College of Information Science and Engineering, Southeast University

digital systems course design lab report

Title: Vending Machine

study Period: 2022-2023-4 (Summer School)

Student's name: Xiang Guo

Student ID:04021414

StudentName:Shu xin Wang

Student ID:04021113

Instructor.Yan Zhang

Outline.

I. System design

(Please ① brief analysis of the topic; and give ② system top-level design, can be used state machine, flowchart, etc. (not limited); give the relevant ③ input and output (I / O) planning description, do not have to correspond to the specific chip pins only to describe the IO module and state, etc.)

① Topic Analysis.

Design an FPGA module to simulate the working process of a vending machine with the following requirements.

（1） The vending machine has two input holes for coins and banknotes.The vending machine has two input holes for coins and banknotes.The recognizable range of coins is $1 coins, and the recognizable range of banknotes is $5, $10, $20, and $50. Customers can input coins several times in succession.

（2） Customers can choose from 16 kinds of products, respectively, A11 ~ A44, customers can enter the product number to realize the selection of goods. Nexys4 DDR development board has five buttons, the provisions of the two buttons (btn [3], btn [1]) as the selection of product number rows and columns of the button, each respectively, press the button rows and columns of the number plus one. Two buttons can complete the selection of rows and columns, reducing the number of ports.

(3) After selecting the product, the customer can choose the desired quantity, which is 1-3 pieces. In the same way as selecting the item number, two buttons are used to increase and decrease the quantity of the item respectively.

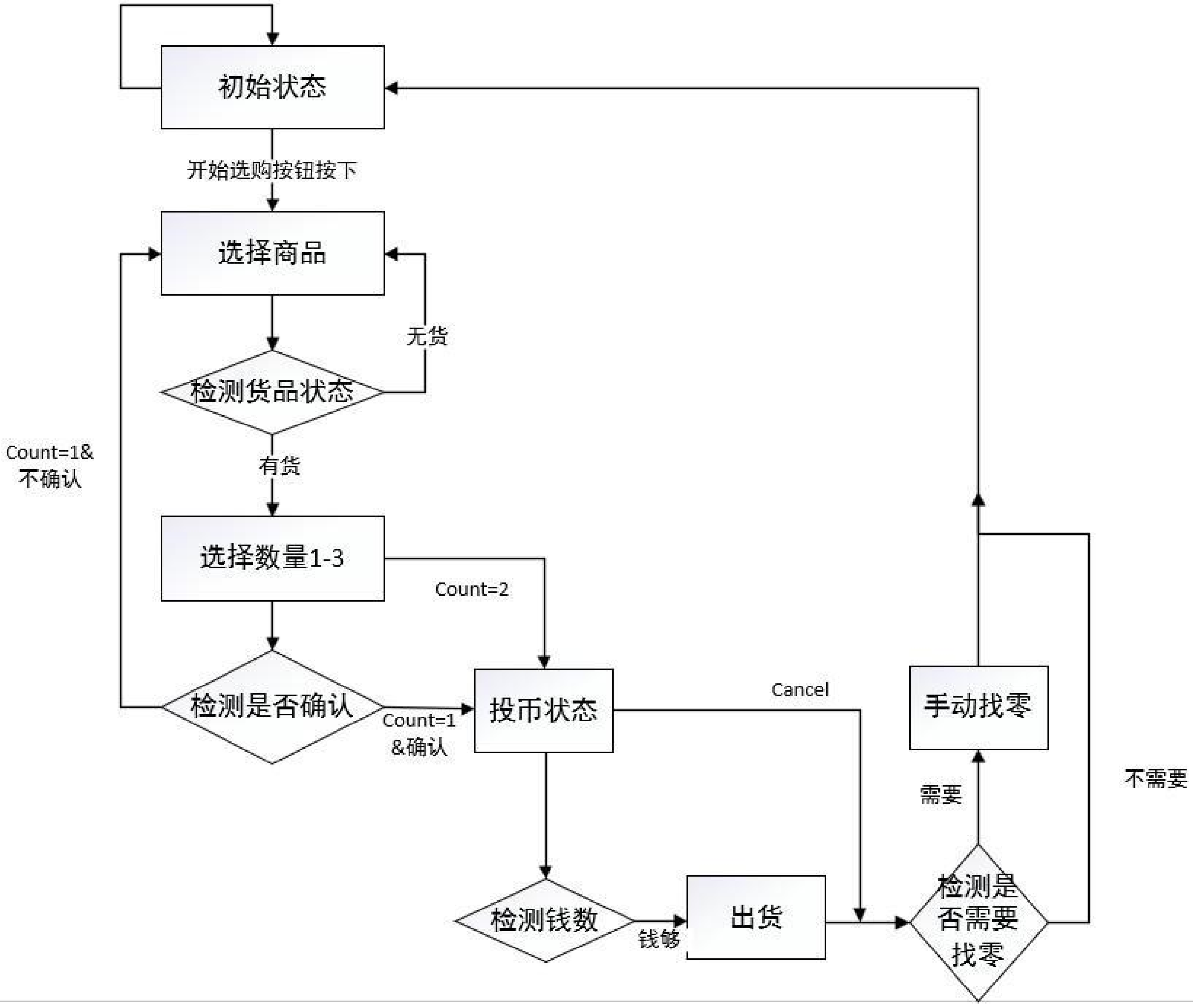
(4) Then you can continue to select the product and its quantity (you can also choose to press the skip key (btn[2]) to skip the selection of products directly into the coin section), up to a maximum of two times to select the product, press the confirmation key (btn[0]), and then display the total amount of goods.

(5) Customers can put in five denominations of coins at this time, while the digital tube displays the current total value of coins that have been put in.

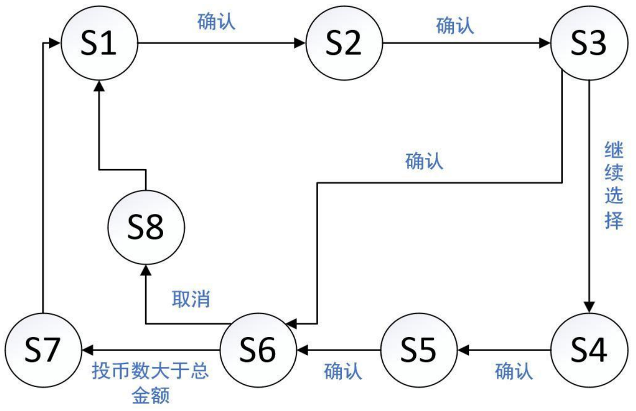
(6) When the coin value reaches or exceeds the required coin value, the vending machine will ship and deduct the required amount and display the amount of change required. During the period of coinage, the customer can press the cancel button to cancel the operation, and press the change button to withdraw the coins.

② Systematic top-level design

(1) Flowchart.

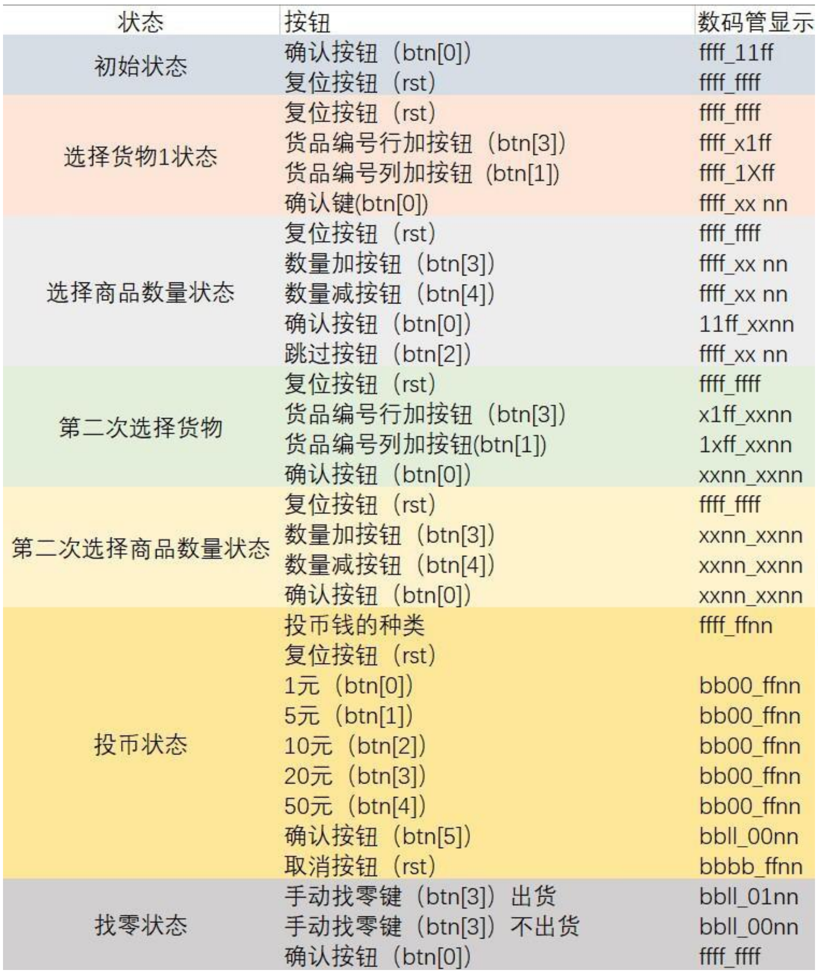


(2) Statechart.

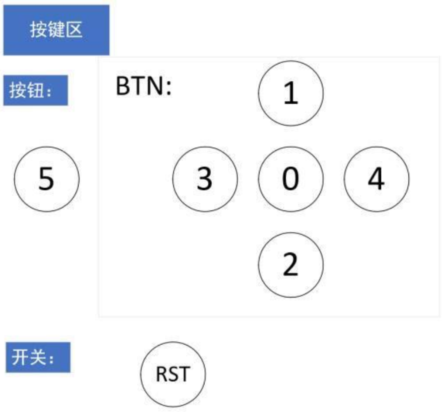


S1：初始状态
S2：选择货物1
S3：选择货物1的数量
S4：选择货物2
S5：选择货物2的数量
S6：投币
S7：出货并退币
S8：只退币

③ input/output (I/O) planning



状态	按钮	数码管显示
初始状态	确认按钮（btn[0])	ffff_11ff
	复位按钮（rst)	ffff_ffff
选择货物1状态	复位按钮（rst)	ffff_ffff
	货品编号行加按钮（btn[3])	ffff_x1ff
	货品编号列加按钮 (btn[1])	ffff_1Xff
	确认键(btn[0])	ffff_xx nn
选择商品数量状态	复位按钮（rst)	ffff_ffff
	数量加按钮（btn[3])	ffff_xx nn
	数量减按钮（btn[4])	ffff_xx nn
	确认按钮（btn[0])	11ff_xxnn
	跳过按钮（btn[2])	ffff_xx nn
第二次选择货物	复位按钮（rst)	ffff_ffff
	货品编号行加按钮（btn[3])	x1ff_xxnn
	货品编号列加按钮(btn[1])	1xff_xxnn
	确认按钮（btn[0])	xxnn_xxnn
第二次选择商品数量状态	复位按钮（rst)	ffff_ffff
	数量加按钮（btn[3])	xxnn_xxnn
	数量减按钮（btn[4])	xxnn_xxnn
	确认按钮（btn[0])	xxnn_xxnn
投币状态	投币钱的种类	ffff_ffnn
	复位按钮（rst)	
	1元（btn[0])	bb00_ffnn
	5元（btn[1])	bb00_ffnn
	10元（btn[2])	bb00_ffnn
	20元（btn[3])	bb00_ffnn
	50元（btn[4])	bb00_ffnn
	确认按钮（btn[5])	bbll_00nn
	取消按钮（rst)	bbbb_ffnn
找零状态	手动找零键（btn[3]）出货	bbll_01nn
	手动找零键（btn[3]）不出货	bbll_00nn
	确认按钮（btn[0])	ffff_ffff



Note: The "digital tube display" part of the above figure is the number of eight digital tubes on the FPGA board displayed in each state, of which "ff" represents no display, "x" represents the type or quantity of goods selected according to the key, "nn" represents the total amount of current goods, "bb" represents the total amount of current coins, "ll" represents the total amount of current coins, "lll" represents the total amount of current coins. ff" represents no display, "x" represents the type or quantity of goods selected according to the key, "nn" represents the total amount of current goods, "bb" represents the total amount of current coins, and "ll" represents the total amount of current change. In addition, 01 in the change status represents the current shipment, and 00 represents no shipment.

II. Module design

(Introduce and describe the specific state module or process used in the system design, only give the function of the relevant module or process, and define the input/output signals, and the relationship between modules/processes, do not need to give the specific code.)

The system consists of four modules, which are state machine module, keystroke debounce module, digital tube display module and led light display module.

1. state machine module

The state machine module is divided into seven states, according to the order of the process are the initial state (s0), the first product category number (line) selection state (s1), the first product quantity selection state (s2), the second product category number selection state (s3), the second product quantity selection state (s4), the coin state (s5), the coin return and change state (including the shipment and the direct return of coins in two cases) (s6).) (s6).

2. keystroke deactivation module

Since the keys are mechanical buttons, there will be burrs when they are pressed, and the burrs will affect the judgment of the program on the keys, so it is necessary to perform dither processing on the keys before using them. Here the shaking method is to start counting when the rising edge occurs, and the waveform stabilizes for 20ms before it is judged as a key press. The default clock frequency of the development board is 100MHZ, after counting 999999 times, the judgment reaches 20ms, during the simulation process, in order to reduce the memory consumption of the computer, we set the stabilization time as 2ms.

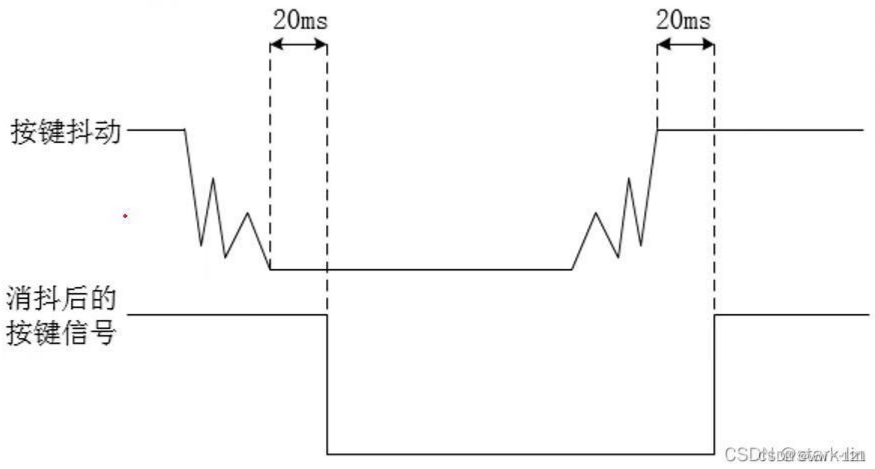


Fig. 1 Sampling after the frequency of signal change has stabilized and lasted for 20ms

The inputs of this module are clock (clk), reset signal (rst), and key signal ([5:0]btn). The output is the dither waveform ([5:0]key_out). When the main module calls the module, clk corresponds to the main module clk, rst corresponds to the main module rst, btn corresponds to the main module btn, and key_out corresponds to the main module key, and is used to judge the key presses afterwards.

3. Digital Tube Display Module.

The role of the digital tube is to display in different states corresponding to the need to show the value, where the display of the digital tube is a dynamic display, take turns to send out the bit selection signal to each digital tube, the use of the principle of human visual temporary storage, so that people feel that the digital tube at the same time in the display, the actual digital tube is a one-by-one turn to display the rotation speed is very fast, the human eye can not distinguish it.

The inputs are the clock signal (clk) and the values displayed by the 8 digital tubes (smg7, smg6, smg5, smg4, smg3.) The inputs are the clock signal (clk) and the values displayed by the 8 digital tubes (smg7, smg6, smg5, smg4, smg3),

The outputs are digital pipe enable signal (led_an) and digital pipe pin signal (led_ca). The calling part in the main module is: clk corresponds to the main module clk, smg7~0 corresponds to the main module smg7~0, and led_an corresponds to the main module smg7~0 led_an,led_ca for the main module led_ca.

The digital tube needs to indicate not only the number 0 ~ 9, but also need to indicate the letter BEGIN, with the output characters a, b, c, d, e, respectively, corresponding to the display of B, E, G, I, N five characters required for the pin.

4. led lamp display module

The role of the led light display module is to see more clearly which state the customer is in when selecting products. 16 led lights on the FPGA board, S0~6 state will be divided into six groups of running lights, respectively, 4,2,2,2,2,2,2. For example, when in the first state, the first four led lights every 0.5s sequentially light up and cycle, and so on, S1 ~ 6 state are two led lights sequentially light up to led lights to indicate the state, clear and concise, but also for the experimental process of debugging the board to find out the problem provides a convenient.

Principle of the led light display module.

Flow light, that is, to LED like flowing water on and off. Light up the first one for a period of time, and then light up the second one at the same time when the first one is off, etc. Here, we set each LED to light up for 0.5 seconds. Here, we set the time for each LED to light up to 0.5 seconds.

Since the clk used in this design is 100MHz, 50000000 cycles are exactly 0.5 seconds. Since the counter starts counting from 0, the counter only needs to record up to 50000000-1.

According to the information, in verilog, the first special function of "{}" is bit splicing, {a,b} is equivalent to splice a and b as a whole, and the high bit is a and the low bit is b. The output of led is always 3 zeros and 1 zero. {a,b} is equivalent to splice a and b as a whole, and the high bit is a and the low bit is b. The output of the led is always 3 zeros and 1 one, and the position of the one is shifted every 1 second, from the beginning to the end, and then to the beginning again. This phenomenon can be realized by using the idea of shifting. That is: led[3]<=led[2]; led[2]<=led[1]; led[1]<=led[0]; led[0]<=led[3]; if the assignment is composed of a whole, that is led, the assignment is composed of a whole is {led[2:0], led[3]}.

When the initial value of 4'b0001 is assigned to the led, the first led lights up; after 0.5 seconds, output 4'b0010, the second led lights up; after 0.5 seconds, output 4'b0100, the third led lights up; after 0.5 seconds, output 4'b1000, the fourth led lights up; after 0.5 seconds, output 4'b0001, the first led lights up, and so on, the cycle goes on and on. After 0.5 seconds, when outputting 4'b1000, the fourth led lights up; after 0.5 seconds, when outputting 4'b0001, the first led lights up, and so on, the cycle repeats itself, forming a running light.

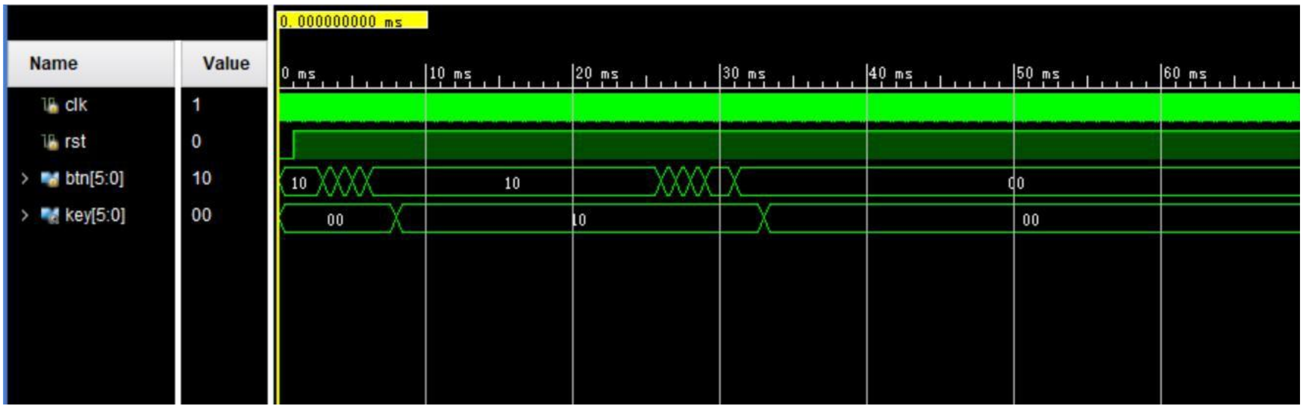
III. Experimental results and conclusions

(Including simulation methods and simulation files, simulation results, a brief description of the demonstration results, problems or subsequent improvements that can be made, and necessary experimental conclusions.)

①Dithering file code (see

appendix) Dithering

simulation results.

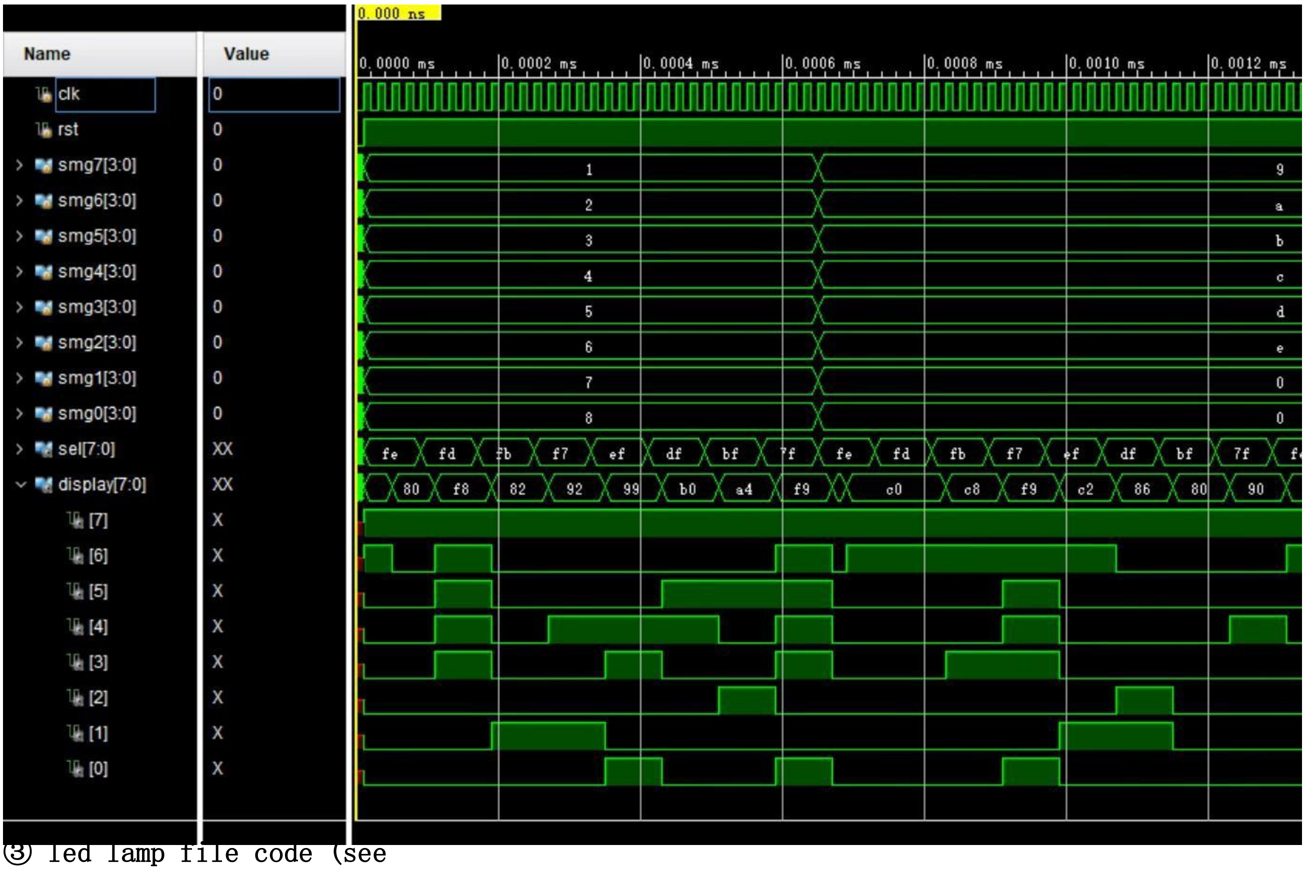


It can be seen that when the button is pressed to generate a fluctuation, the level of the waveform (key_out) after the stabilization of the fluctuation for 2ms pulls up, and the same thing happens at the end, and when the button is released and stabilized for 2ms, the level of the waveform after the stabilization pulls down

② digital tube file code

(see appendix) digital tube

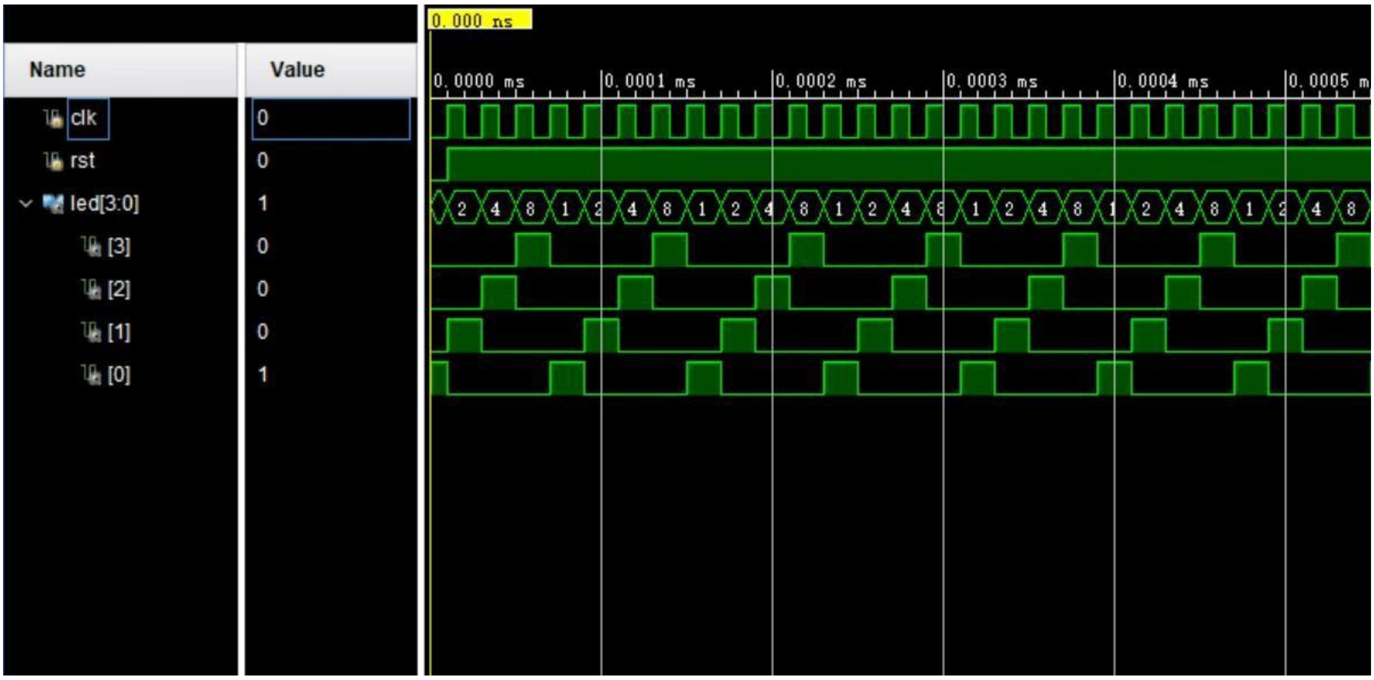
simulation results.



③ led lamp file code (see

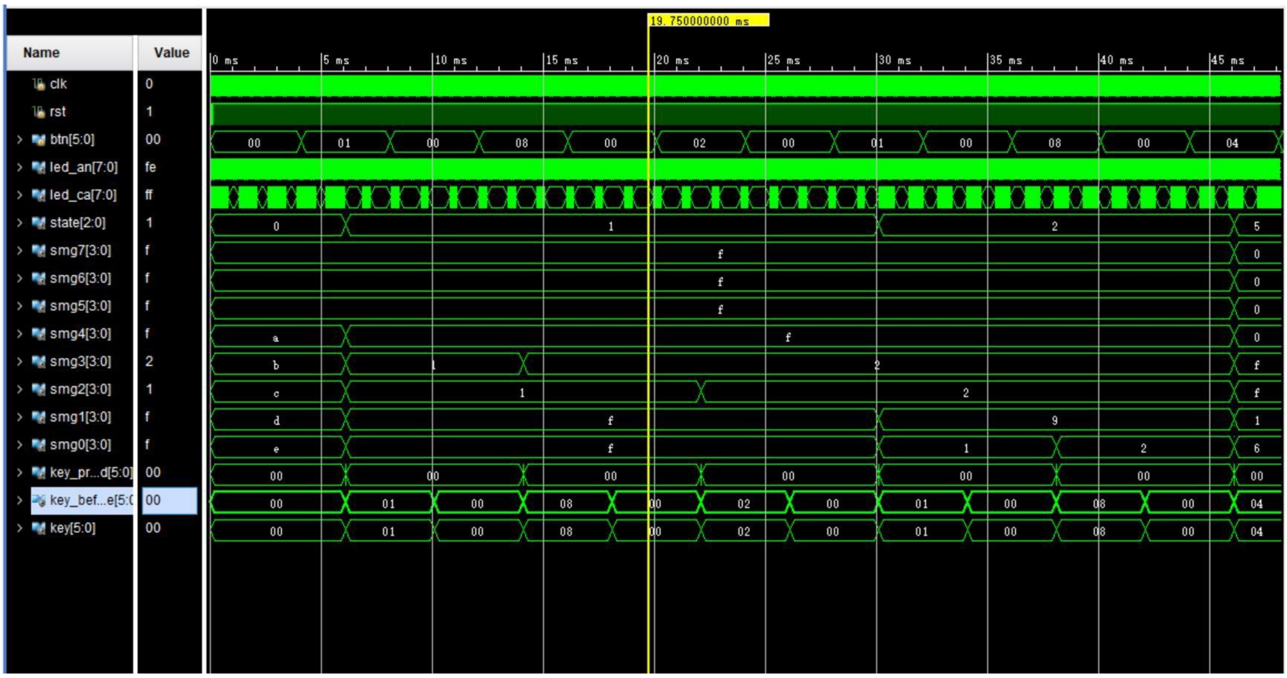
appendix) led lamp

simulation results.



From the simulation diagram, we can see that the four bits of led[3:0] generate a high level in turn, and the cycle is repeated, forming a running light.

④ Total program
simulation code:
simulation results.



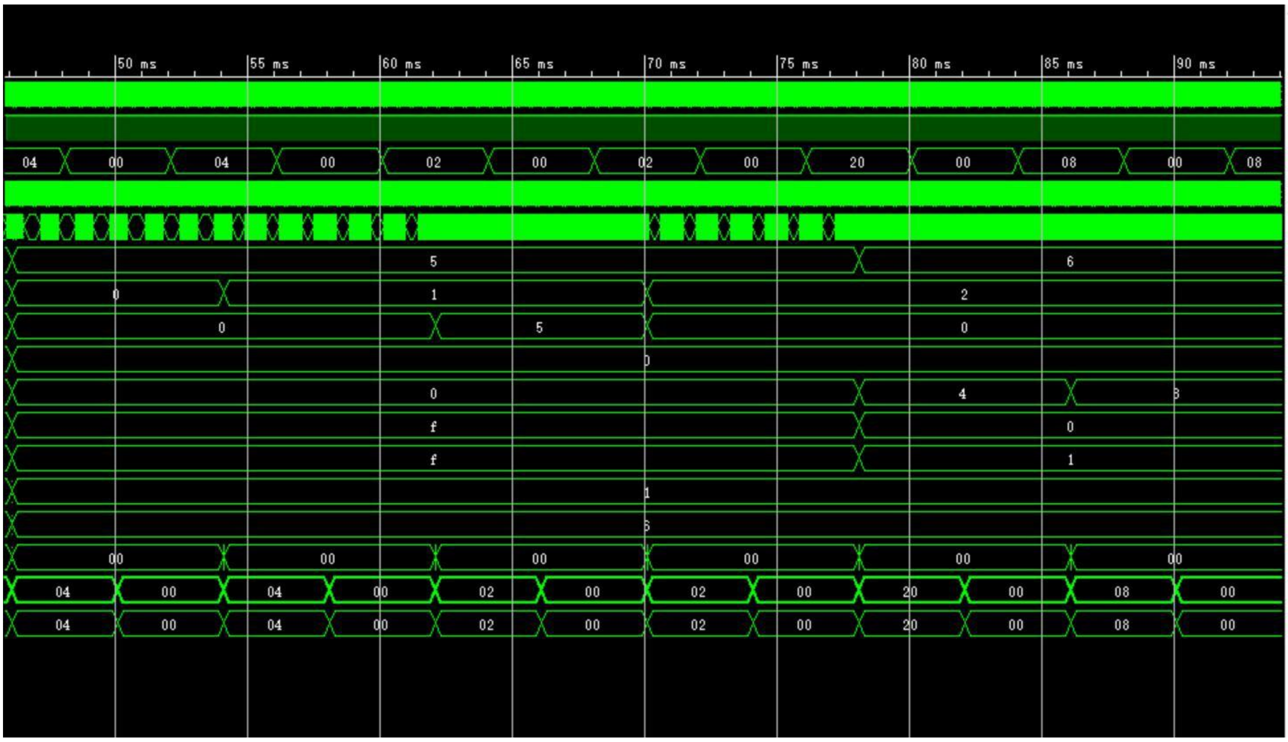
①Initial state, the digital tube displays xxxB_EGIN

② btn[5:0]=01(i.e. 000001,btn[0]) when the confirmation key is pressed, the state enters the first selection of goods

When btn[5:0]=08(i.e. 001000,btn[3]), press the downward increment key, and when btn[5:0]=02(i.e. 000010,btn[1]), press the following increment key, and the digital tube will display xxxx_22xx, and the digital tube will display xxxx_22xx

④ Press the confirmation key btn[5:0]=01 (i.e. 000001,btn[0]), enter into the state of selecting quantity, the digital tube displays xxxx_2291 (9 is the stock, 1 is the selected quantity).

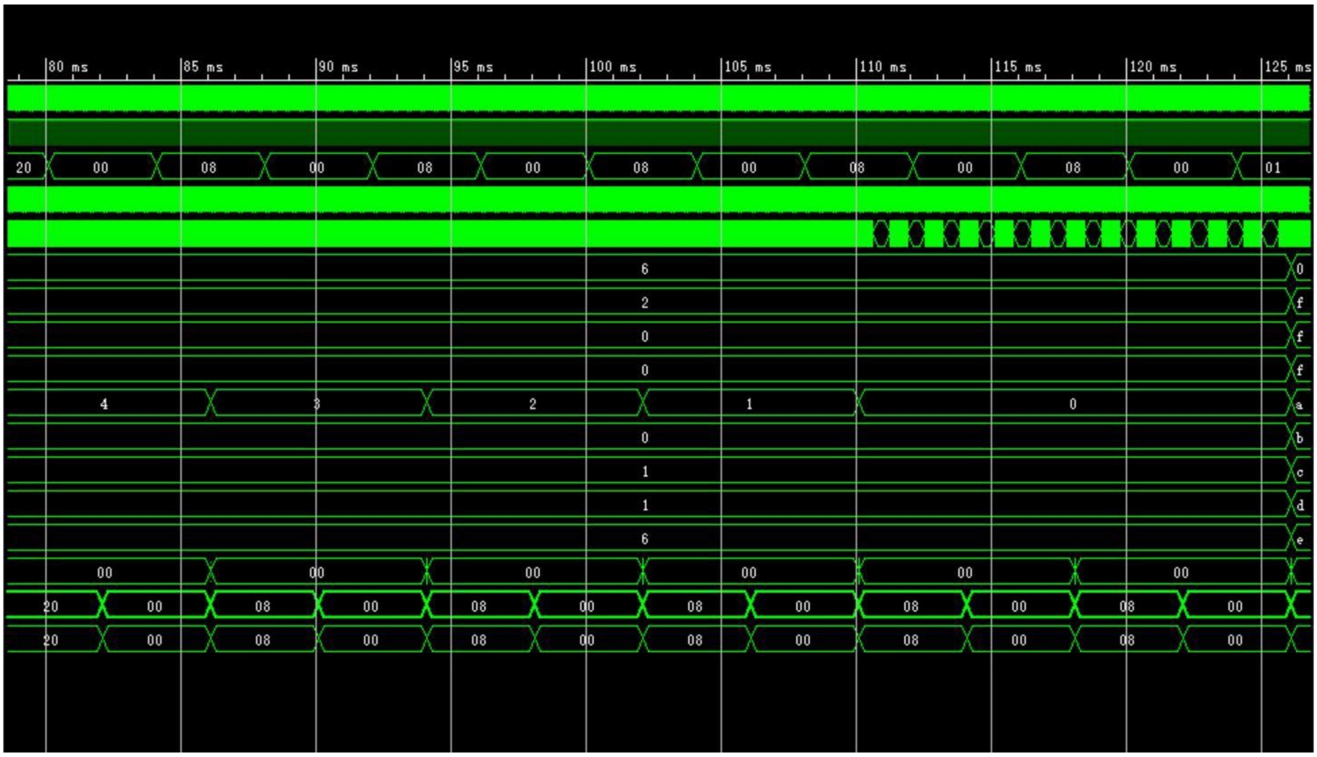
⑤ Press quantity increase key btn[5:0]=08(i.e. 001000,btn[3]), the quantity will be increased by one, and the digital tube will display xxxx_22 9 2 ⑥ Press skip key btn[5:0]=04(i.e. 000100,btn[2]) to enter into coin-operated state, and the digital tube will display 0000_xx 16 (the total amount is 16 RMB).



(7) press the coin $ 10 key btn[5:0]=04 (i.e., 000100, btn[2]), and then press the coin $ 5 key btn[5:0]=02 two times, the total amount of $ 20 coins, digital tube display 20 00_xx 16

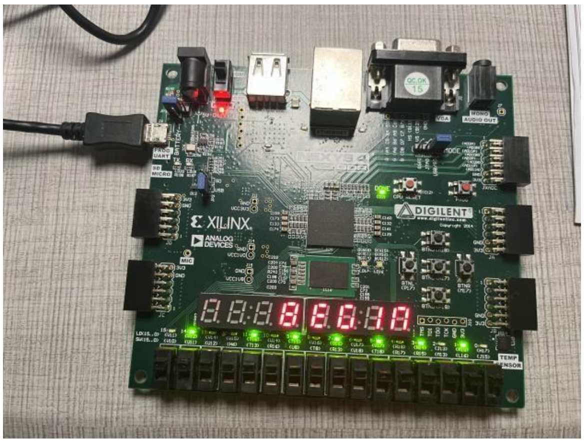
⑧ press the confirmation key btn[5:0]=20 (i.e., 100000, btn[5]), the digital tube displays 20 04 (the amount of change) 01 (has been shipped) 16

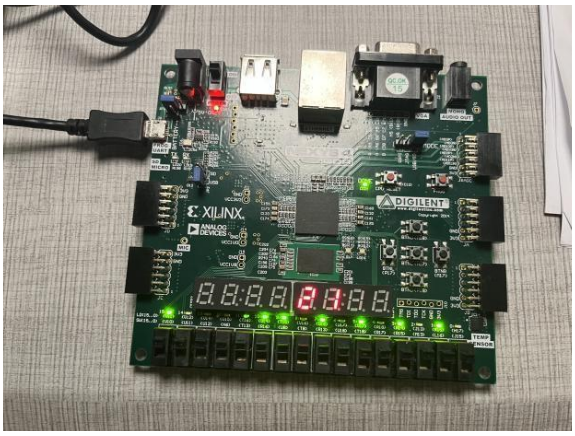
⑨ press the coin return key btn[5:0]=08(i.e. 001000,btn[3]), press four after the digital tube display 20 00 01 16

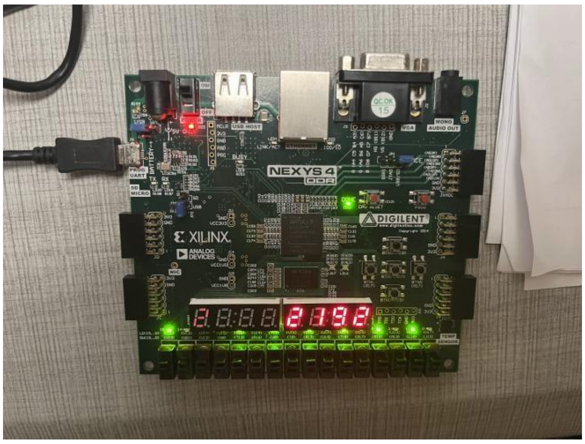


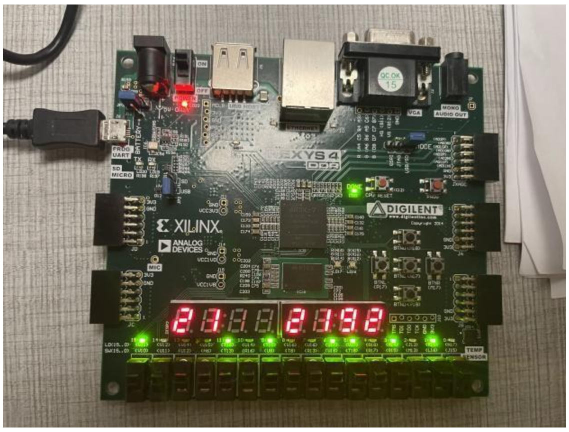
⑩ Press the confirmation key btn[5:0]=01, return to the initial state digital tube display xxxB_EGIN Physical result.

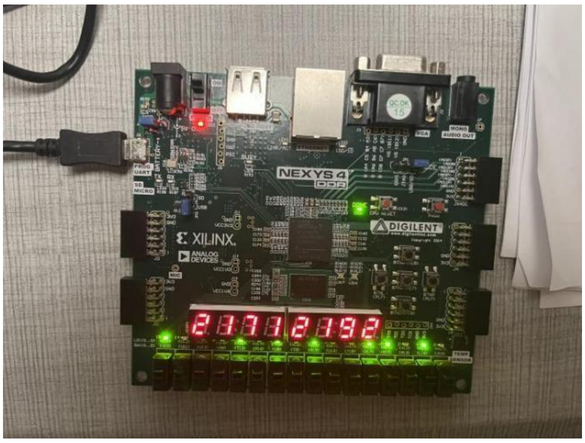
Condition I.

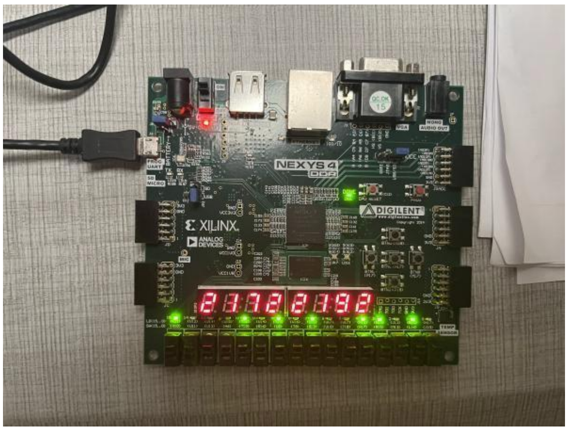


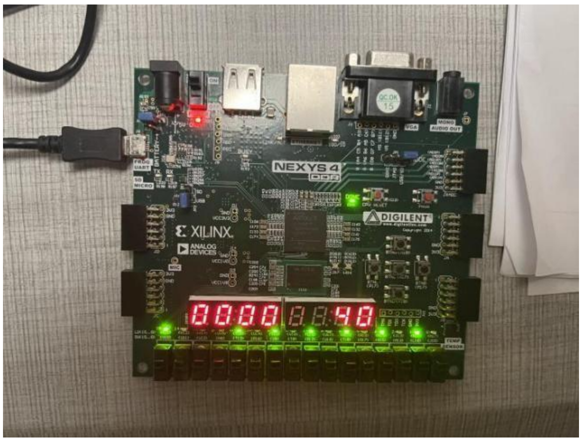


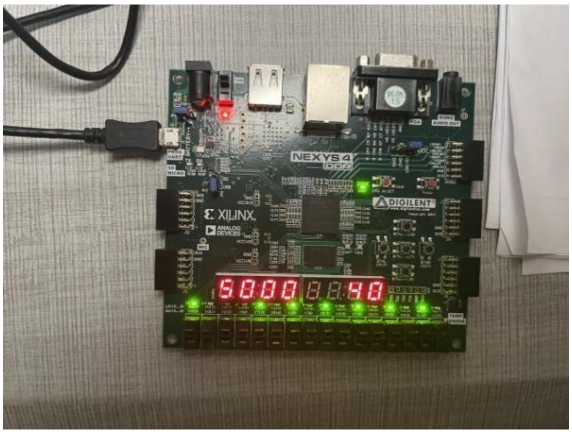


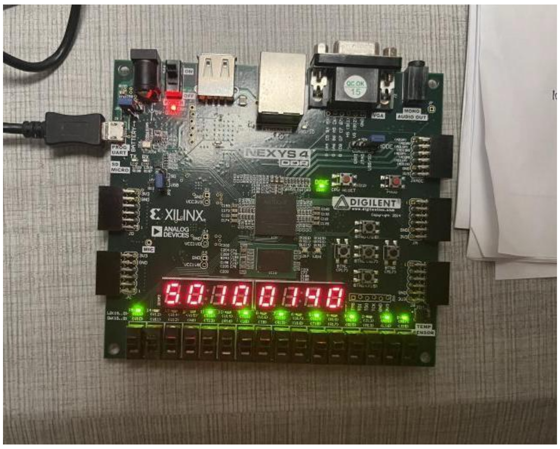


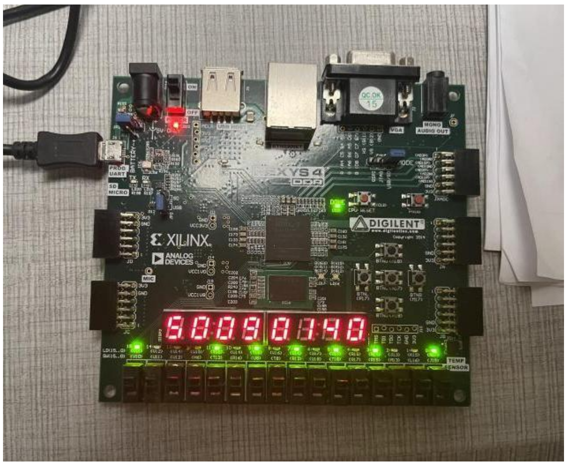

NEXYS 4 DDR
XILINX
ANALOG DEVICES
DIGILENT
21712192

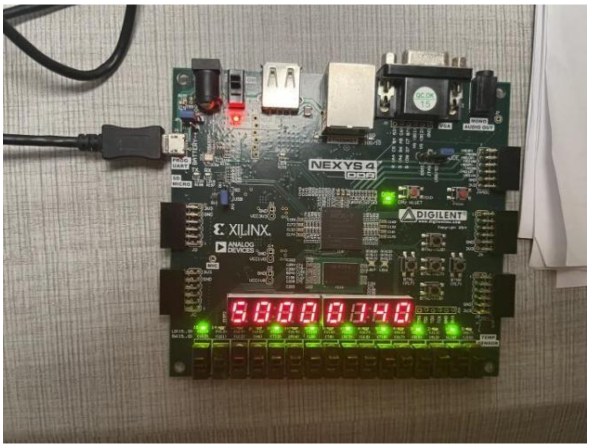

NEXYS 4 DDR
XILINX
ANALOG DEVICES
DIGILENT
21722192

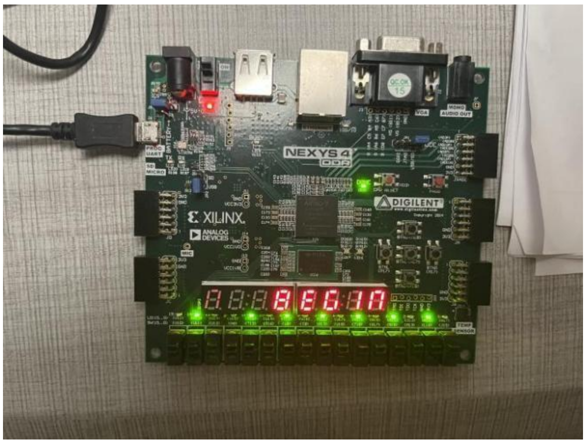

NEXYS 4 DDR
XILINX
ANALOG DEVICES
DIGILENT
0000 40


NEXYS 4 DDR
XILINX
ANALOG DEVICES
DIGILENT
5000 40

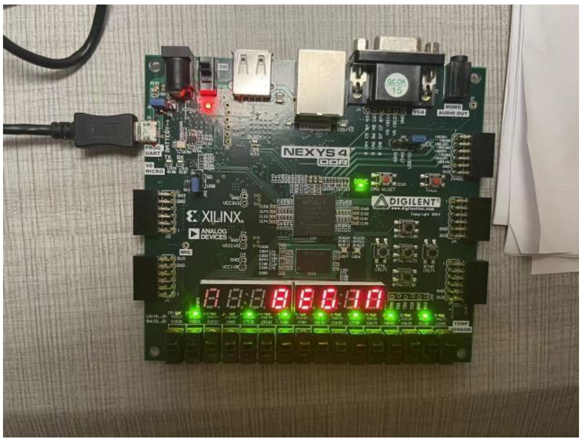

NEXYS 4 DDR
XILINX
ANALOG DEVICES
DIGILENT
50100140

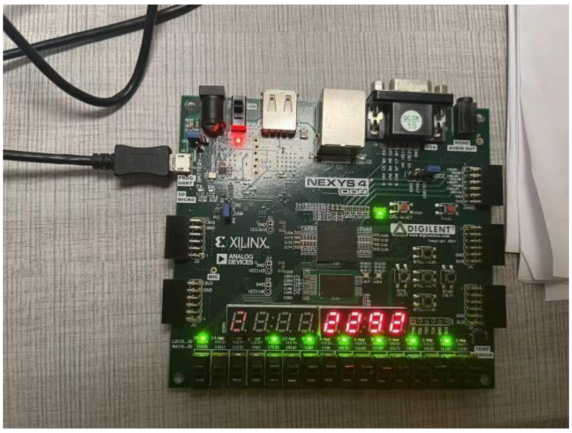

NEXYS 4 DDR
XILINX
ANALOG DEVICES
DIGILENT
50090140

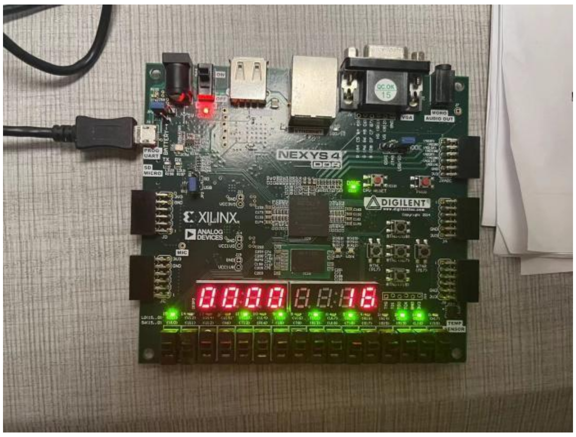


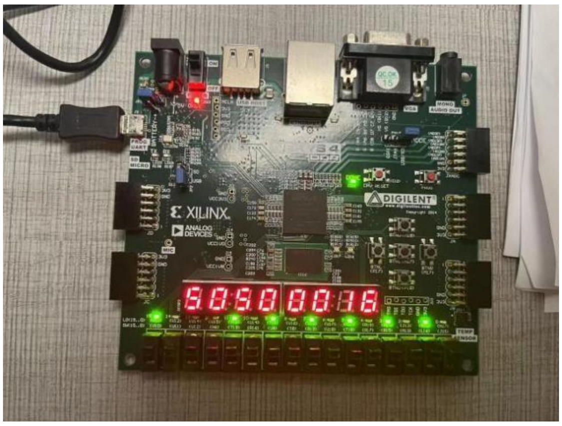


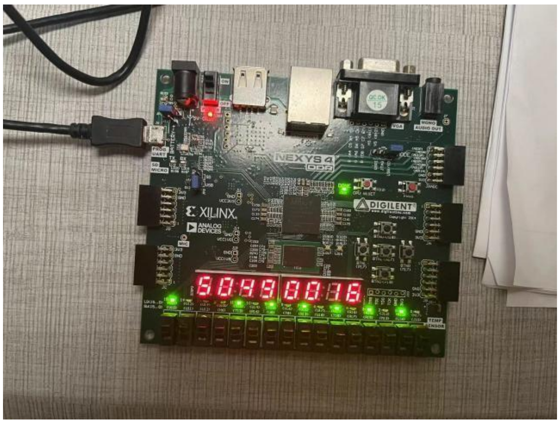
Scenario 2: Skipping the second purchase and selecting "Cancel" at the time of the coin drop.

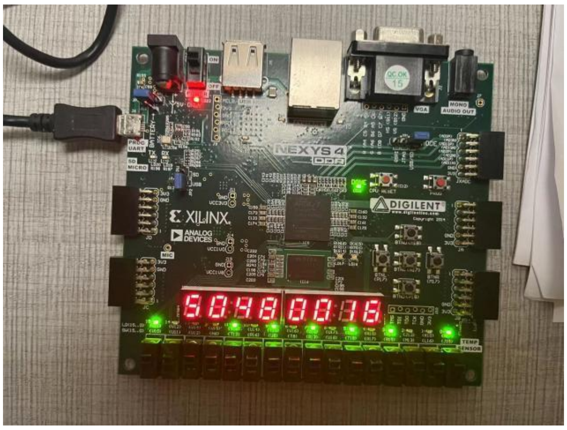


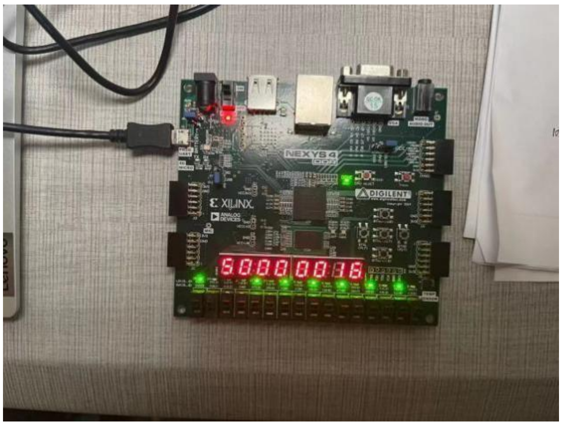


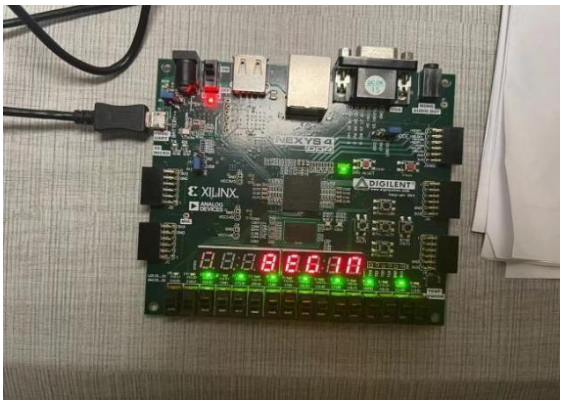












IV. Design features, and description of problems encountered during the design process and their solutions

(Please describe the characteristics of the design; please describe as detailed as possible in the design of the phenomenon of the problem, error message screenshots, etc., as well as the analysis of the relevant problems, the process of dealing with)

1. In the process of designing the state machine, it was written in a three-paragraph style at the beginning, but later it was found that the three-paragraph style would easily cause confusion, and then it was changed to a one-paragraph structure. According to the structure of state - button - realize the function of writing, so that the whole structure is easier to understand

2, in the individual responsible for the state simulation, found that when the button is pressed, directly skipped several states, the reason is that there is no press the switch, the formation of a section of the high level to meet the requirements of the confirmation of the different states, and therefore the need to add a key to confirm the module to provide a rising edge so that the state will not jump one after another.

3, at the beginning of the programmer burned into the board, the digital tube does not display any value, through the simulation found that there is no initial value, so I added initial before the code for initialization, the simulation can be achieved, but the burner is not successful. So I removed the initial in the burn code and put the initialization into always module, the digital tube can light up normally, but the display is garbled.

4, found that the digital tube appears garbled, the preliminary judgment is that the digital tube module problems. In the digital tube module in a separate assignment, and burned to the board, found that the digital tube can not be displayed normally. After checking, found in the constraints file on the enable and pin connection error, after correcting the successful display.

5, and then put it in a new short top-level file call, the digital tube and appear garbled. Guess it may be due to the counting bit is too low, change too fast, resulting in the human eye can not distinguish. The original count bit [4:2] changed to [15:13] can be displayed. 6, digital tube debugging is completed, want to carry out state transition, found that the button can not be pressed. Guess it is due to the problem of the dithering module, so temporarily give up the operation on the board, the simulation. When I simulated the dithering module separately, I changed the delay time to 2ms to verify the correctness of the dithering, because the delay time was 20ms, which required more performance from the computer. In the simulation file, I found that the assignment of the digital tube is wrong, and the assignment is not put into the state, which leads to the assignment of the initial state every time the clock appears. After the success of the simulation file, we will burn the board to realize the button function.

7, LED light module part: at the beginning of the board only the first light, the other led lights no matter what key will not light up, after debugging and checking found that the reason is that each group of led lights must be assigned at the same time the initial value, in order to carry out the cycle of running lights. And then modify the code to solve the problem, seven groups of led lights can be lit, and will be according to the current state of the corresponding lamp group began to run water light up.

8, the main module part: to find the zero part of the beginning of the search for zero, that is, at this time the total amount of zero to 0, and then press the manual zero key, the digital tube displays "99" (i.e., 00-1 = 99) of the situation, and then found that in the manual zero key to take effect in the conditions of the addition of the "find," the manual zero key, the digital tube displays "99" (i.e., 00-1 = 99) of the situation, and then found in the manual zero key effective condition.

The total amount of zero is not 0″ necessary conditions, you can make the customer change is completed, and then press the manual change key will not continue to make change.

V. Distribution of personnel tasks

Guo Zhong (member 1): 55%	**Shuxin Wang (member 2): 45%**
1、Responsible for the state machine s0-s4 state writing 2, the preparation of keystroke stabilization module, sampling frequency of 50Hz, each, the Sample the key value once in 20ms. 3. Modify the original dynamic use of digital tube code to make each number The code tube is capable of displaying the desired number. 4, the state of the machine on the board debugging, the realization of the vending machine-like Stable maintenance of states, transitions between states and simple transfers The operation, i.e., the corresponding state of the digital tube display, establishes A complete and stable state machine framework was developed. 5. the individual portion of the digital systems course design report. 6, the framework of the simulation program, as well as the individual responsible for The state machine part of the case is written.	**1、Responsible for the writing of state machine s5-s6** 2、Dynamic use of digital tube code writing, bugs. 3、Responsible for the display program of led lights, different colors of led The lights correspond to different states. 4. Integrate all the state machine frameworks and specific models established in the previous period Block for on-board testing. 5. The simulation program is filled out in the part where the individual is responsible for the state machine. 6. the individual portion of the digital systems course design report.