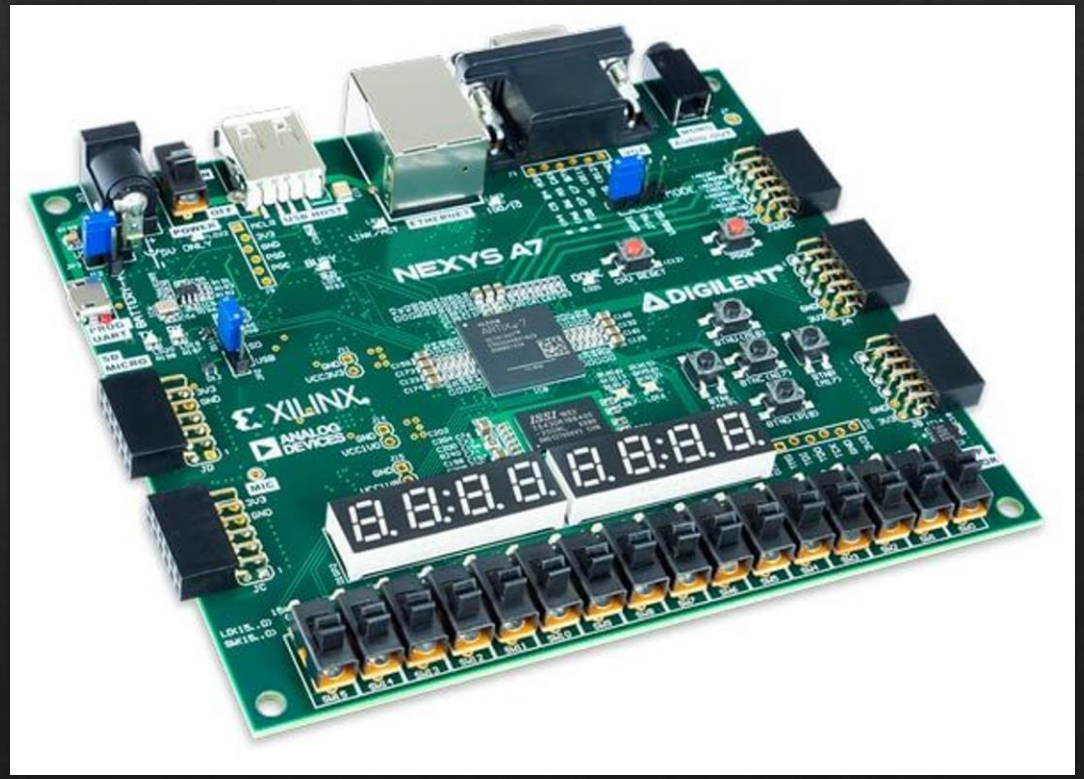
Don’t forget your bits (Combinational Lock)

Zachary Hill | Zachary Jump | Aidan Gallagher | Stephen Hayes

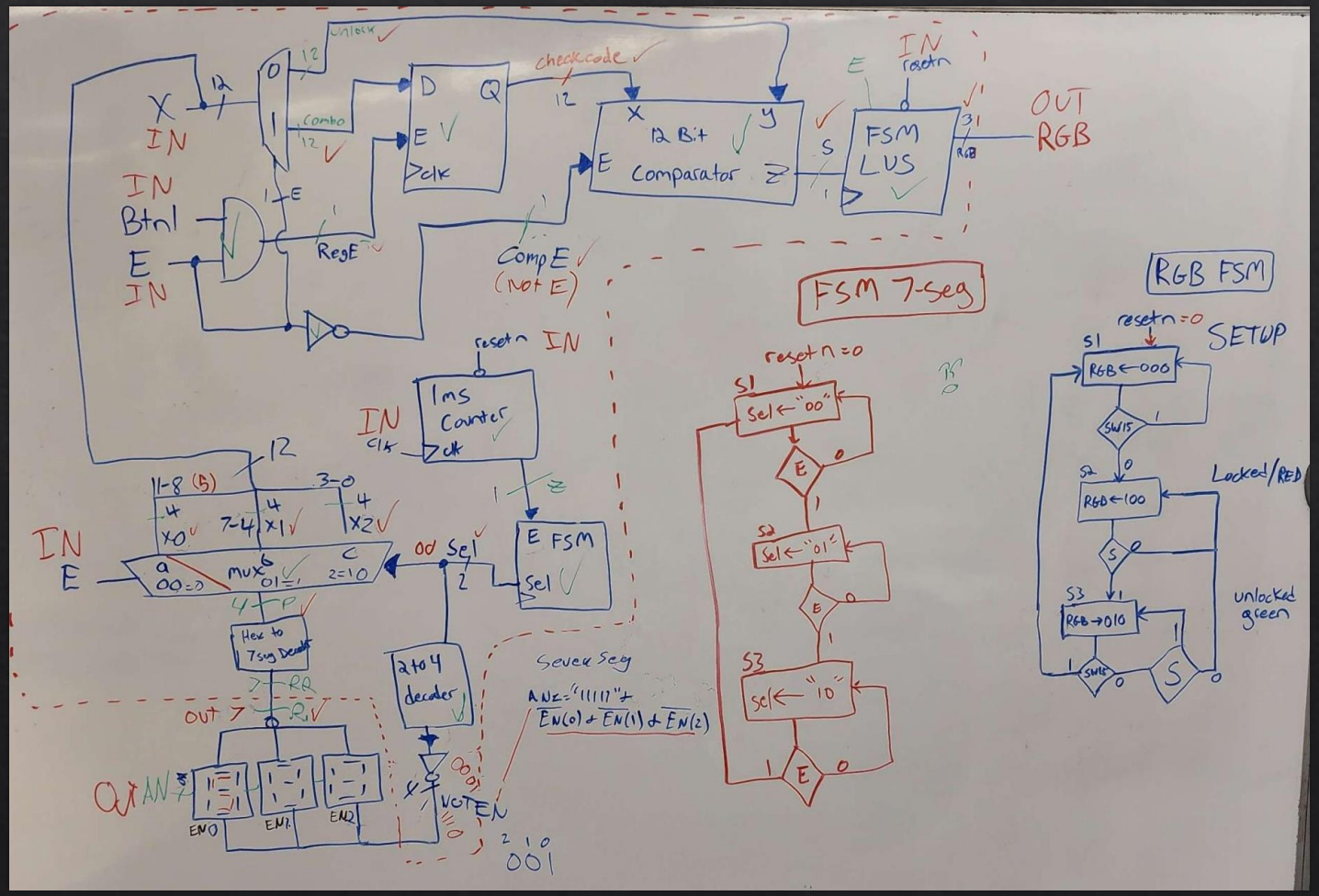
ECE 2700 | Digital Logic Design | Dr. Daniel Llamocca

Objective & Function

- This digital combinational lock has the ability to:
  - Store a user's combination consisting of 3 initial 4 bit inputs including 0-9 and A-F with a button
  - Allows the user to attempt the stored combination using the switches
  - Provide input feedback and validation outputs when guessing combination (Seven Segment Display & LED)
    - Green light = correct combination
    - Red light = incorrect combination



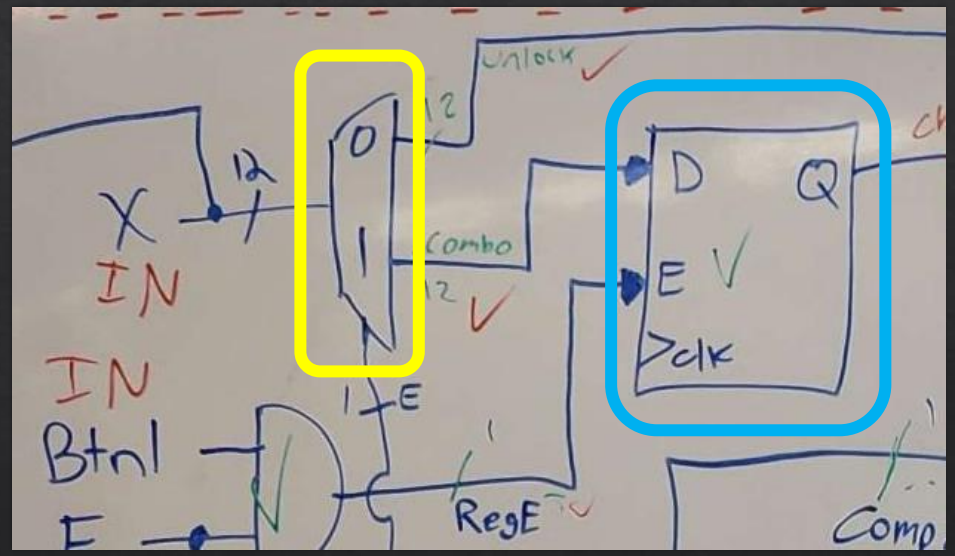
Block Diagram & Components



- Multiplexor & Demultiplexer
- Register
- Comparator
- Counter
- Decoder
- FSM(s)
  - Seven Segment FSM
  - RGB LED FSM

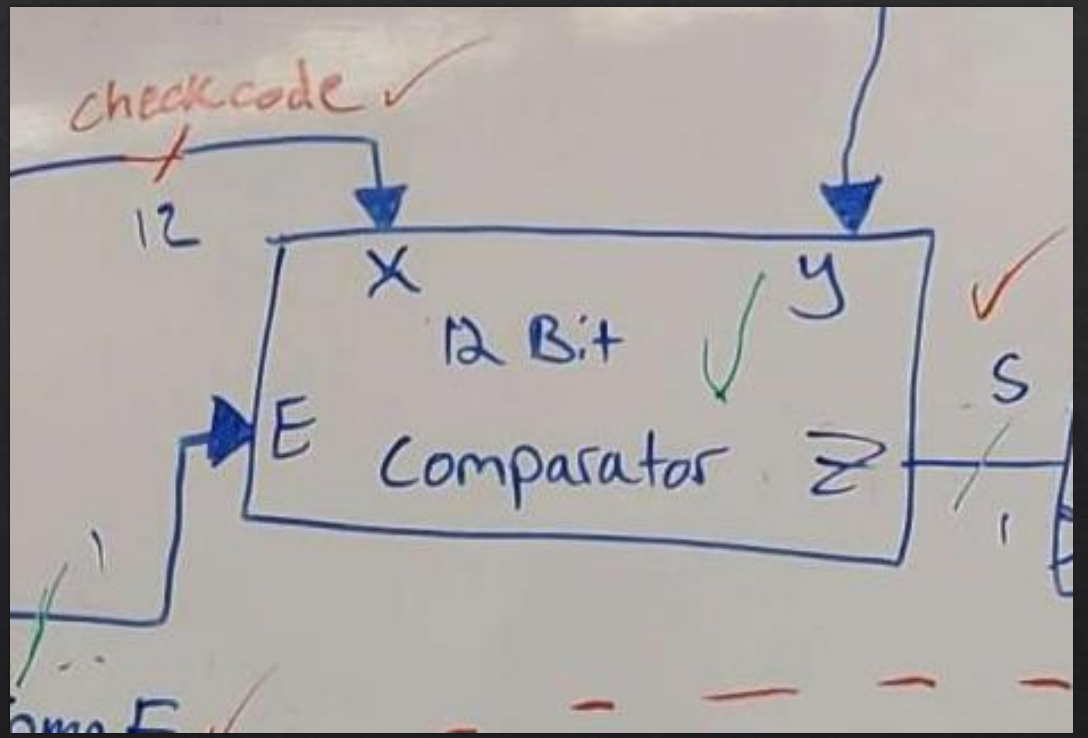
Demultiplexor (Demux) & Register

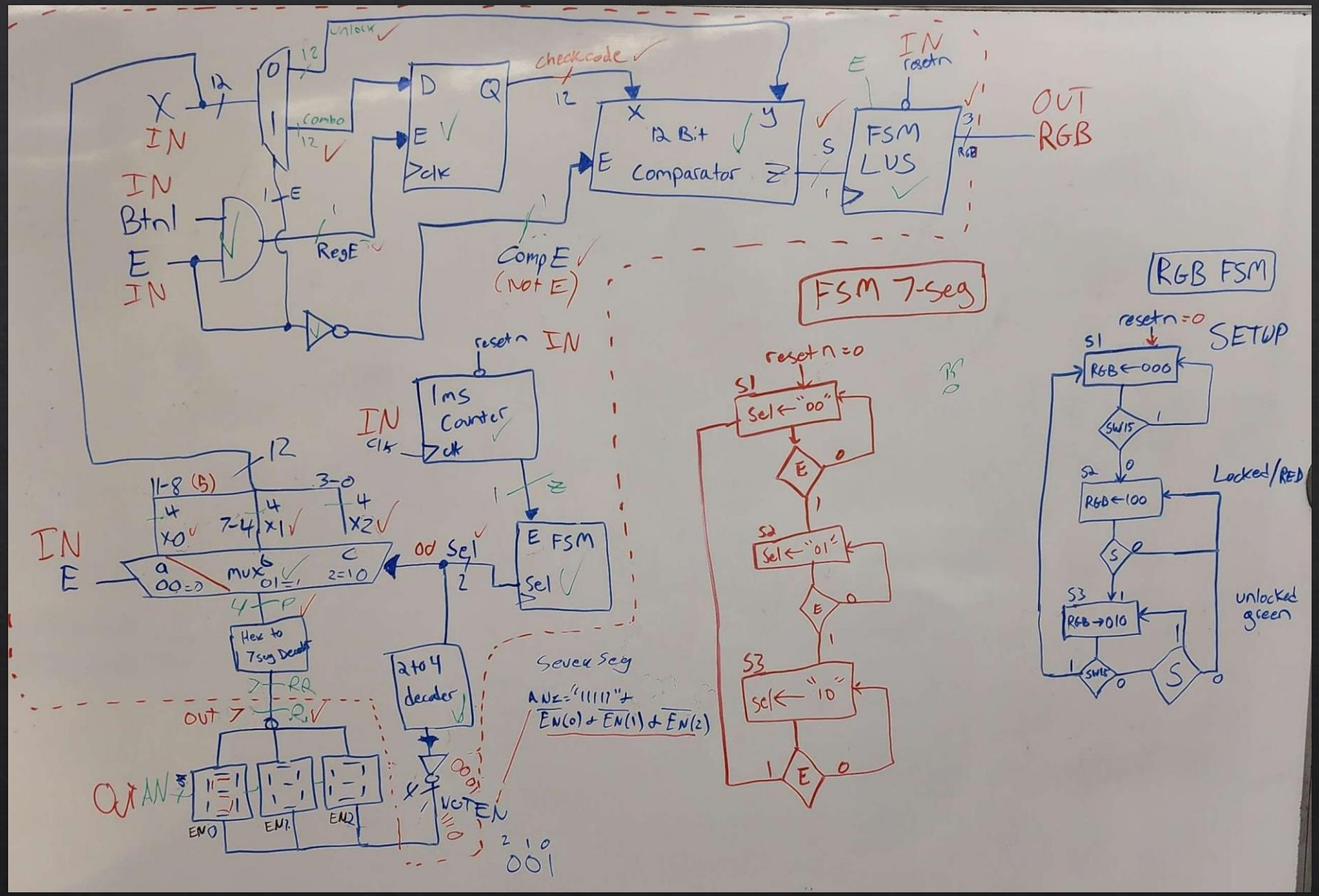
- The Demux passes the 12-bit code to the register or comparator based on the state of the enable switch
- The Register has 2 inputs: "D" & "E"
  1 output: "Q" that goes to the comparator
  - **D is the signal “combo” outputted from the Demux when E = ‘1’**
  - **E is Btn1 and the enable switch**



Comparator

- Three inputs: "x", "y", and "E"
- One output: "z"
- While "E" is '1', the component will compare "x" and "y", outputting '1' when they are equal
- While "E" is '0' the component will output "z" as '0'



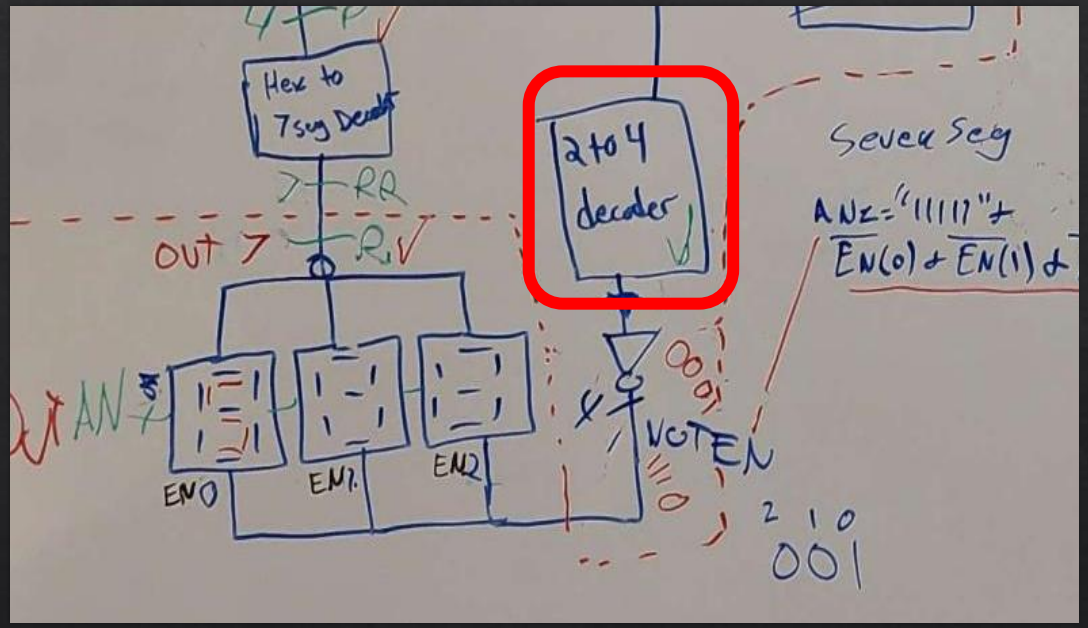

unlock
checkcode
combo
X
IN
Btnl
E
IN
RegE
D
Q
E
clk
12 Bit
Comparator
x
y
z
E
IN
resetn
S
FSM
LUS
31
RGB
OUT
RGB
CompE
(Not E)
FSM 7-seg
RGB FSM
resetn IN
1ms
Counter
clk
IN
clk
11-8 (5)
7-4
3-0
X0
X1
X2
IN
E
mux
a
00=0
b
01=1
c
2=10
00 Sel
E FSM
Sel
Hex to
7seg Decoder
RR
out 7
RV
2 to 4
decoder
AN
EN0
EN1
EN2
VCC
EN
Seven Seg
AN="1111"+
EN(0) & EN(1) & EN(2)
2 1 0
0 0 1
resetn=0
S1
Sel←"00"
E
S2
Sel←"01"
S3
Sel←"10"
resetn=0
SETUP
S1
RGB←000
SW15
S2
RGB←100
Locked/RED
S
S3
RGB→010
unlocked
green
SW15
S

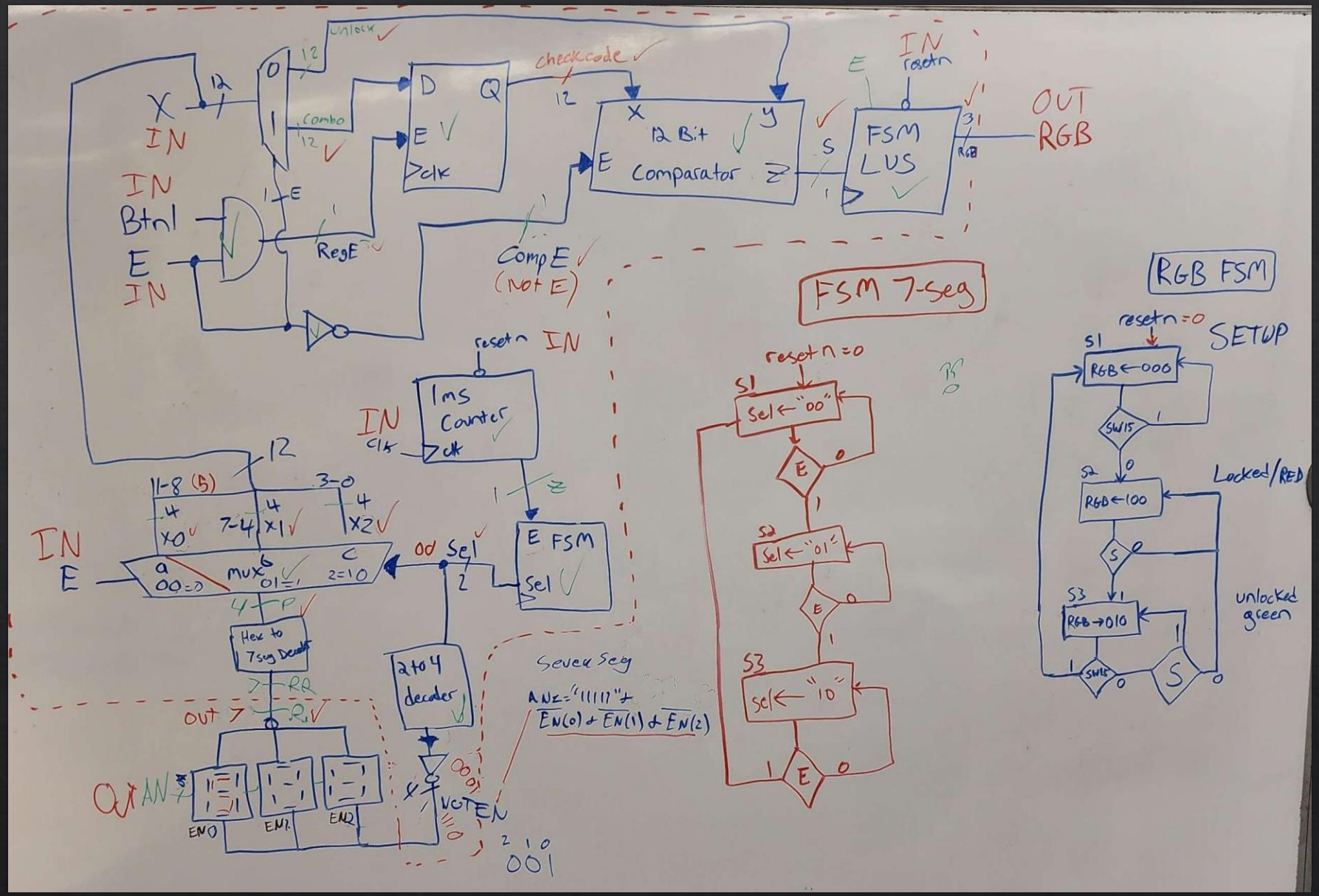
2 to 4 Decoder

- 2 to 4 decoder
  1 input: "sel"
  1 output: "y"

- The 2 to 4 decoder takes the value of sel and decodes it to activate the respected Seven Segment displays.

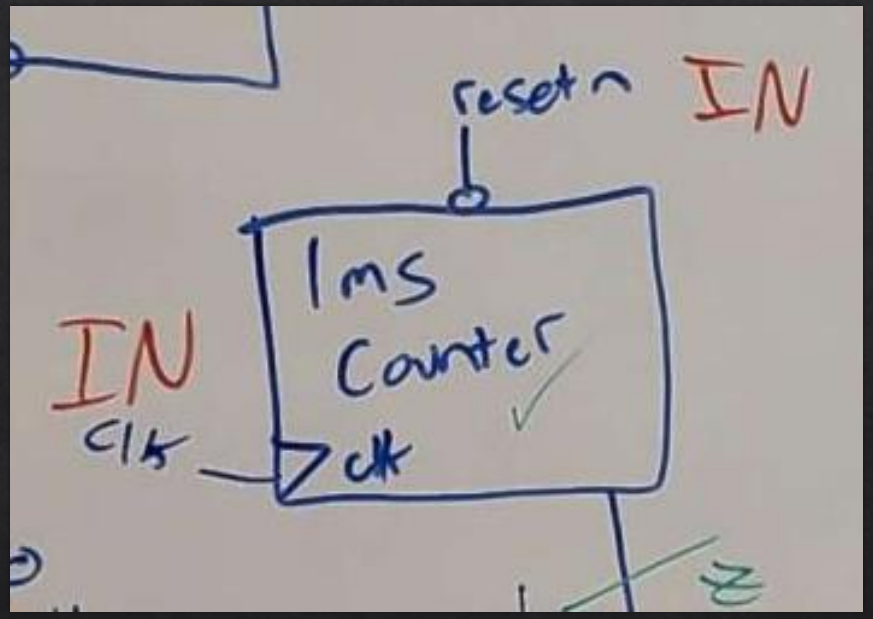
- The decoder will assign a 1 to the bit indicated by the decimal representation of the 2-bit input so 00 would decode to 0001 , 01 would decode to 0010 and so on




unlock
checkcode
combo
X
IN
Btnl
E
IN
RegE
D
Q
E
clk
12 Bit
Comparator
x
y
z
E
IN
resetn
S
FSM
LUS
31
RGB
OUT
RGB
CompE
(Not E)
FSM 7-seg
RGB FSM
resetn IN
1ms
Counter
clk
IN
clk
11-8 (5)
7-4
3-0
X0
X1
X2
IN
E
mux
a
00=0
b
01=1
c
2=10
00 Sel
E FSM
Sel
Hex to
7seg Decoder
RR
out 7
RV
2 to 4
decoder
AN
EN0
EN1
EN2
VCC
EN
Seven Seg
AN="1111"+
EN(0) & EN(1) & EN(2)
2 1 0
0 0 1
resetn=0
S1
Sel←"00"
E
S2
Sel←"01"
S3
Sel←"10"
resetn=0
SETUP
S1
RGB←000
SW15
S2
RGB←100
Locked/RED
S
S3
RGB→010
unlocked
green
SW15
S

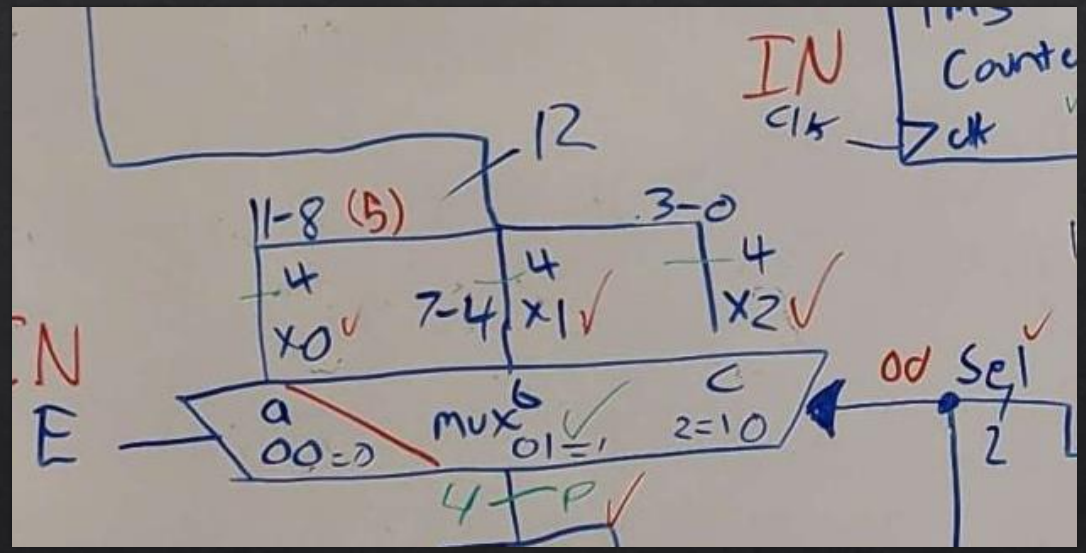
Counter

- Two inputs: "clk" and "resetn"
- One output: "z"
- At the height of every clock tick, a signal tracks how many times the circuit has ran.
- In this case, an output of '1' is given when enough cycles of T=10ns have ran to equate to 1 ms.



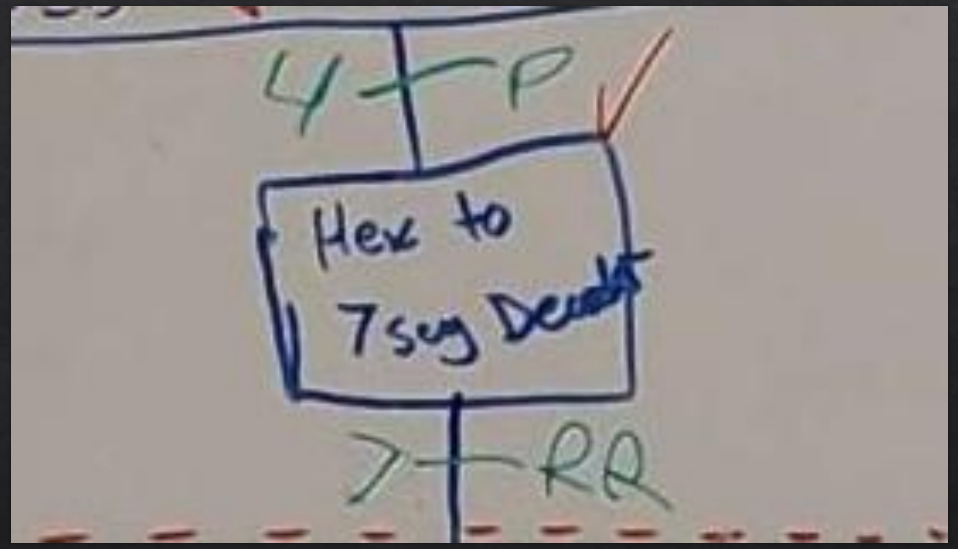
Multiplexor (Mux)

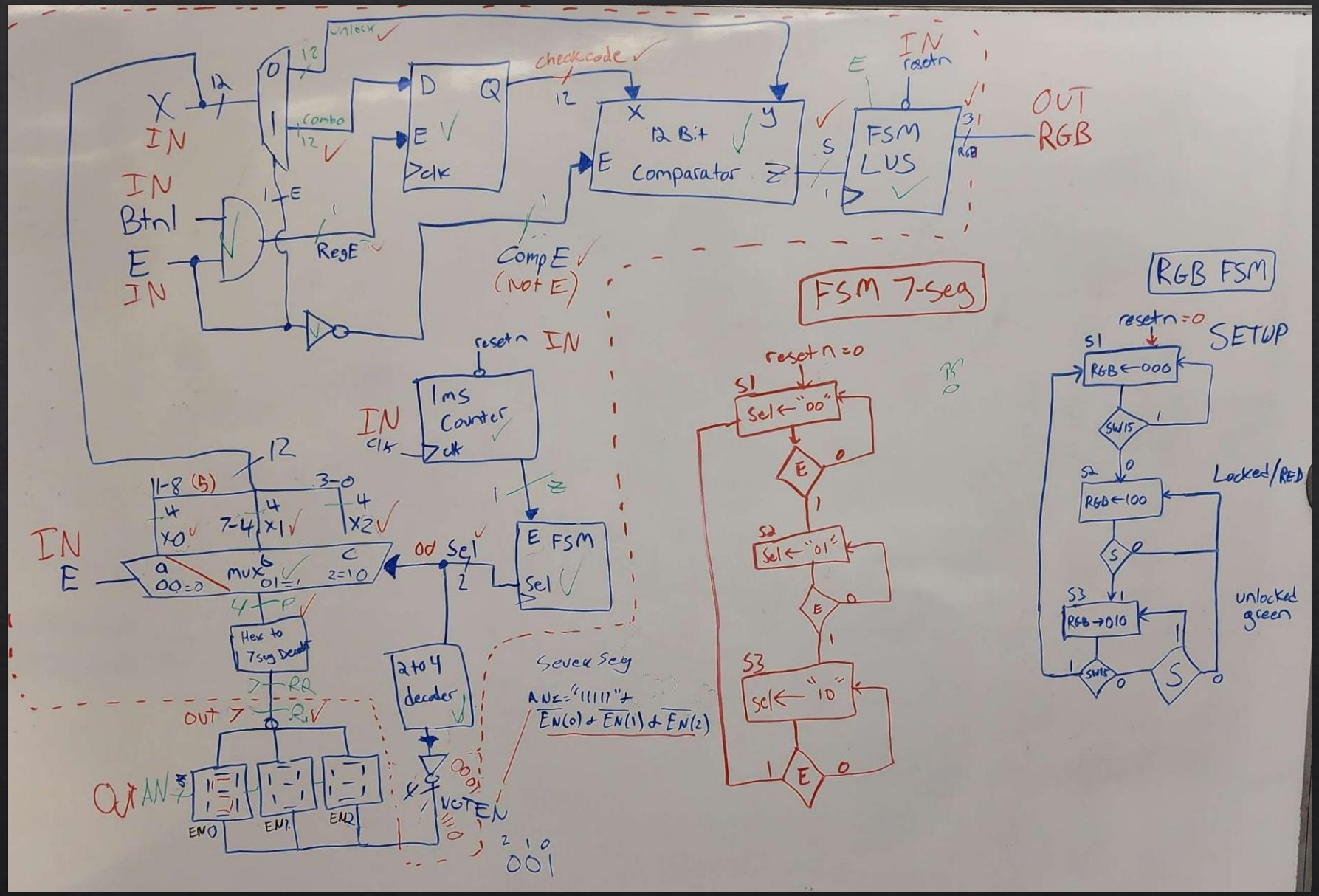
- 5 inputs: A, B, C, E, and Sel.
- A, B, and C are each 4 bits that are dependent on the input X from the 12 switches on the FPGA.
- E (Enable) and Sel (Select)
- There is one 4-bit output P



Hex to 7 Segment Decoder

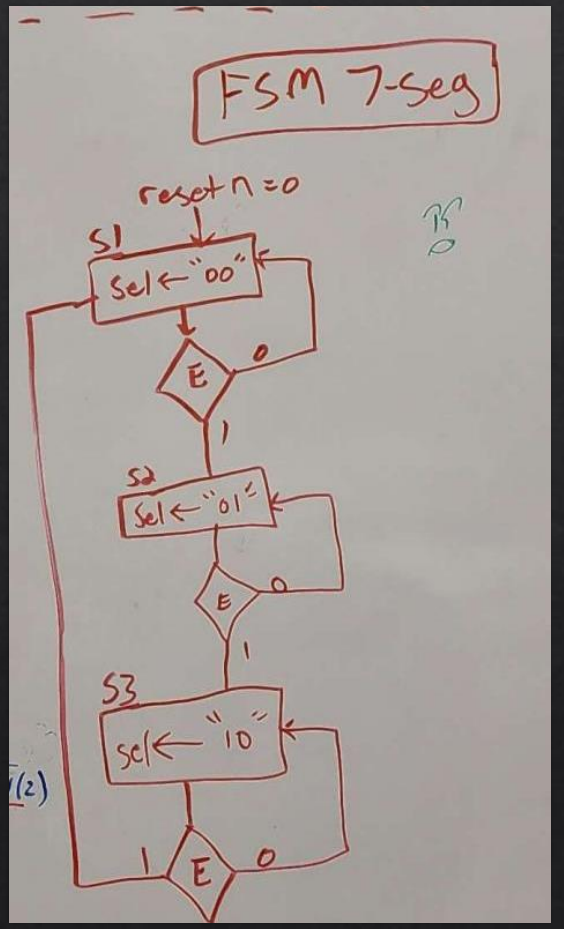
- The Hex 7Seg Decoder has one input P and one output RR
- It takes the value given to it by the Mux and converts it to a 7-bit output that can will display on the seven segment displays




unlock
checkcode
combo
X
IN
Btnl
E
IN
RegE
D
Q
E
clk
12 Bit
Comparator
x
y
z
E
IN
resetn
S
FSM
LUS
31
RGB
OUT
RGB
CompE
(Not E)
FSM 7-seg
RGB FSM
resetn IN
1ms
Counter
clk
IN
clk
11-8 (5)
7-4
3-0
X0
X1
X2
IN
E
mux
a
00=0
b
01=1
c
2=10
00 Sel
E FSM
Sel
Hex to
7seg Decoder
RR
out 7
RV
2 to 4
decoder
AN
EN0
EN1
EN2
VCC
EN
Seven Seg
AN="1111"+
EN(0) & EN(1) & EN(2)
2 1 0
0 0 1
resetn=0
S1
Sel←"00"
E
S2
Sel←"01"
S3
Sel←"10"
resetn=0
SETUP
S1
RGB←000
SW15
S2
RGB←100
Locked/RED
S
S3
RGB→010
unlocked
green
SW15
S

Seven Segment FSM

- 3 State FSM
  - “Sel” 2-bit value tells the 3to1Mux which seven segment display to use
  - Enable value transitions between each states
- Transitions : "y" (controlled by resetn, clock, E)
- Outputs : "sel" (controlled by y, E)
- This FSM controls the MUX and the Decoder at the same time. This allows the 7 segment display to show different values on each of the displays

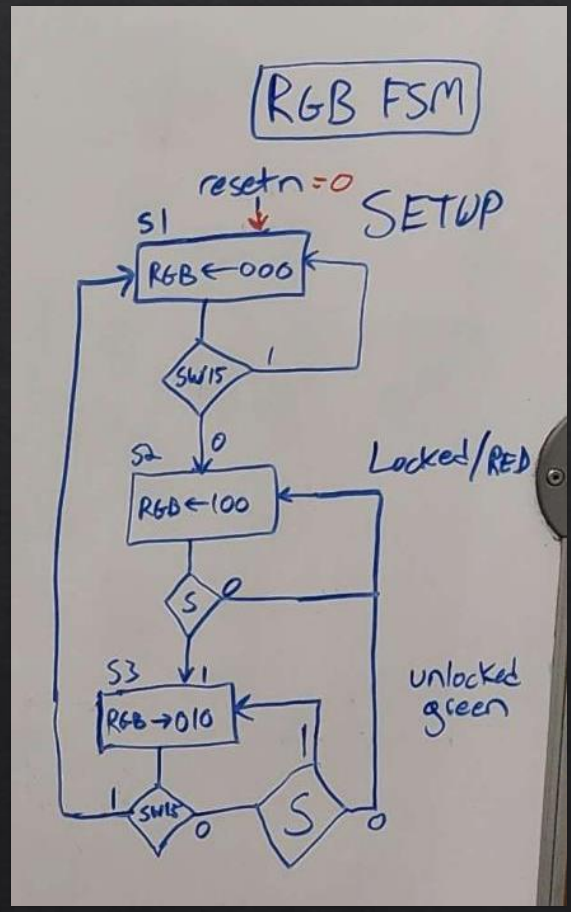


RGB LED FSM

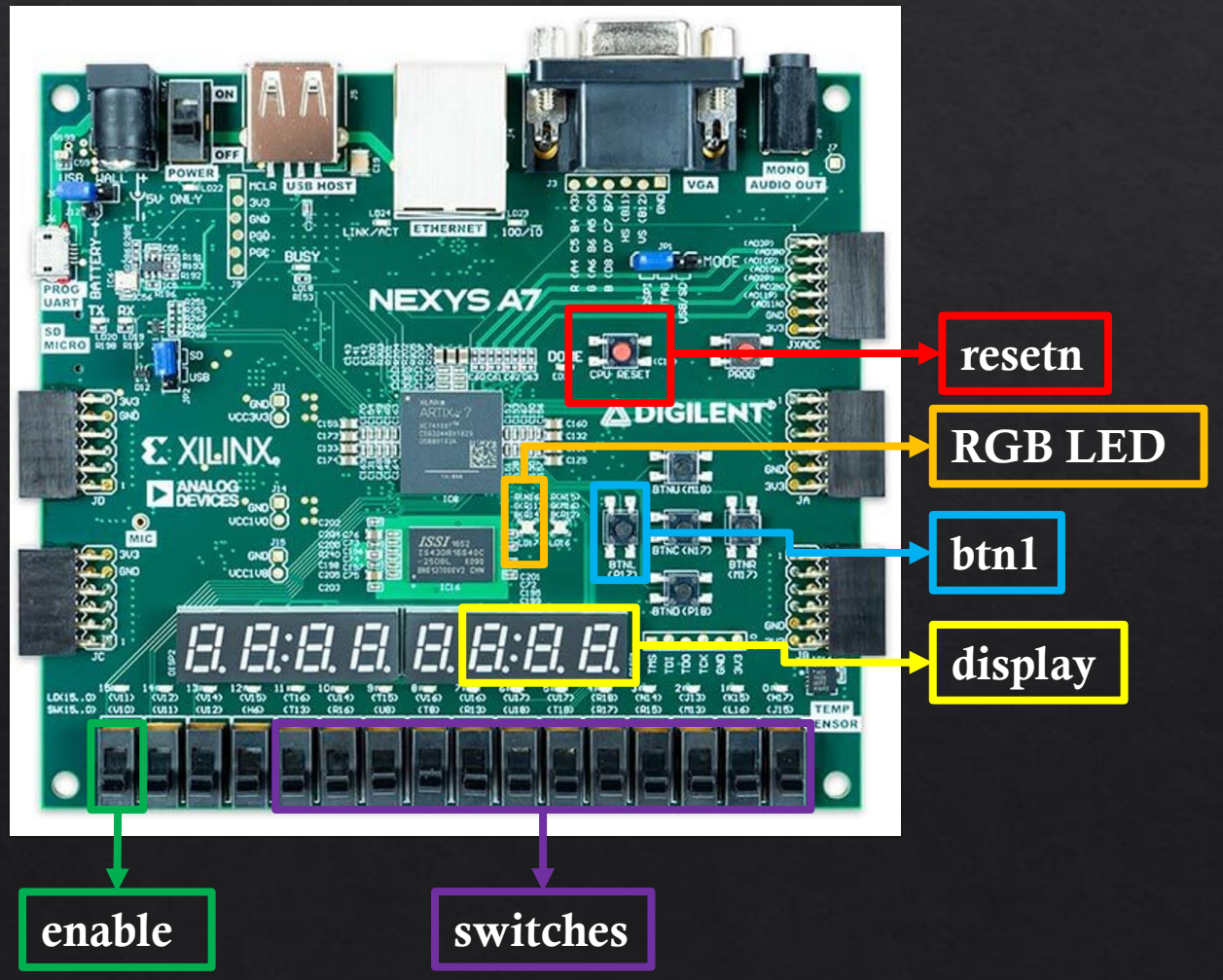
- 3 State FSM
  - Setup
  - Locked
  - Unlocked
- RGB Color definitions

R	G	B
0	0	0
1	0	0
0	1	0

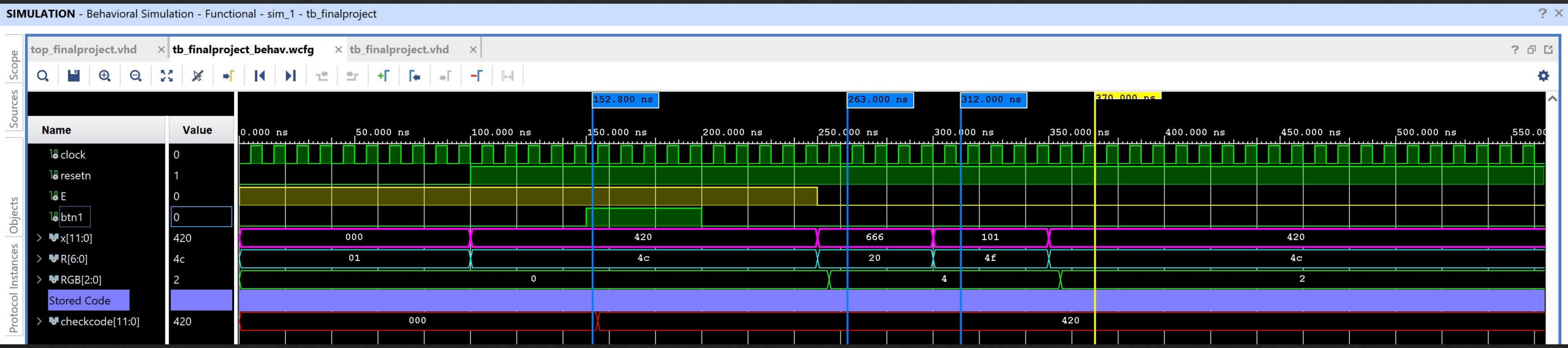
  - Off
  - Red (Incorrect combination)
  - Green (Correct combination)
- Transitions : y (controlled by resetn, clock, s, SW15)
- Outputs : RGB (controlled by y, s, SW15)



Constraints



Simulation



- User sets combination to 420
- Attempted combinations 666 and 101
  - RGB LED is still red (4)
- User attempts combination of 420
  - RGB LED becomes green (2)

Video Demonstration

Thank You!
Questions?