BCD to Binary Converter

Shrutee Rakshit
Mahdee Rahman
Preethi Venkatesan
Sahijnoor Mahal

Function of Circuit

- Conversion from 16 bit BCD to 14 bit binary
- Output
  - 14 bit binary outputted through LEDs
- Input
  - 16 bit BCD inputted through keyboard

9999 (Decimal) → **1111 1111 1111 1111** (BCD) → **10011100001111** (Binary)

Block Diagram

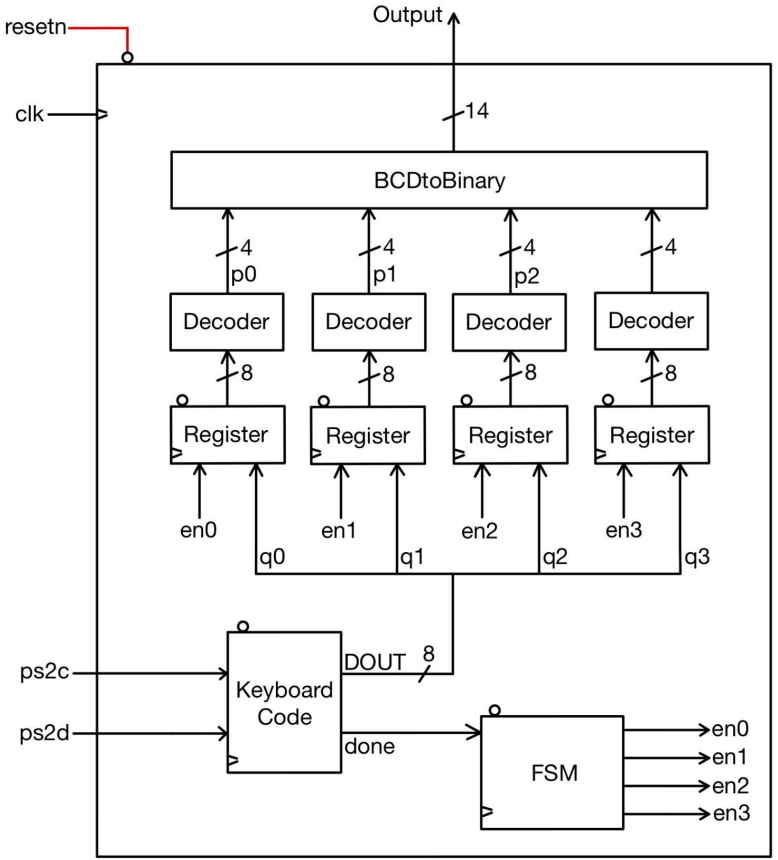
Keyboard Code: provide the output done <= 1 when a key has been released. DOUT outputs the scan code of a key.

FSM: to determine the order of the keys pressed.

Registers: store the data for the values of DOUT.

Decoders: will take the scan code of each key and output the corresponding number in BCD.

BCDtoBinary: converts BCD values from decoders into a binary output.



FSM

S1

Begins with first BCD and is the MSB. In S1 when the user lets go of a key the done will be issued. Enable 0 is now 1 which allows register 1 to be open.

S2

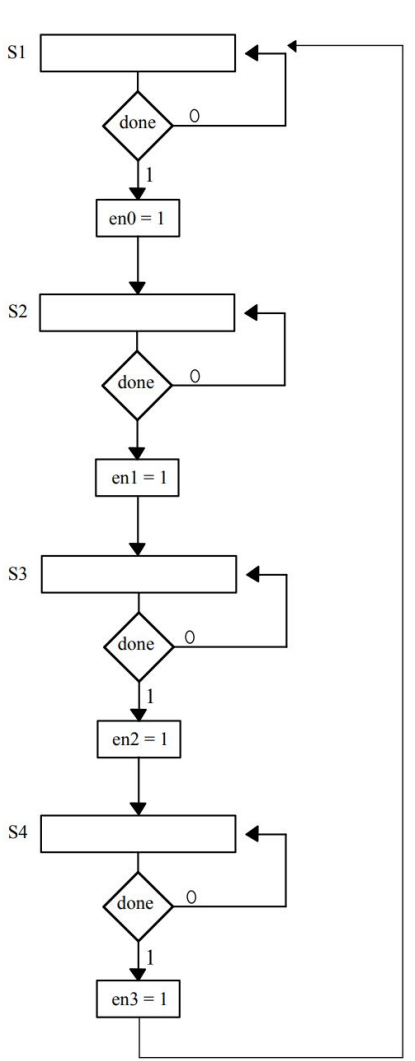
At S2, the second done is issued; at which enable 1 = 1 and this allows register 2 to be open.

S3

At S3, the third done is issued; at which enable 2 = 1 and this allows register 3 to be open.

S4

At S4, the fourth done is issued; at which enable 3 = 1 and this allows register 4 to be open.

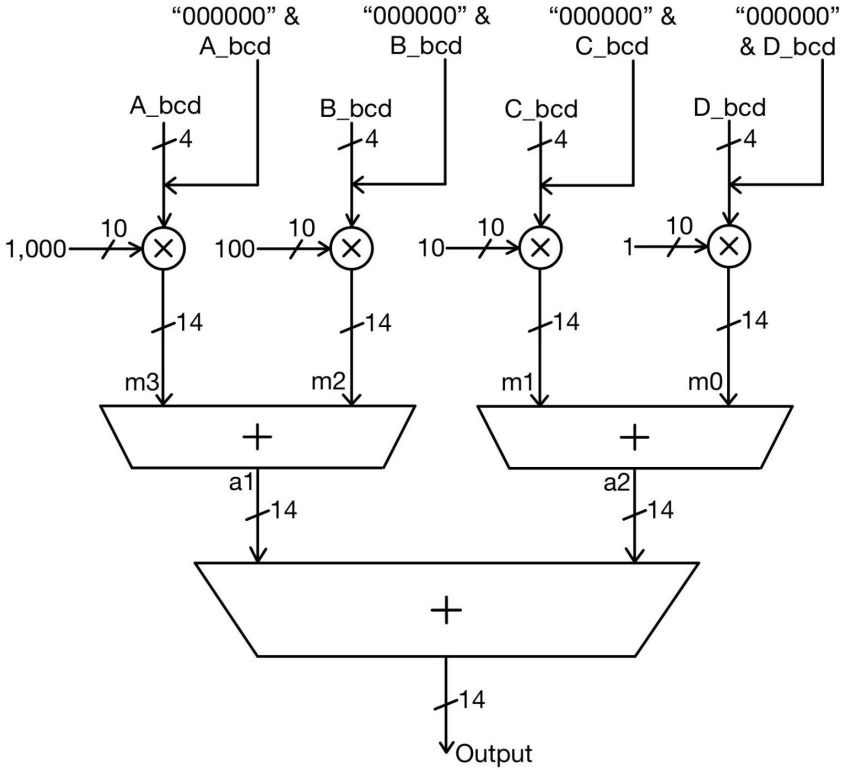


Components - BCD to Binary Converter

4 bit BCD inputs are concatenated to make them 10 bits.

The 10 bit BCD numbers are multiplied by a factor of 10 to get the outputs m3, m2, m1, m0.

The outputs from the multipliers are grouped together and added using full adders. This will output the binary of the input.



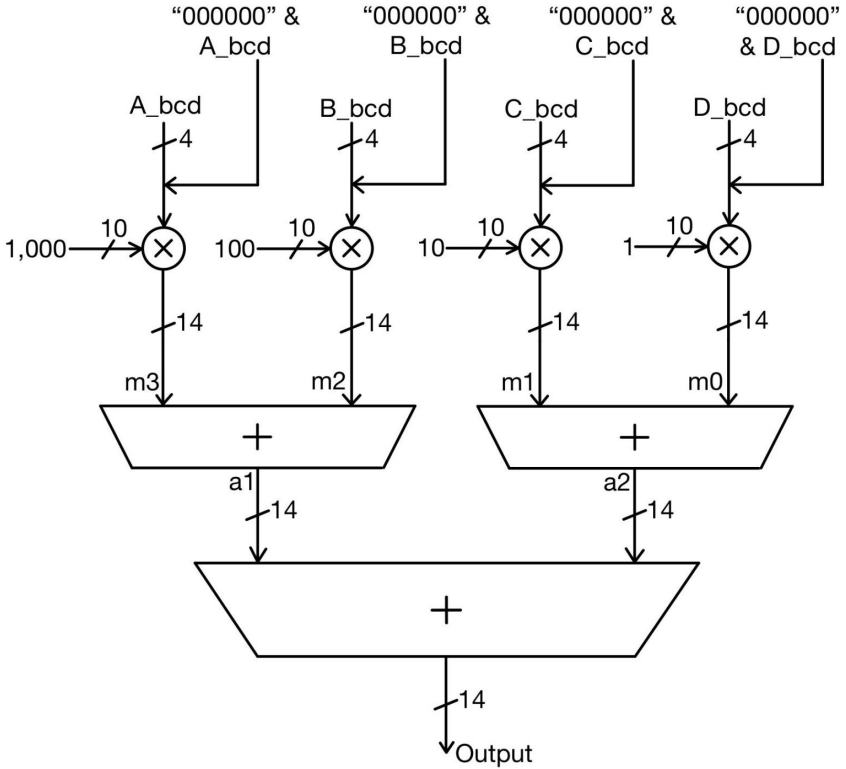
BCD to Binary Converter - Sub-Components

The output of the keyboard will be transferred to the four 4-bit registers as the BCD input. Those 4 registers will be further multiplied using 14-bit multipliers.

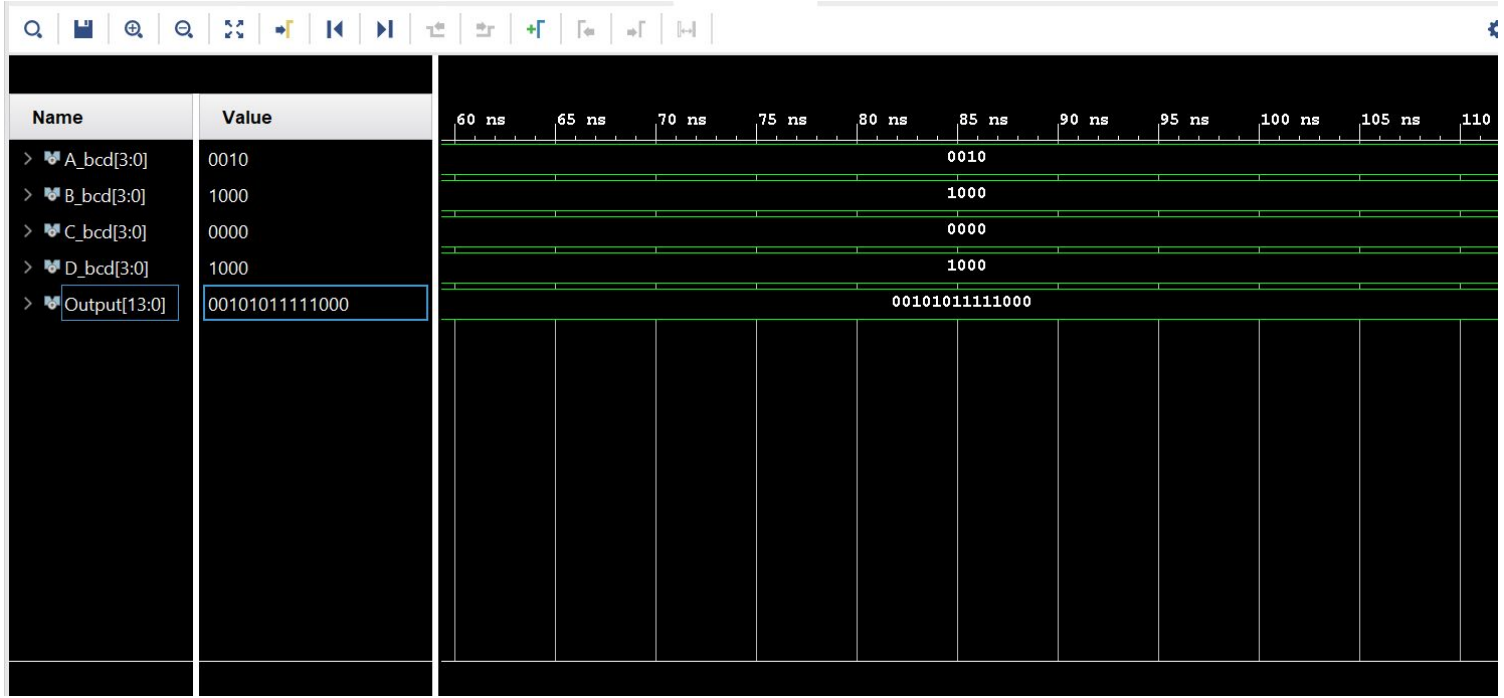
The most significant bit will be multiplied by 1000(10^3) and the second most significant bit will be multiplied by 100(10^2). The 3rd most significant bit will be multiplied by 10(10^1) and the least significant bit will be multiplied by 1(10^0).

Multiplied outputs of the most significant and the second most significant bit will be added together using a 14-bit adder and the multiplied output of the 3rd most significant bit will be added with the least significant bit using a 14-bit adder.

Output of both the 14-bit adders will then be added together using another 14-bit adder. The output of this adder would give us our binary output which will be the result of the BCD input.

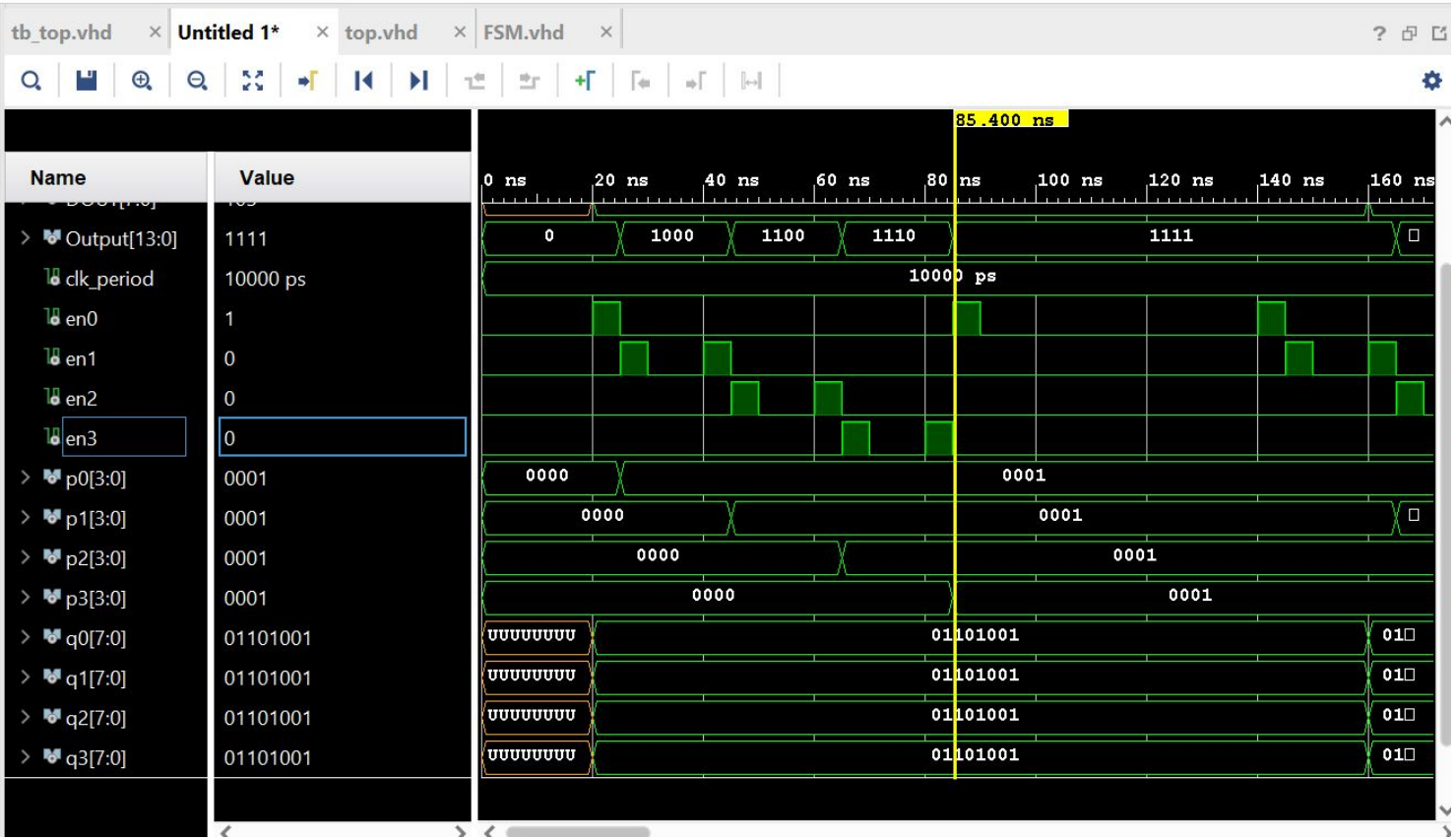


BCD to Binary Converter - Simulation

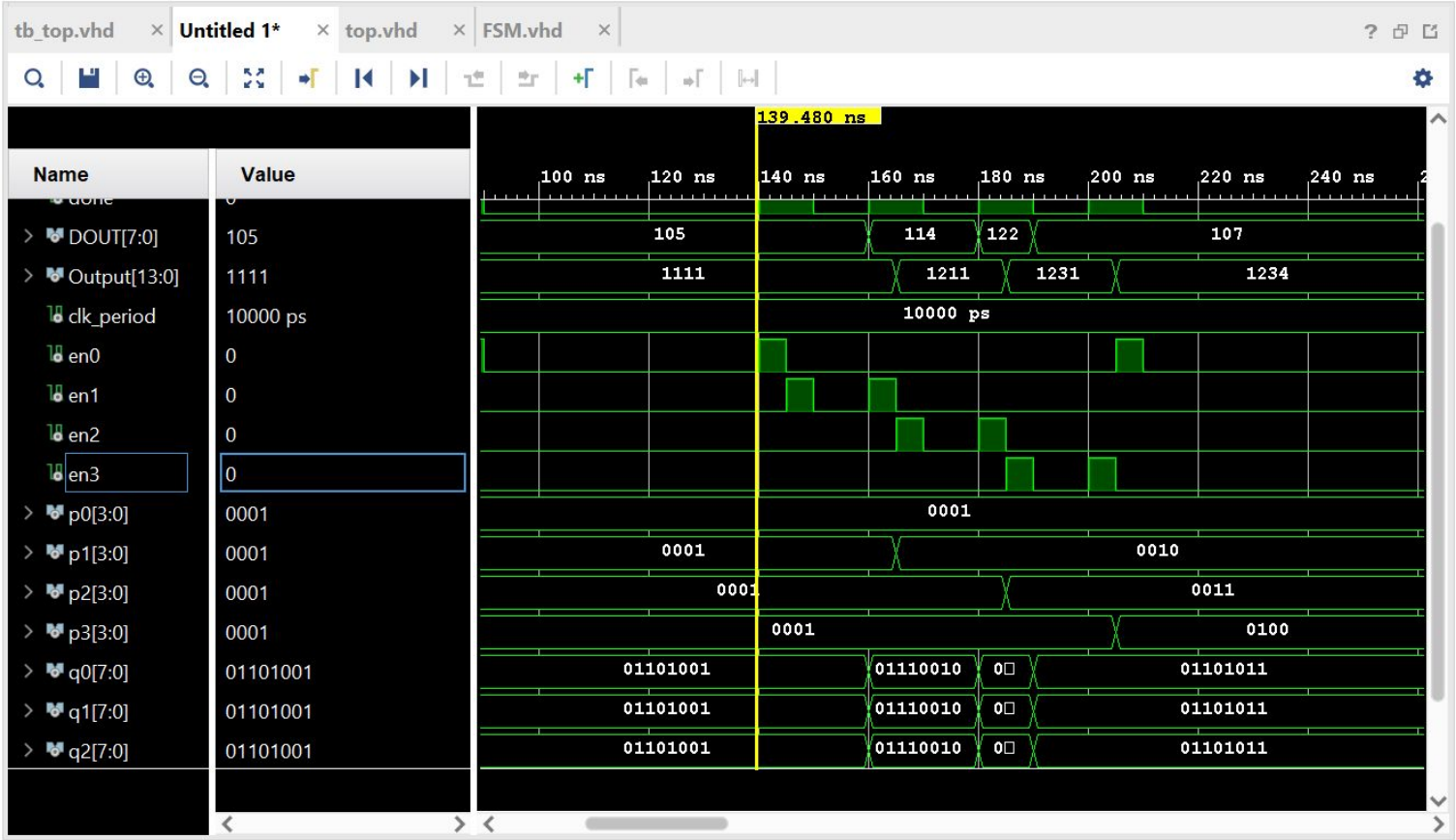


0010 1000 0000 1000 (BCD) => 2808 (Decimal) => 101011111000 (Binary)

Simulation

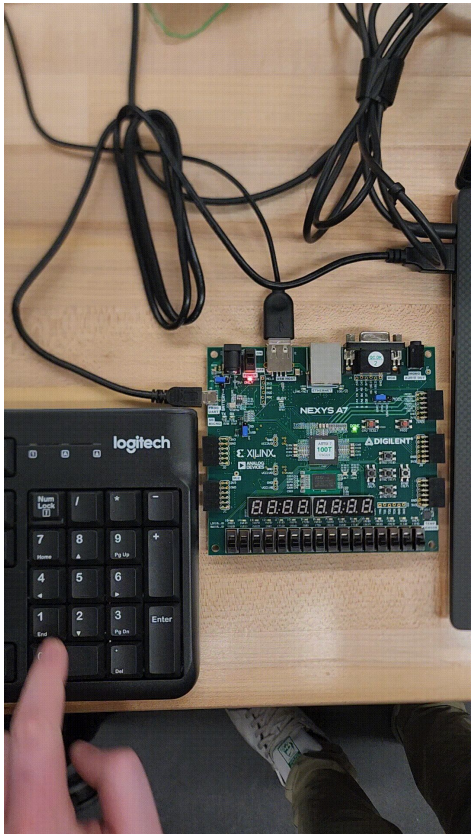

tb_top.vhd
Untitled 1*
top.vhd
FSM.vhd
Name
Value
Output[13:0]
1111
clk_period
10000 ps
en0
1
en1
0
en2
0
en3
0
p0[3:0]
0001
p1[3:0]
0001
p2[3:0]
0001
p3[3:0]
0001
q0[7:0]
01101001
q1[7:0]
01101001
q2[7:0]
01101001
q3[7:0]
01101001
85.400 ns
0 ns
20 ns
40 ns
60 ns
80 ns
100 ns
120 ns
140 ns
160 ns
0
1000
1100
1110
1111
10000 ps
0000
0001
uuuuuuuu
01101001

Simulation


tb_top.vhd
Untitled 1*
top.vhd
FSM.vhd
Name
Value
DOUT[7:0]
105
Output[13:0]
1111
clk_period
10000 ps
en0
0
en1
0
en2
0
en3
0
p0[3:0]
0001
p1[3:0]
0001
p2[3:0]
0001
p3[3:0]
0001
q0[7:0]
01101001
q1[7:0]
01101001
q2[7:0]
01101001
139.480 ns
100 ns
120 ns
140 ns
160 ns
180 ns
200 ns
220 ns
240 ns
105
114
122
107
1111
1211
1231
1234
10000 ps
0001
0001
0010
0001
0011
0001
0100
01101001
01110010
01101011
01101001
01110010
01101011
01101001
01110010
01101011

Thank you!

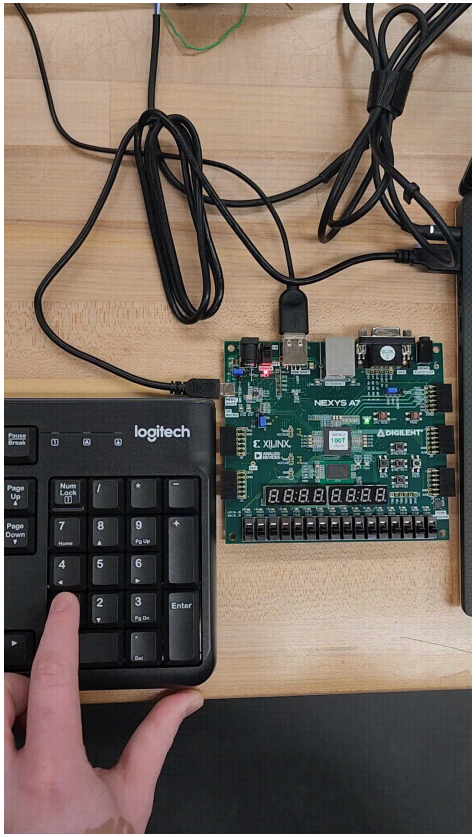
Demo



0001 0001 0001 1000 (BCD)

⇩

1010110110011 (Binary)



0001 0010 0100 0101 (BCD)

⇩

10011011101 (Binary)