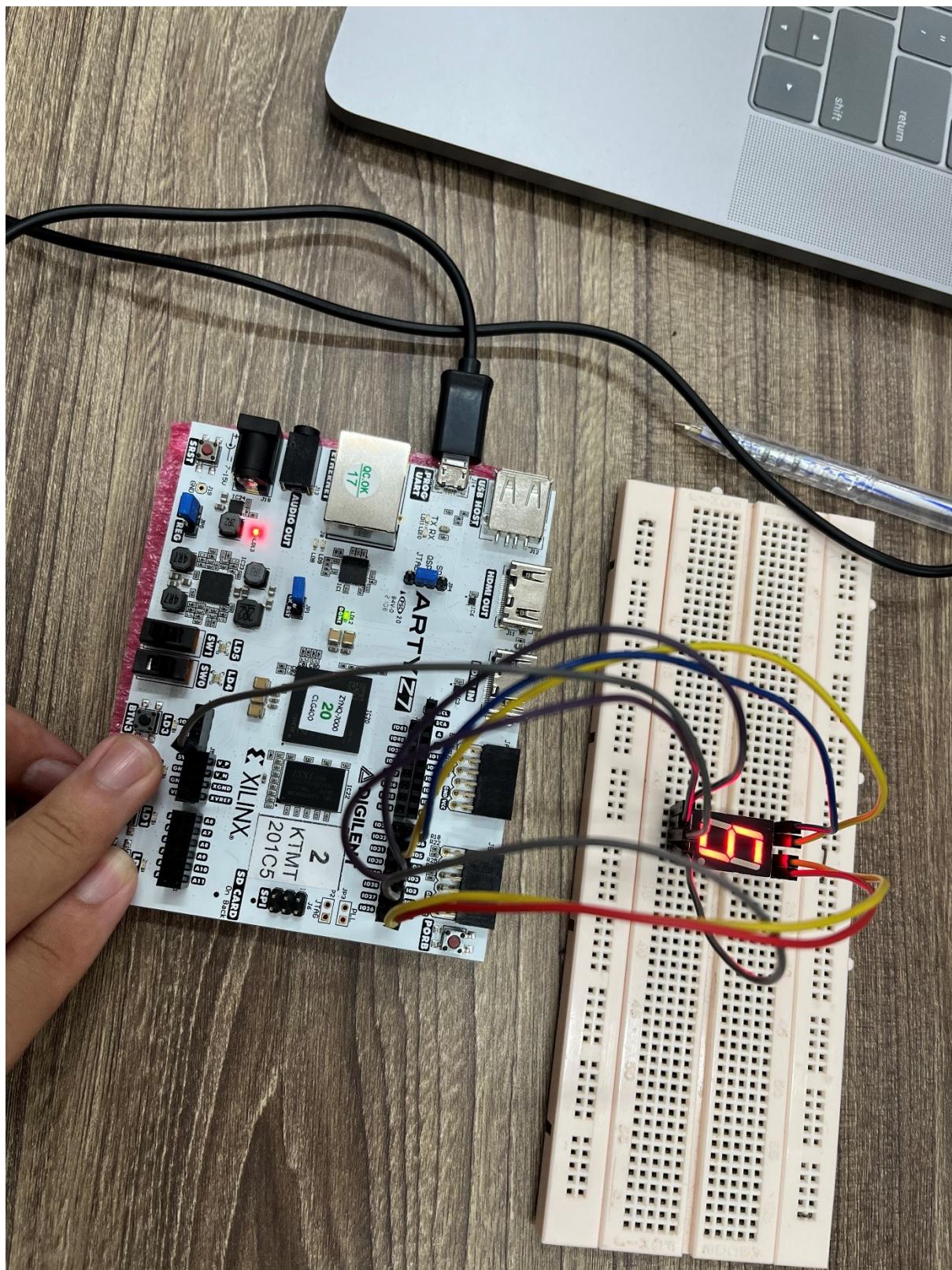


Figure 14: Output is 5

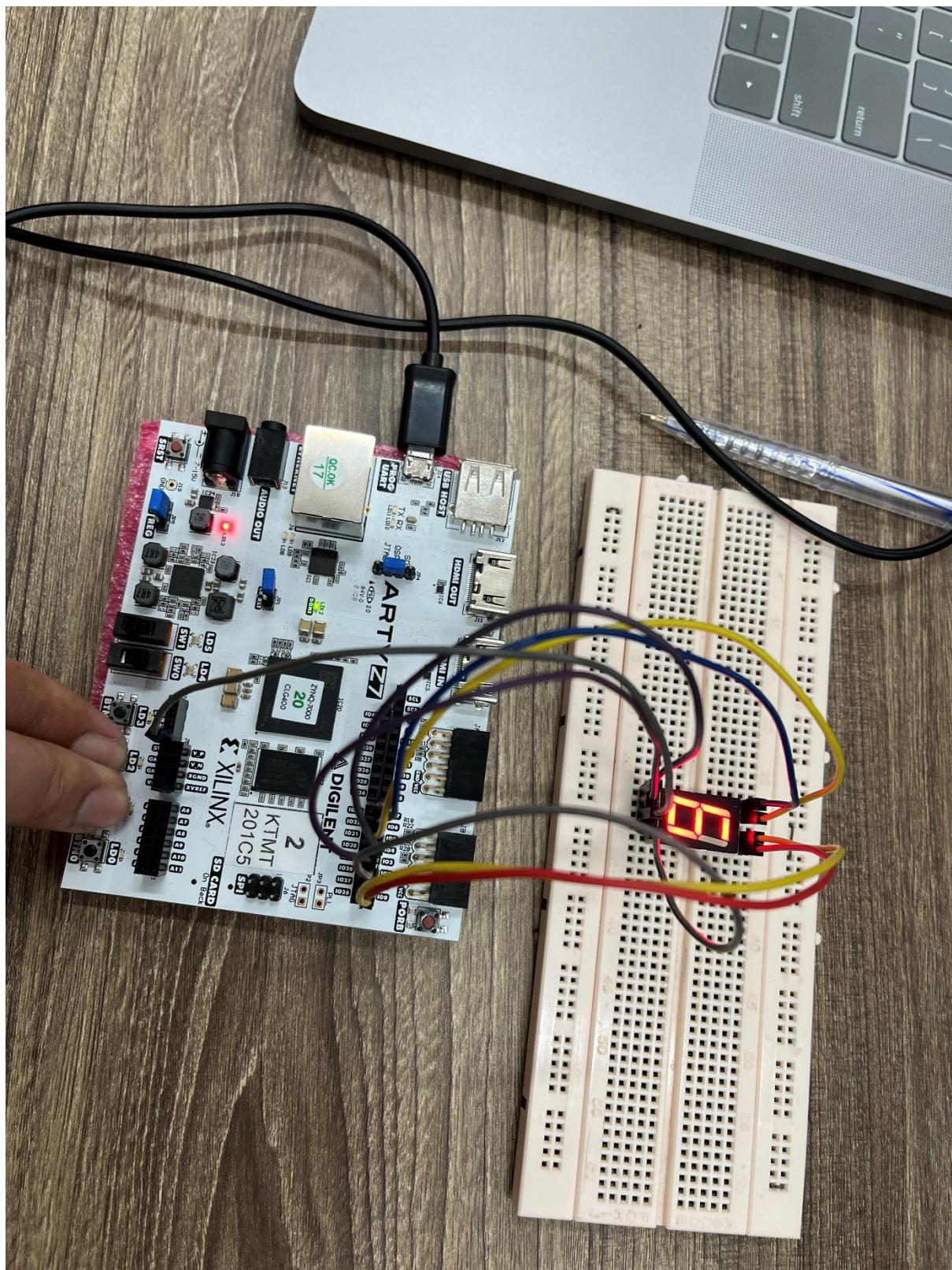


Figure 15: Ouput is 6

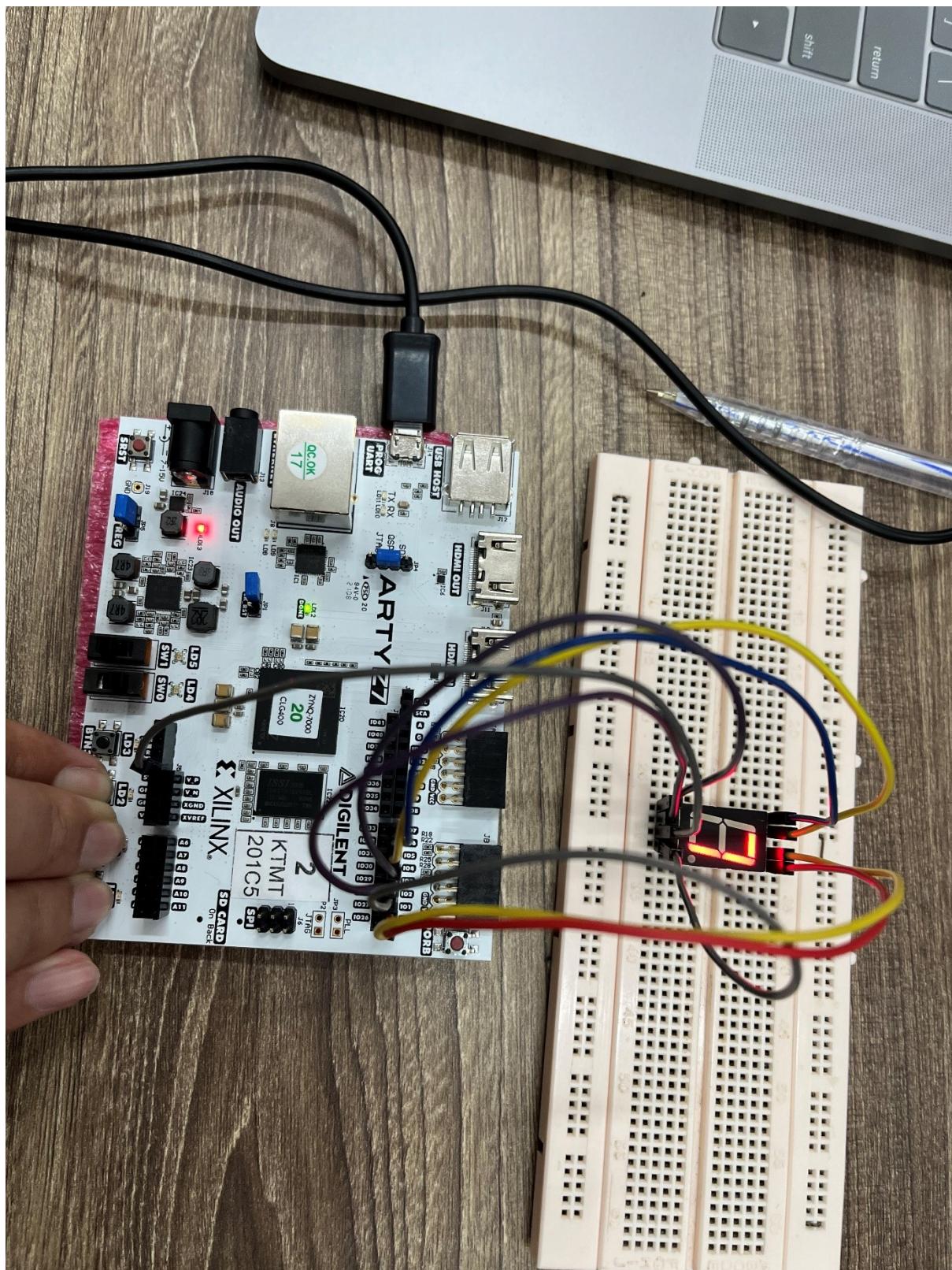


Figure 16: Ouput is 7

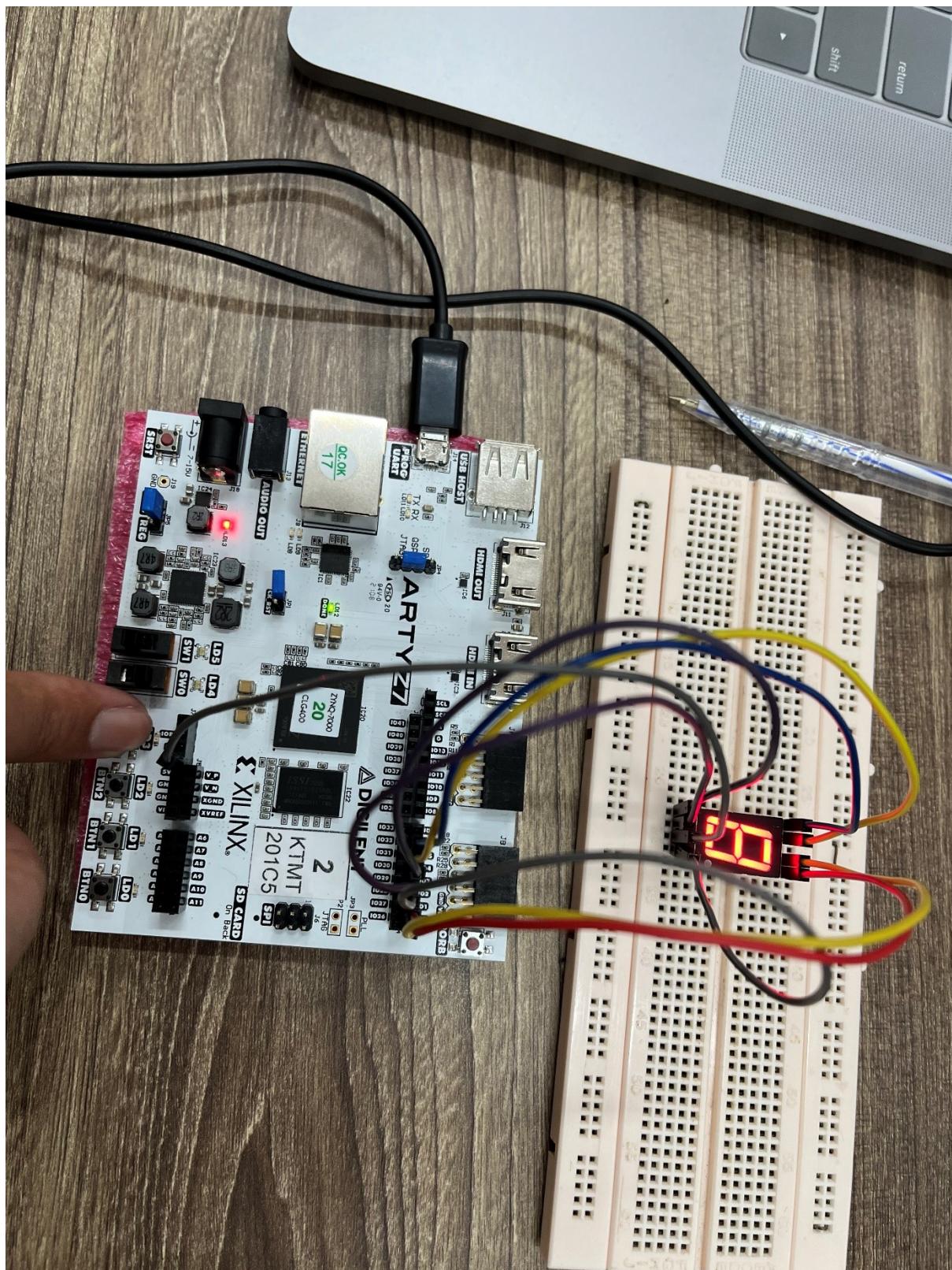


Figure 18: Ouput is 9

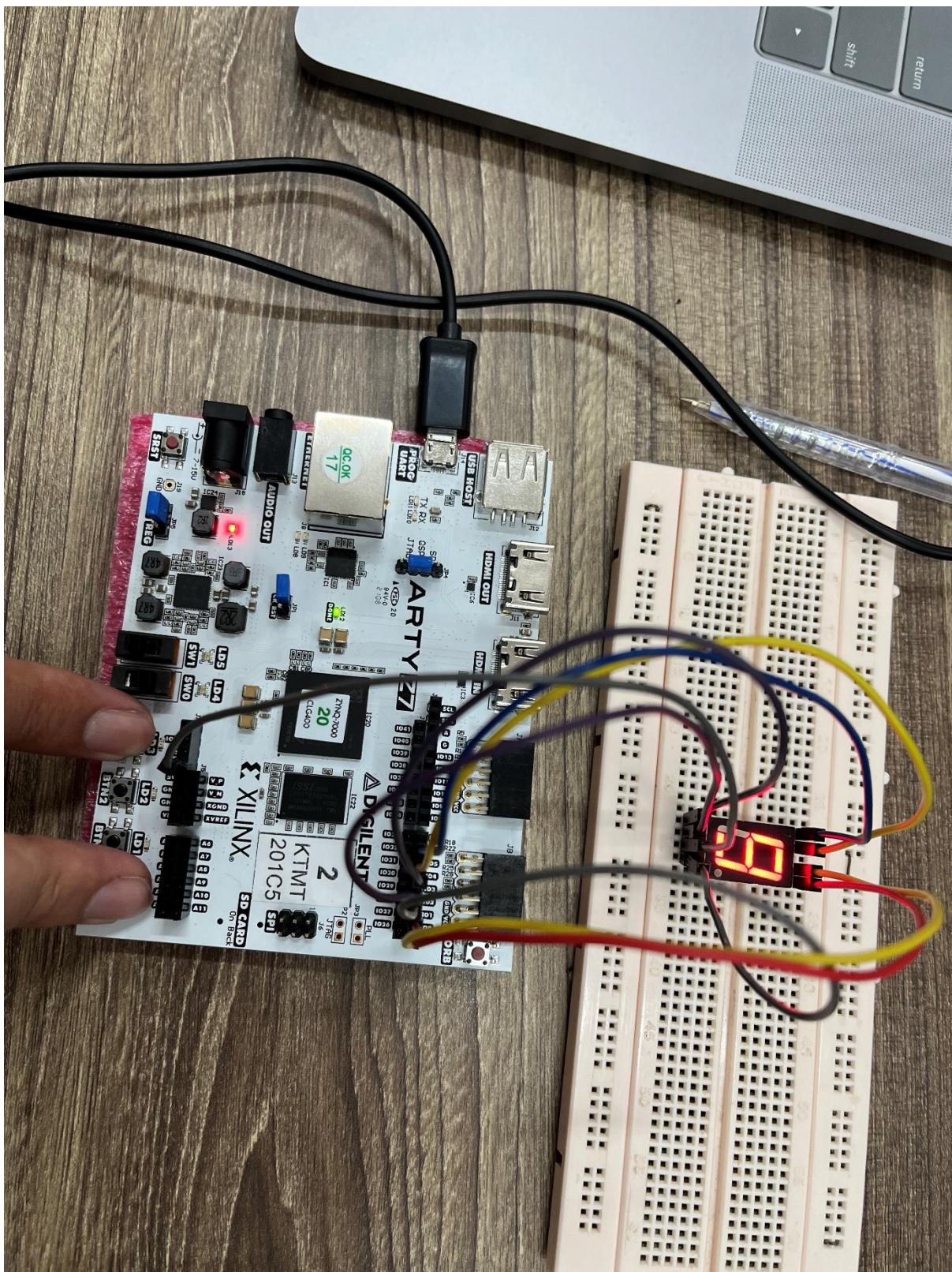


Figure 18: Output is 9

Exercise 3

Design a circuit to control the RGB LEDs on Arty-Z7 board as follow:

Design a circuit to control the RGB LEDs on Arty-Z7 board as follow:

We have file code is **rgb_control.v**

And then we have RTL schematic circuit:

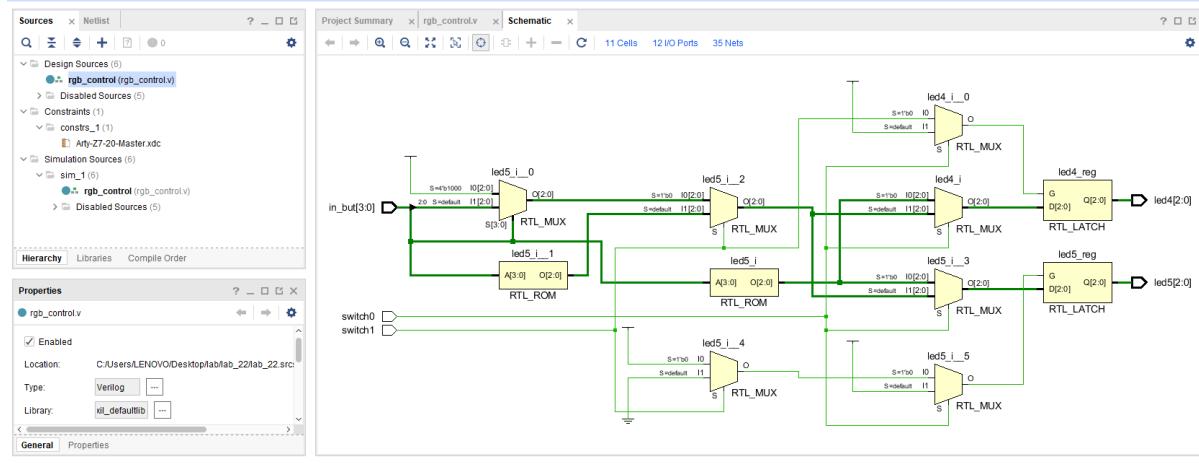


Figure 19: RGB LED in RTL schematic

We have file code for test bench of RGB is **rgb_control_tb.v**

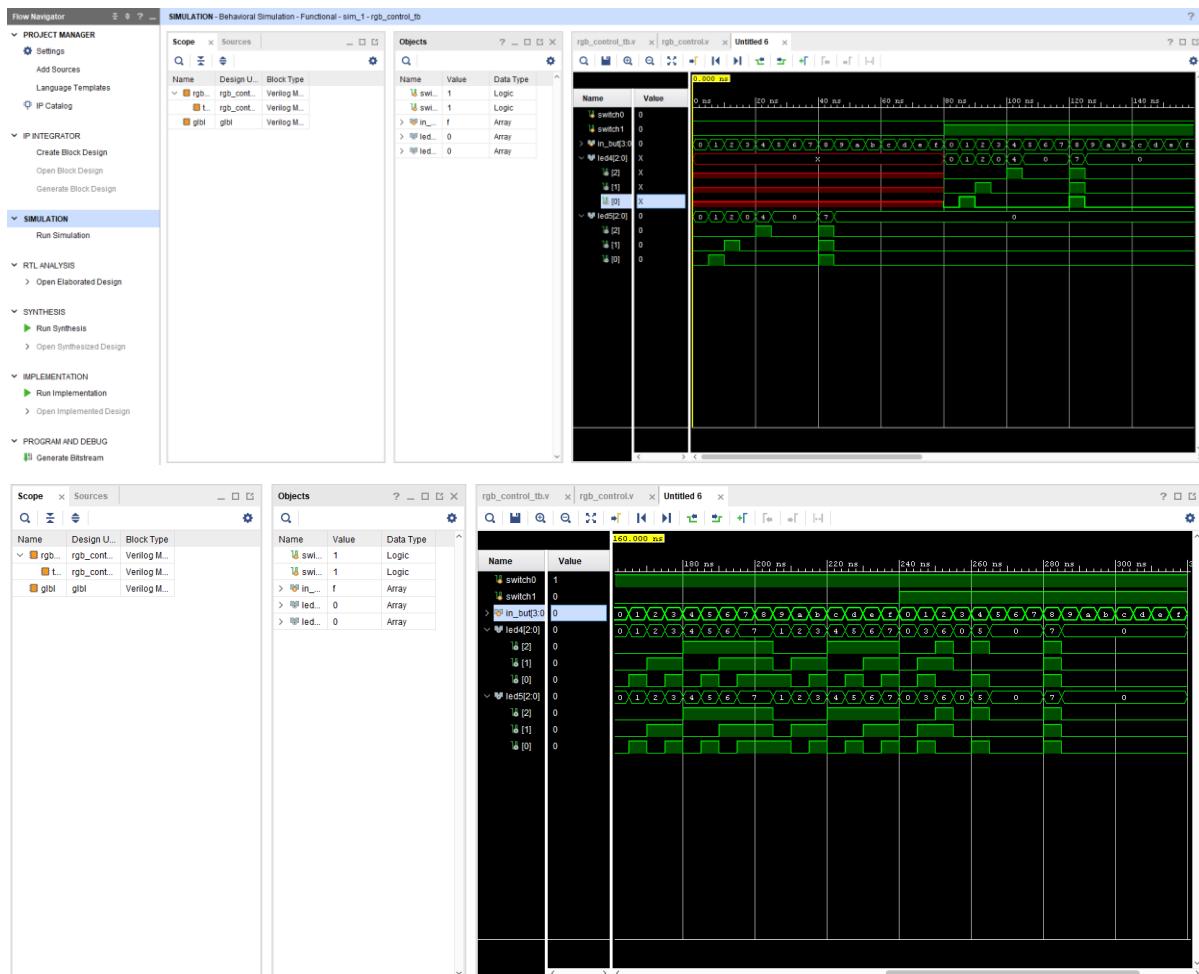


Figure 20: The waveform of test bench for RGB LED

Test the circuit on FPGA using switches, buttons and RGB LEDs on the Arty-Z7 board.

We have some changes in constraint as follow:

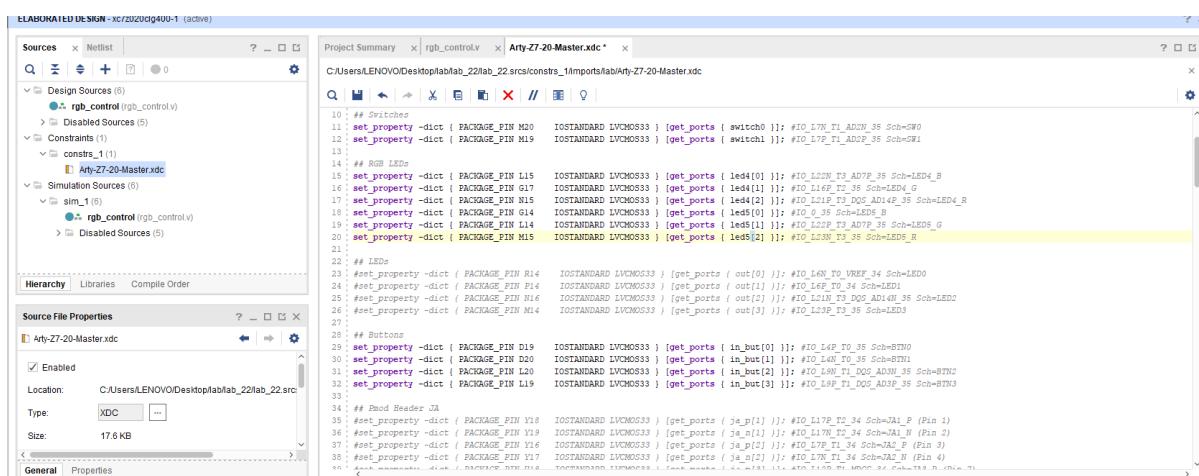


Figure 21: Assign pin for Arty-Z27 before running for FPGA

Then, the generated bitstream file is **rgb_control.bit**

And we have some pictures about led in board as follow:

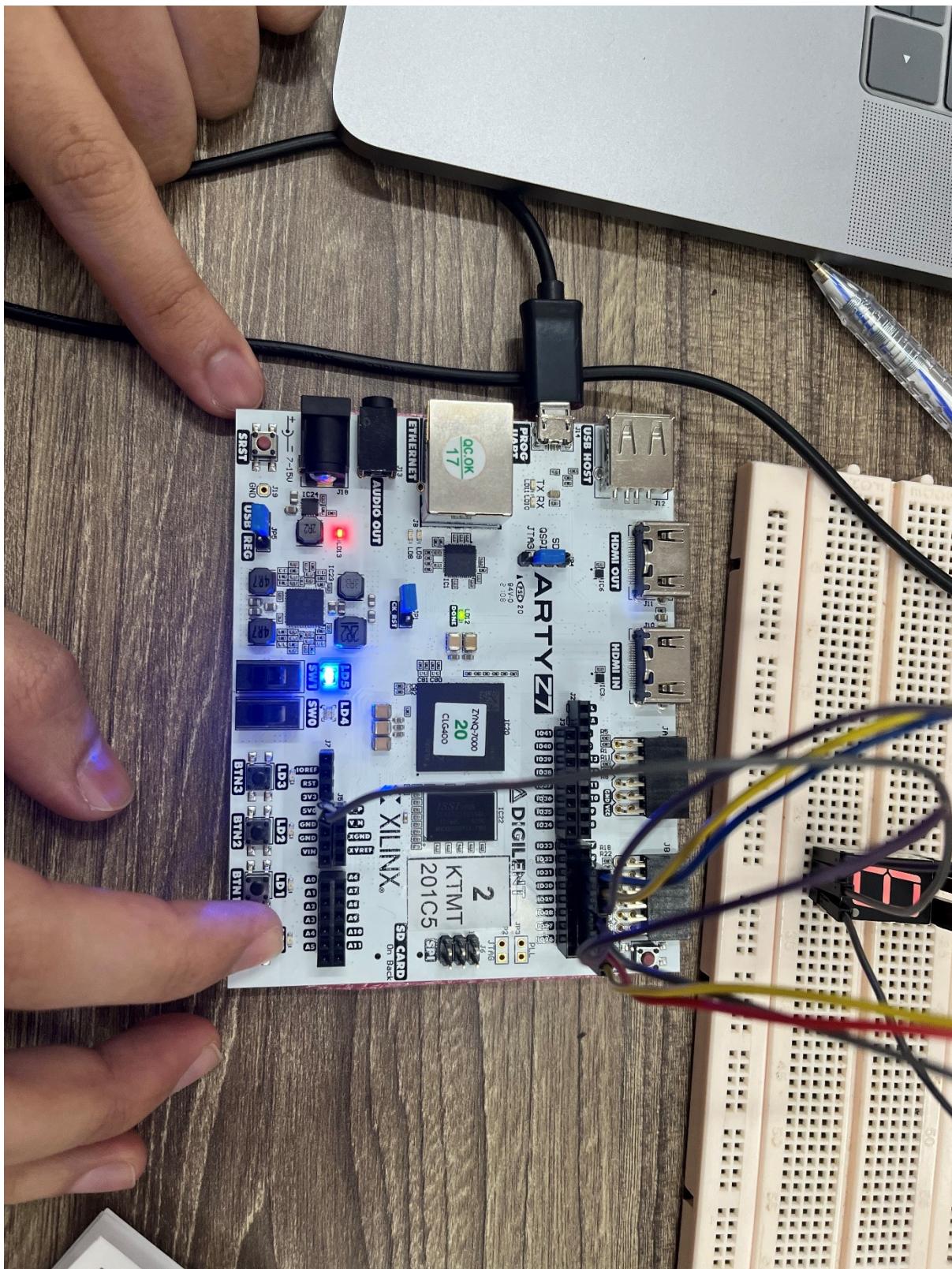


Figure 22: LED BLUE when switch0 is 0 and switch1 is 0 and button is 0001

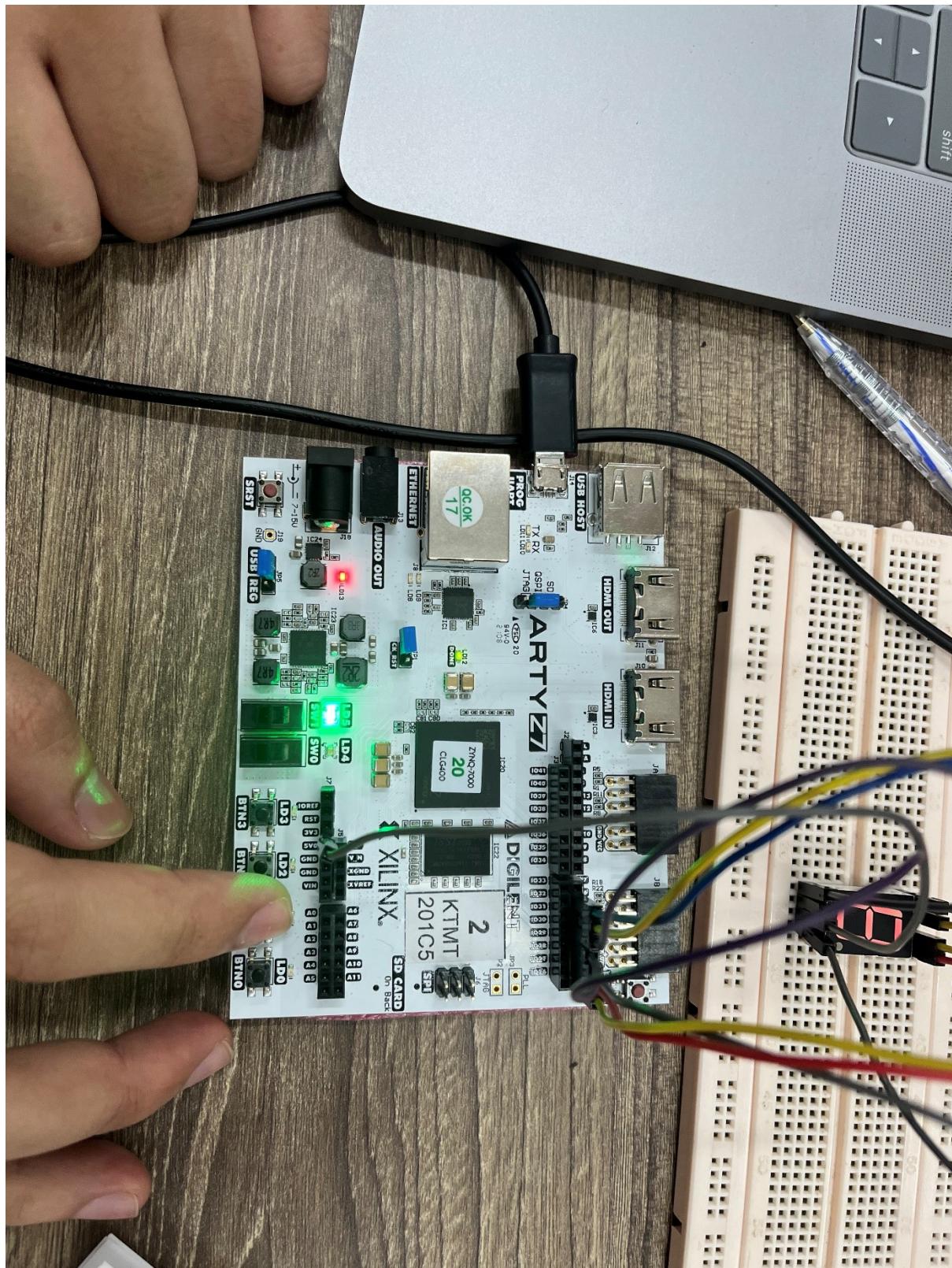


Figure 23: LED GREEN when switch0 is 0 switch1 is 0 and button is 0010

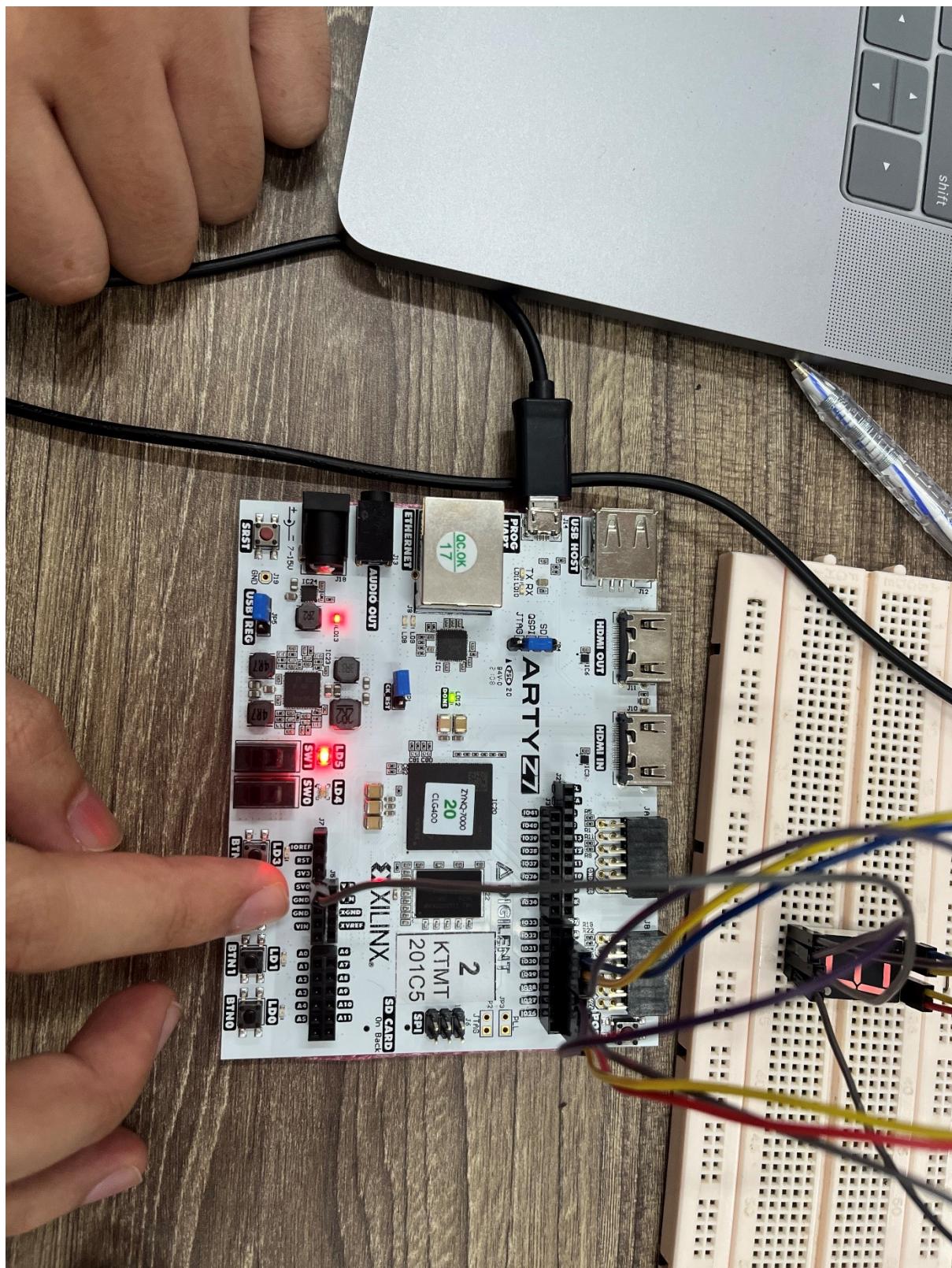


Figure 24: LED RED when switch0 is 0 switch1 is 0 and button is 0100

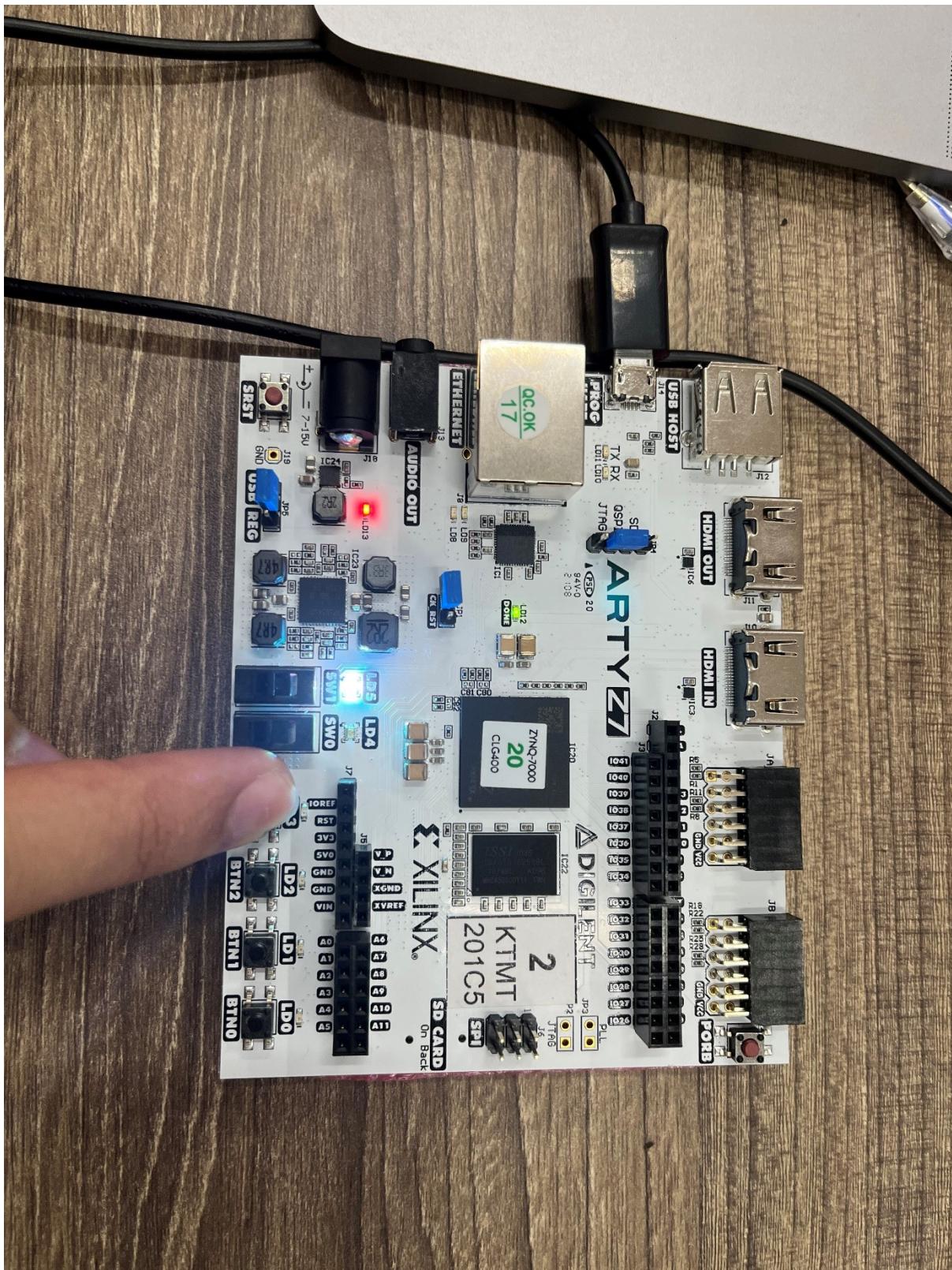


Figure 25: LED WHITE when switch0 is 0 and switch1 is 0 and button is 1000

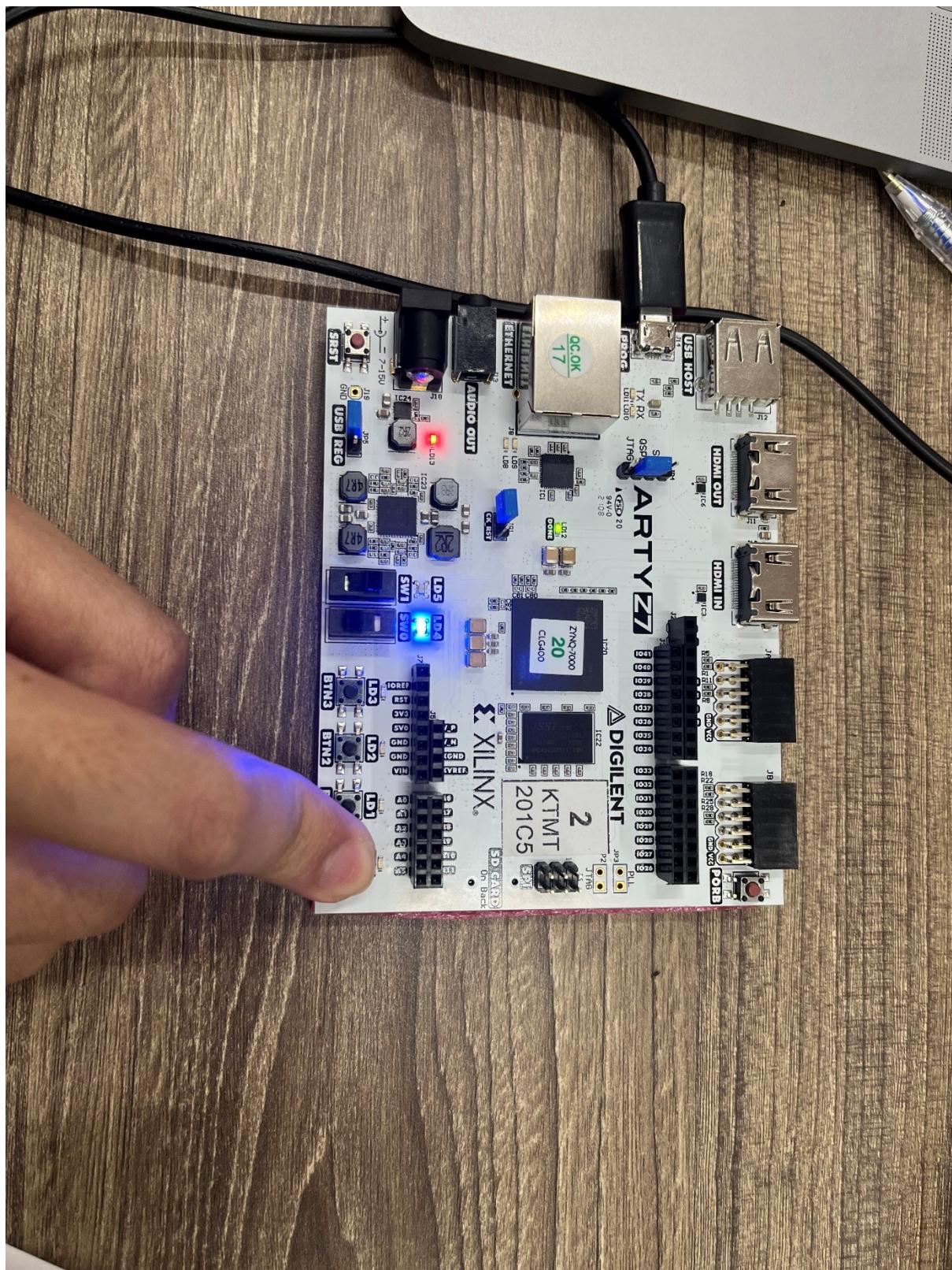


Figure 26: LED BLUE when switch0 is 0 switch1 is 1 and button is 0001

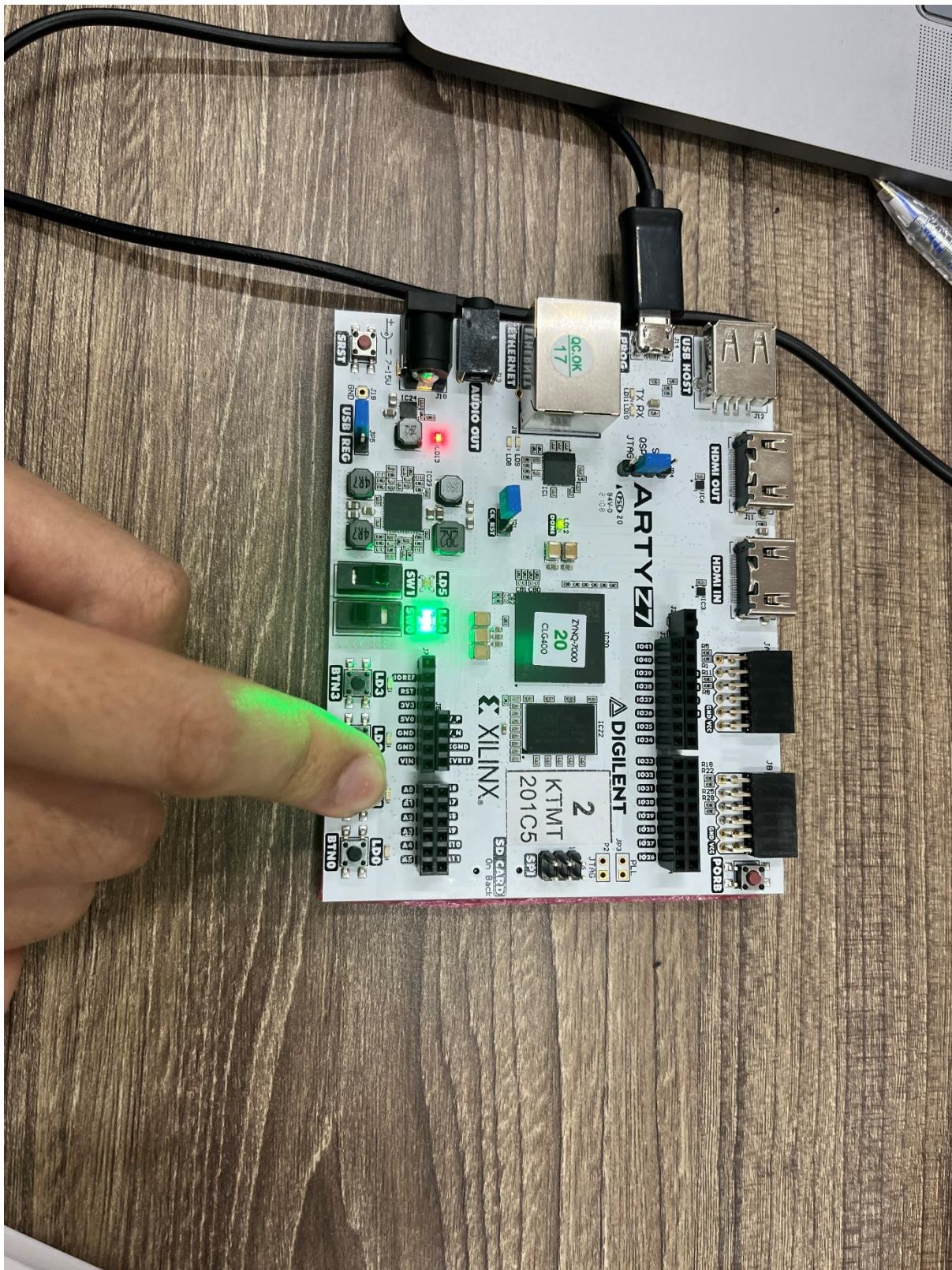


Figure 27: LED GREEN when switch0 is 0 and switch1 is 1 and button is 0010

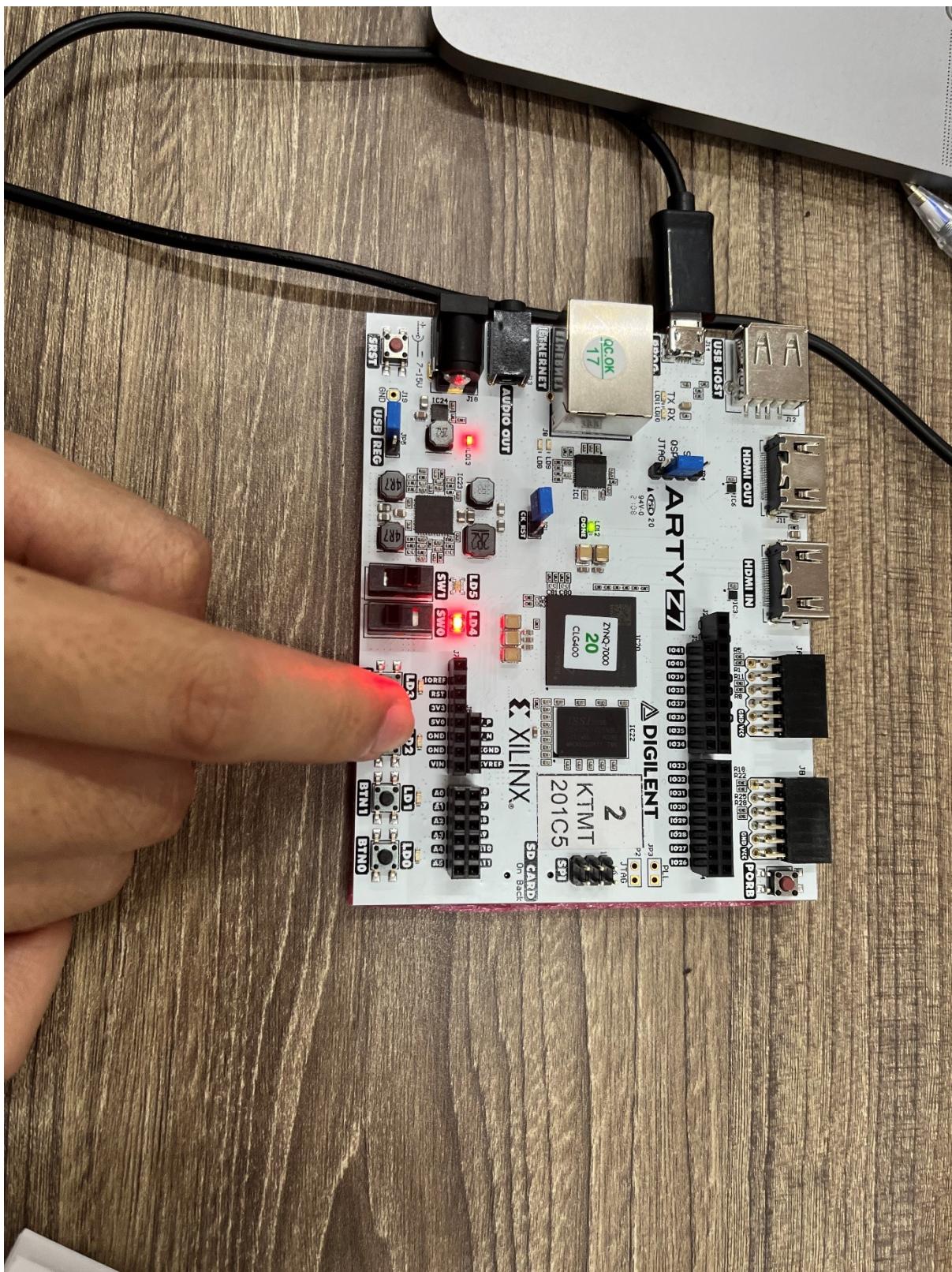


Figure 28: LED RED when switch0 is 0 switch1 is 1 and button is 0100

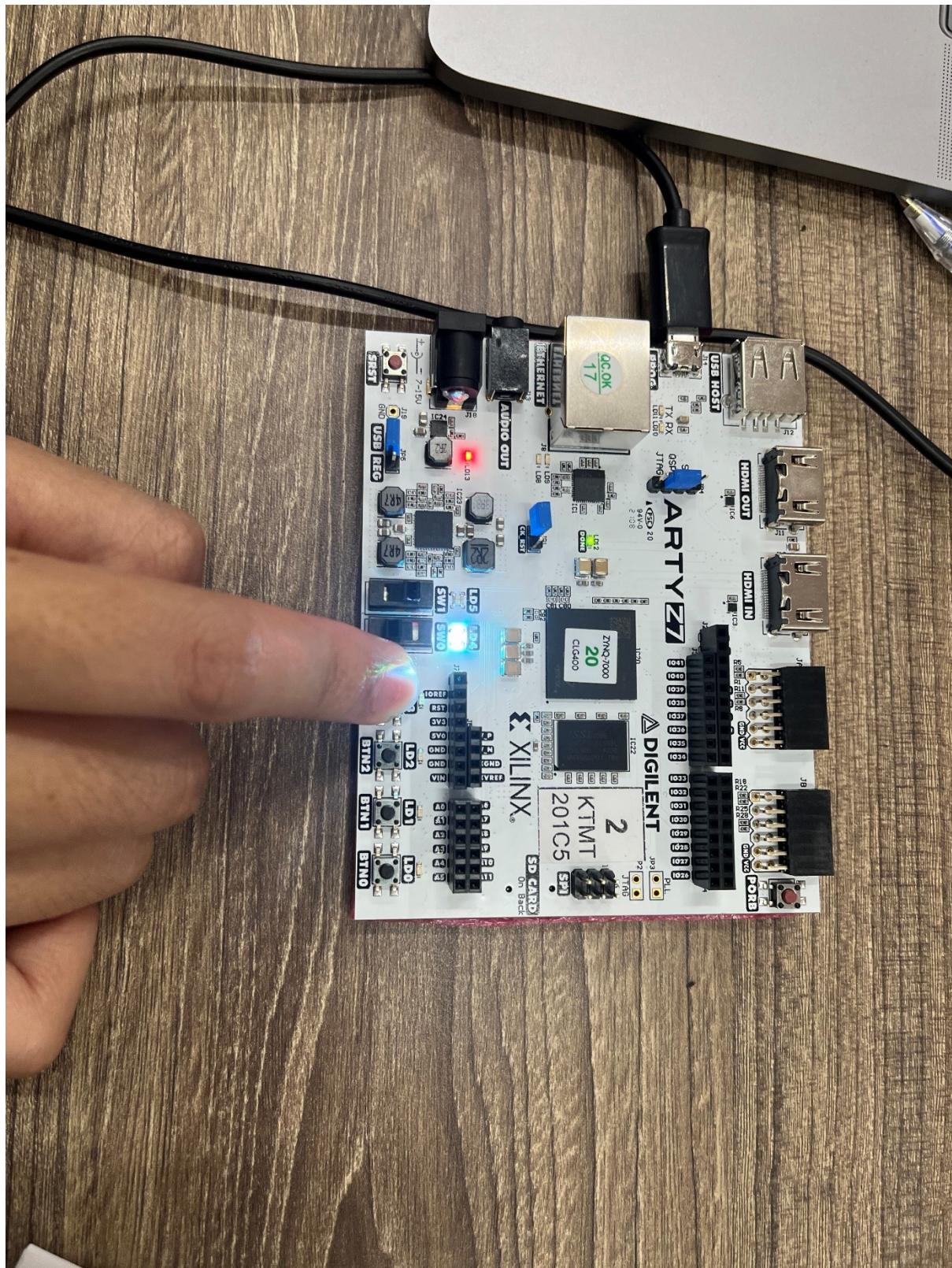


Figure 29: LED WHITE when switch0 is 0 switch1 is 1 and button is 1000

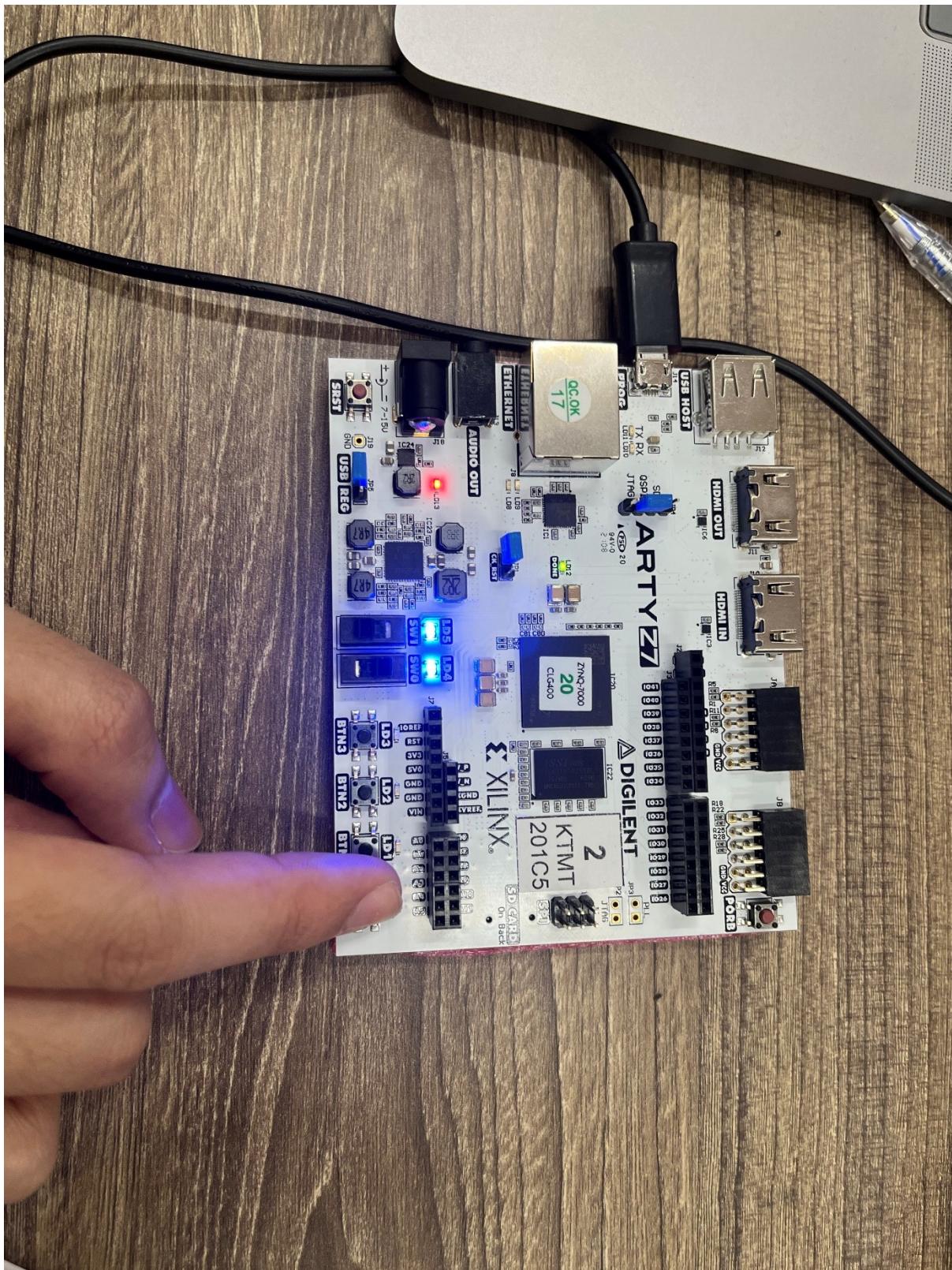


Figure 30: LED BLUE when switch0 is 1 switch1 is 0 and button is 0001

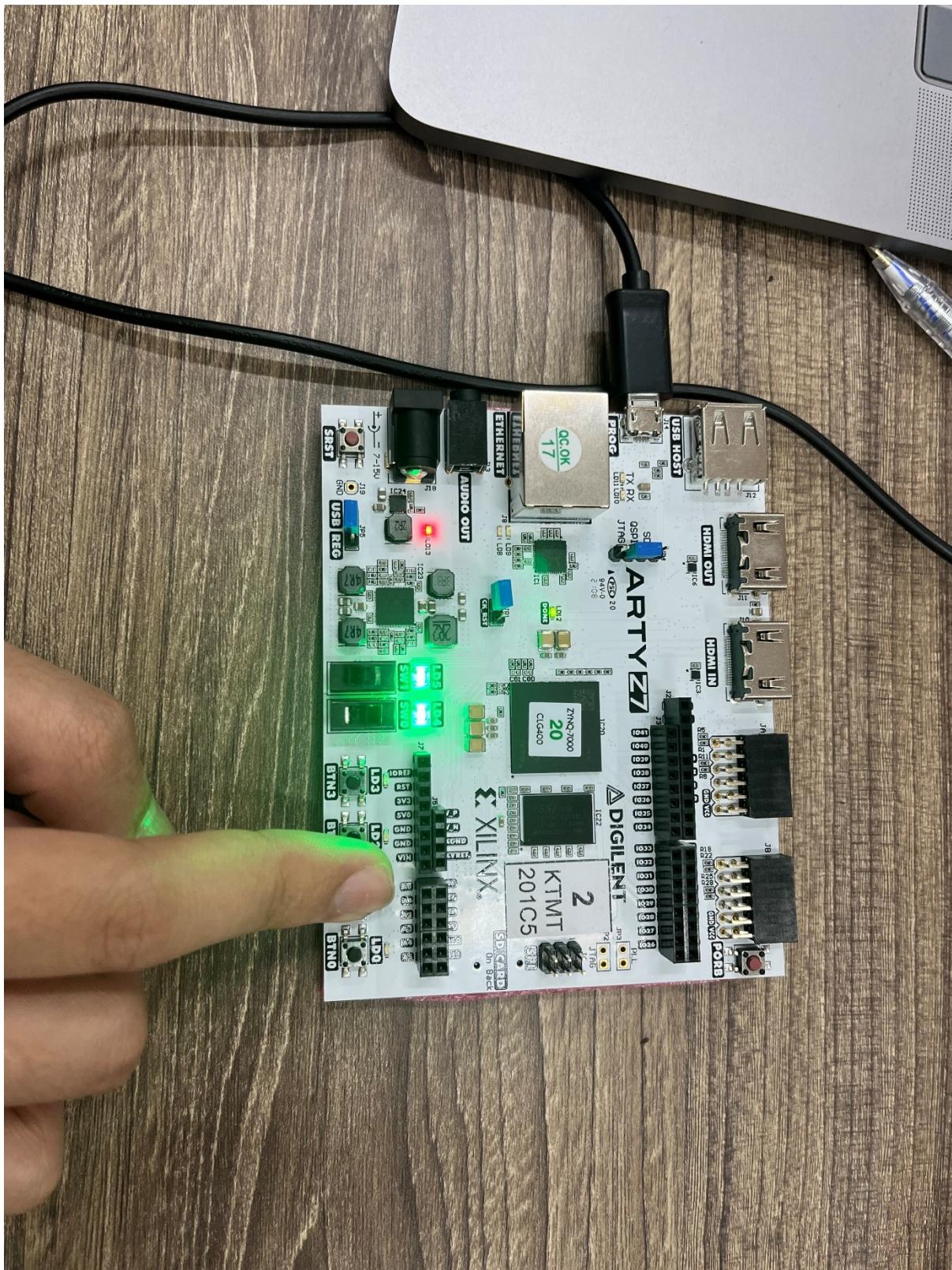


Figure 31: LED GREEN when switch0 is 1 switch1 is 0 and button is 0010

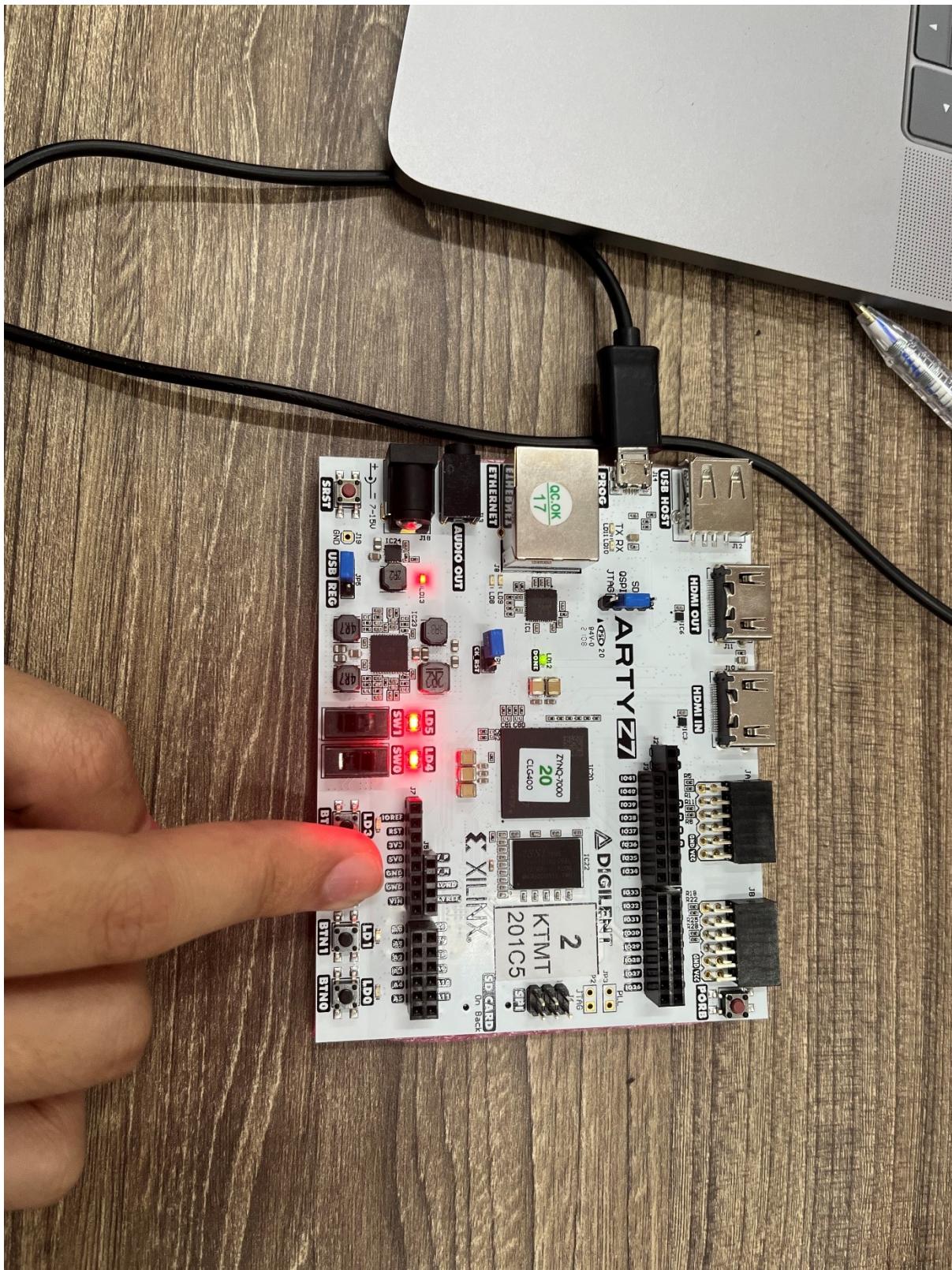


Figure 32: LED RED when switch0 is 1 switch1 is 0 and button is 0100

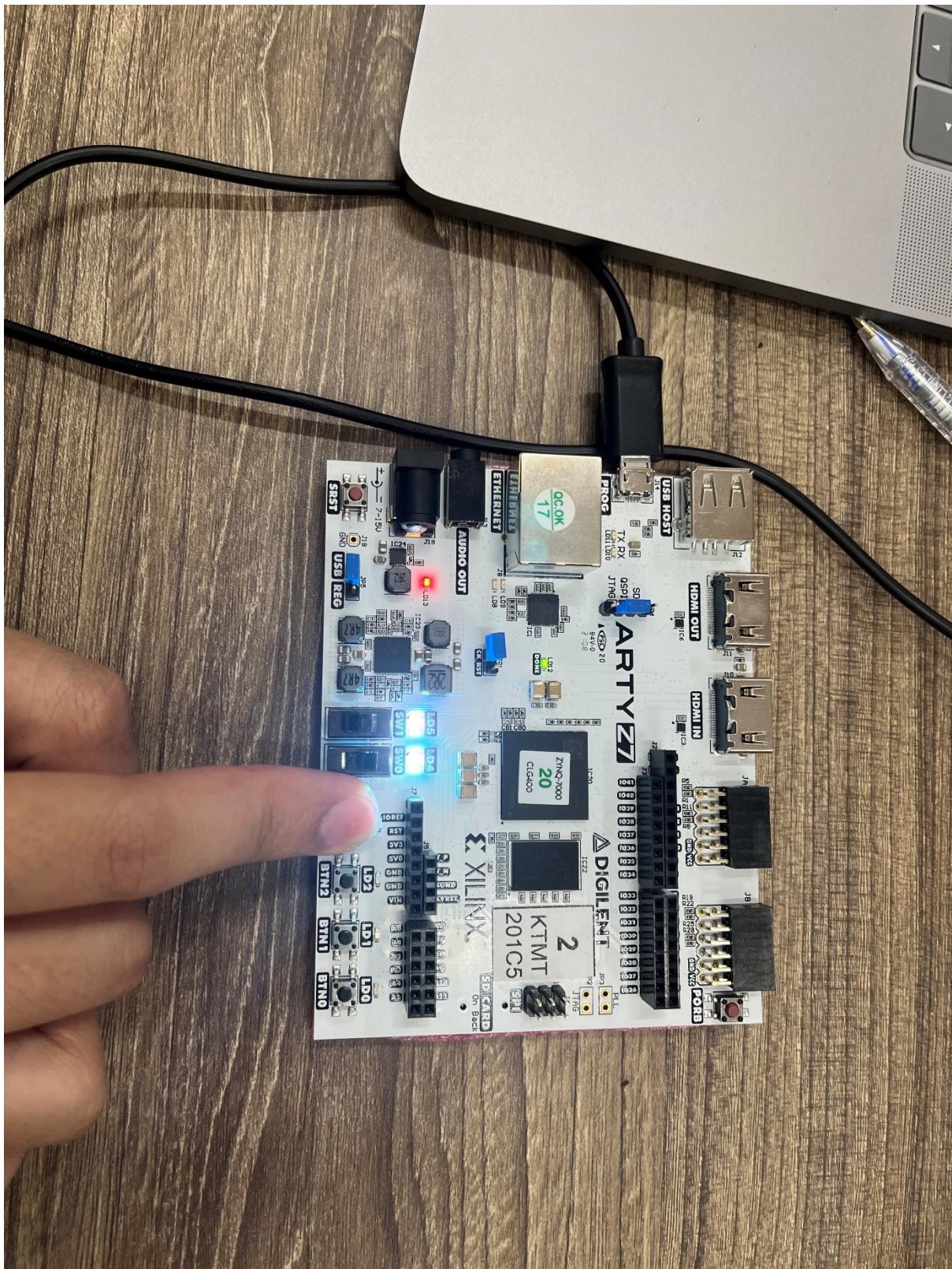


Figure 33: LED WHITE when switch0 is 1 switch1 is 0 and button is 1000

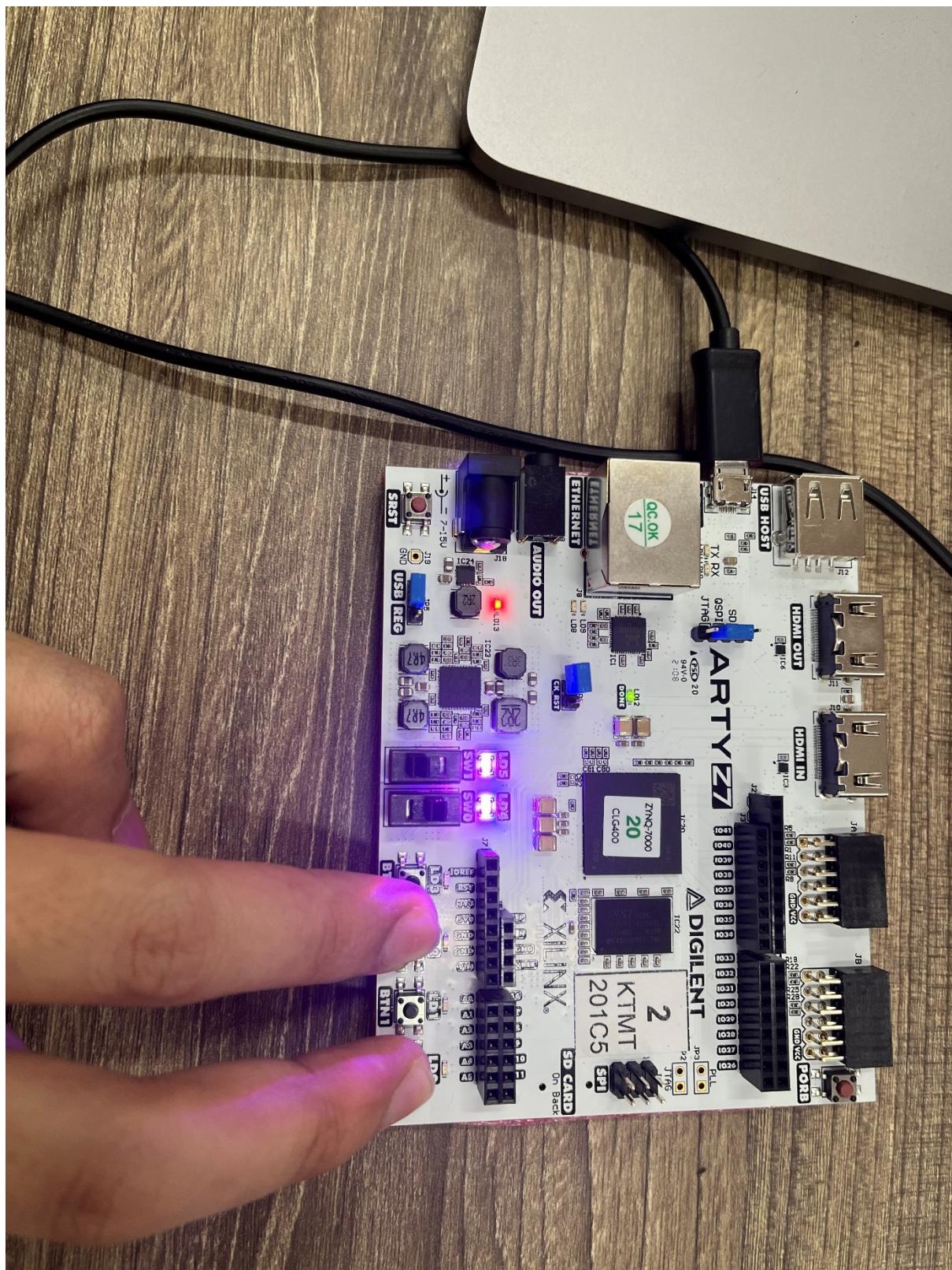


Figure 34: LED MIX when switch0 is 1 switch1 is 0 and button is 0101

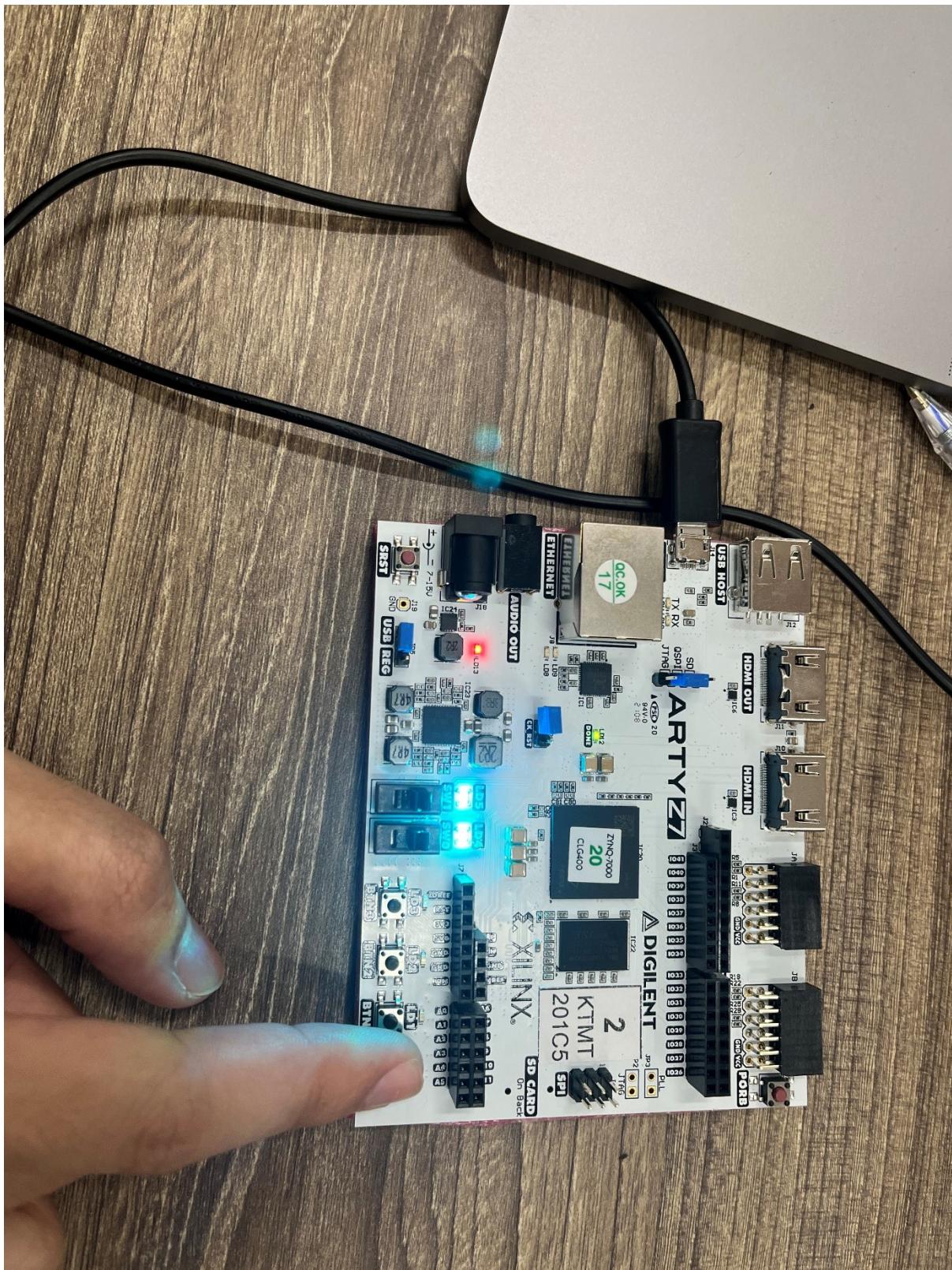


Figure 35: LED CYAN when switch1 is 1 switch1 is 1 and button is 0001

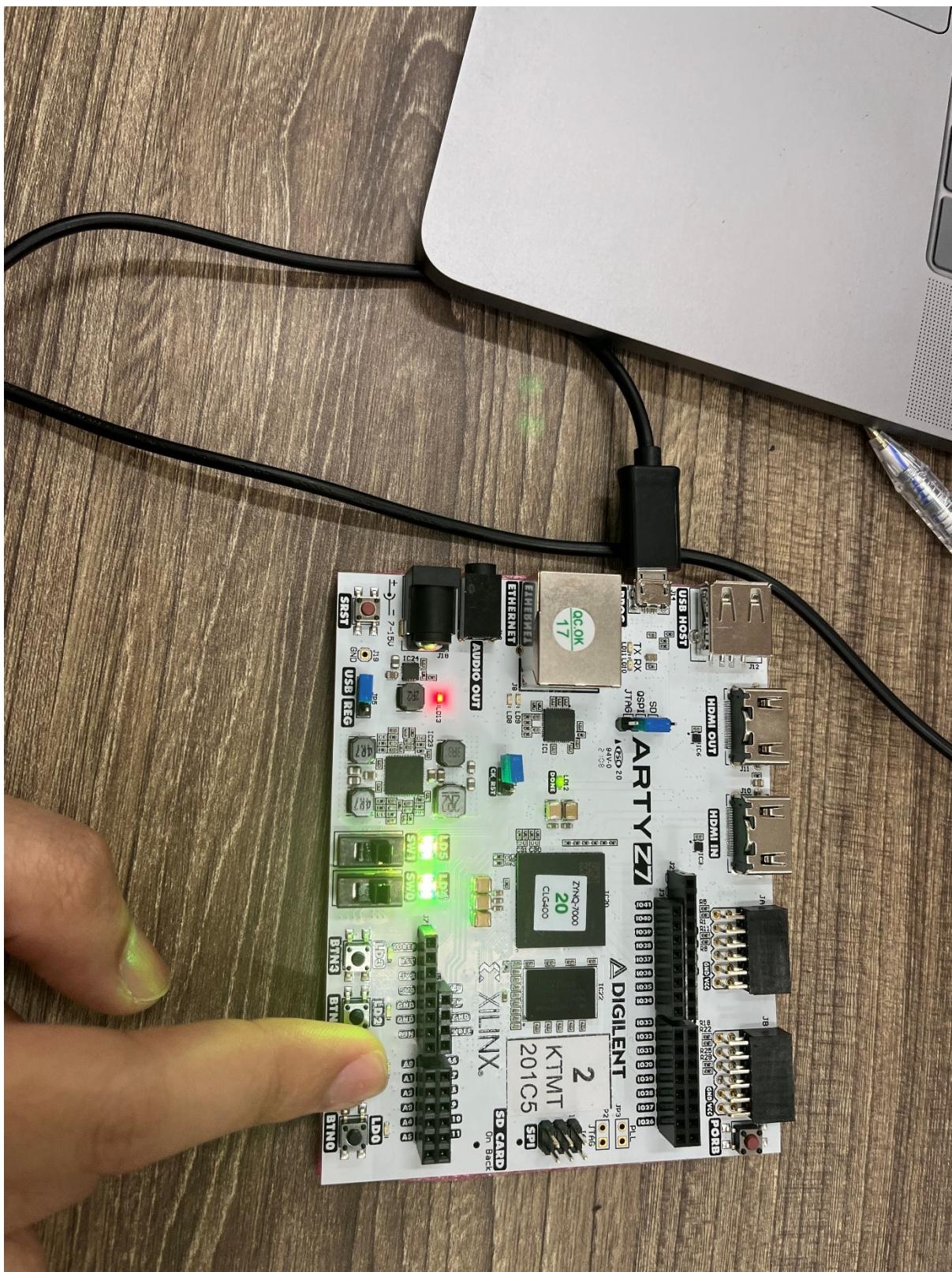


Figure 36: LED YELLOW when switch1 is 1 switch1 is 1 and button is 0010

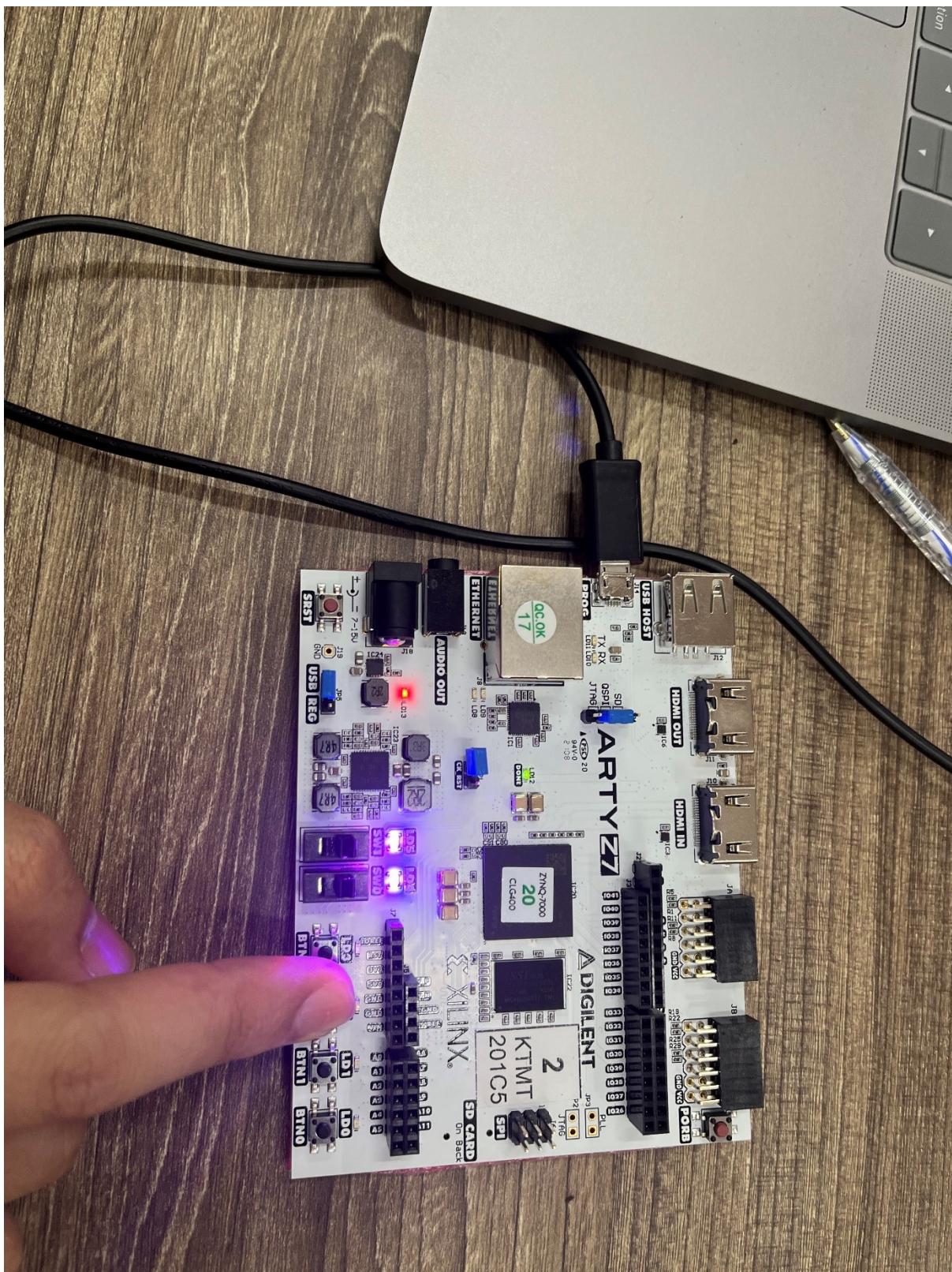


Figure 37: LED PURPLE when switch1 is 1 switch1 is 1 and button is 0100

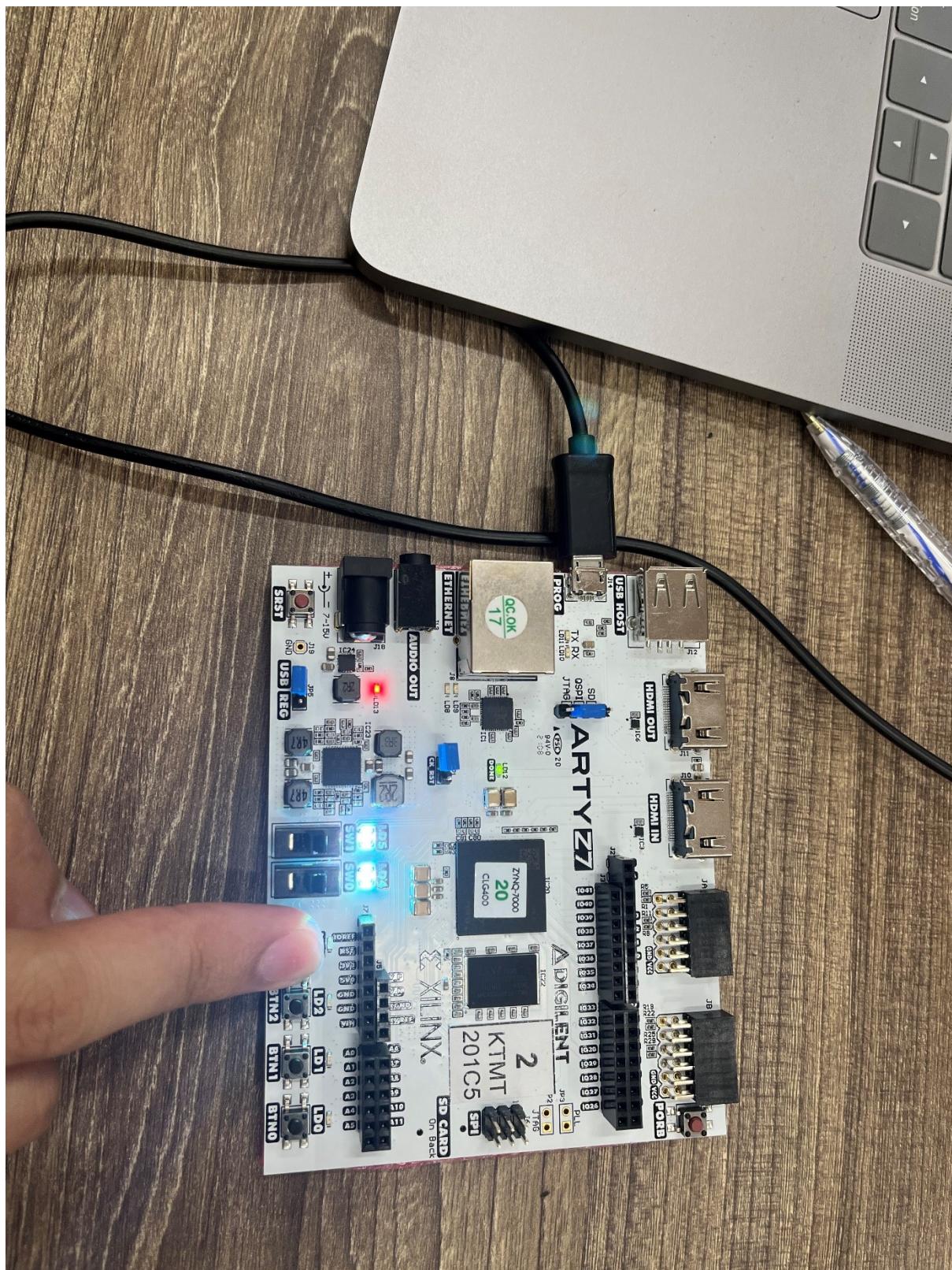


Figure 35: LED WHITE when switch1 is 1 switch1 is 1 and button is 1000