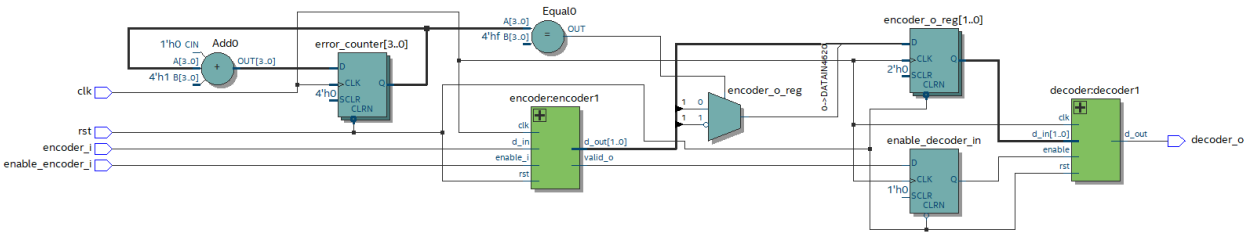
Term Project ECE - 111

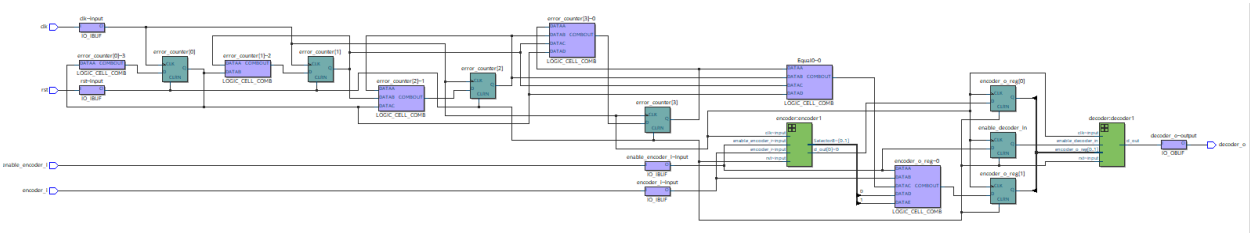
Viterbi Decoder

1.

a. RTL schematic



b. Post mapping schematic



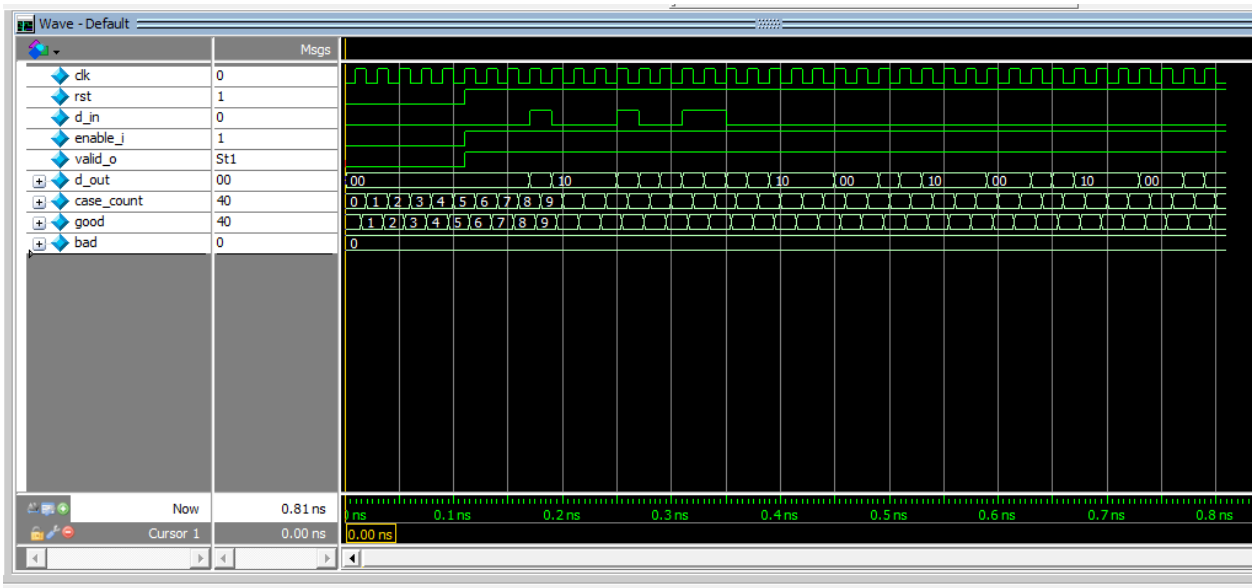
c. Resource usage

	Resource	Usage
1	ALUTs Used	391 / 36,100 (1 %)
1	-- Combinational ALUTs	391 / 36,100 (1 %)
2	-- Memory ALUTs	0 / 18,050 (0 %)
3	-- LUT_REGs	0 / 36,100 (0 %)
2	Dedicated logic registers	255 / 36,100 (< 1 %)
3		
4	Combinational ALUT usage by number of inputs	
1	-- 7 input functions	8
2	-- 6 input functions	73
3	-- 5 input functions	16
4	-- 4 input functions	28
5	-- <=3 input functions	266
5		
6	Combinational ALUTs by mode	
1	-- normal mode	215
2	-- extended LUT mode	8
3	-- arithmetic mode	168
4	-- shared arithmetic mode	0
7		
8	Logic utilization	516 / 36,100 (1 %)
1	-- Difficulty Clustering Design	Low
2	-- Combinational ALUT/register pairs used in final Placement	481
1	-- Combinational with no register	226
2	-- Register only	90
3	-- Combinational with a register	165
3	-- Estimated pairs recoverable by pairing ALUTs and registers as design grows	-38
4	-- Estimated Combinational ALUT/register pairs unavailable	73
1	-- Unavailable due to Memory LAB use	0
2	-- Unavailable due to unpartnered 7 LUTs	8

* Register count does not include registers inside RAM blocks or DSP blocks

d. Simulation
(ZOOM IN TO SEE DATA)

ENCODER

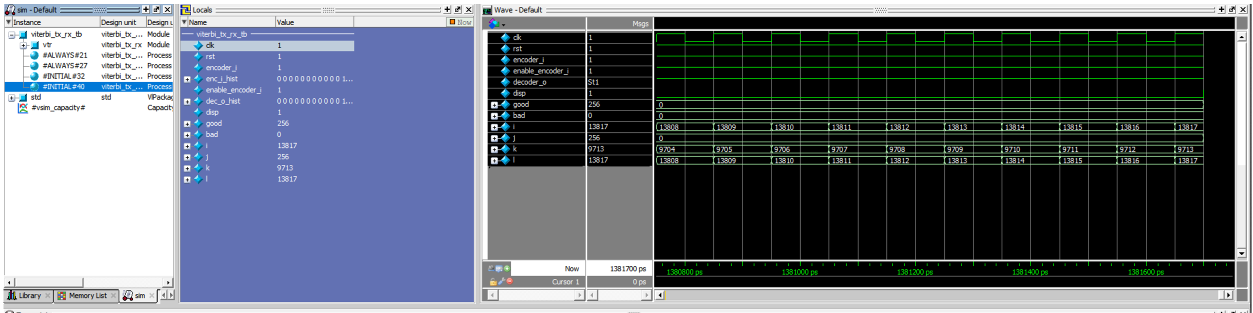


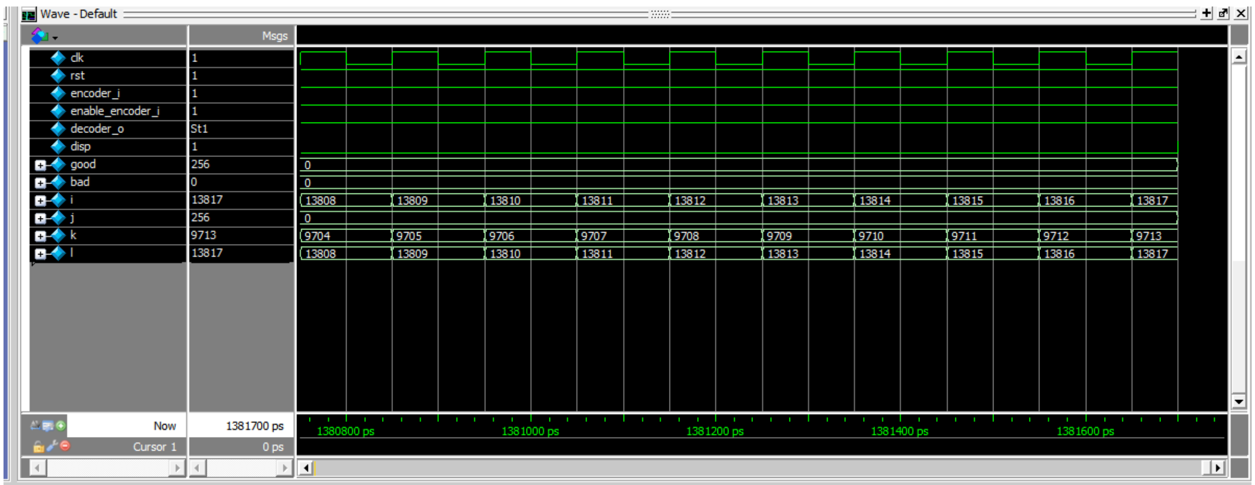

```
# yaa    0    10   10
# yaa    0    10   10
# yaa    0    00   00
# yaa    0    00   00
# yaa    0    10   10
# yaa    0    00   00
# yaa    0    10   10
# yaa    0    10   10
# yaa    0    10   10
# yaa    0    00   00
# yaa    0    00   00
# yaa    0    10   10
# yaa    0    00   00
# yaa    0    10   10
# yaa    0    10   10
# yaa    0    10   10
# yaa    0    00   00
# yaa    0    00   00
# yaa    0    10   10
# yaa    0    00   00
# good =           40, bad =           0
# ** Note: $stop    : C:/Users/Venz/Downloads/encoder/encoder/encoder_tb.sv(51)
#    Time: 810 ps  Iteration: 0  Instance: /encoder_tb
```

Explanation (Encoder) -

Our Encoder is designed by using 8 case states. Depending on each current state a certain value is assigned to the next state concurrent with the state table for the encoder. As well as a d_out_reg. Then on the next clk or rst a value is assigned to current state. Either the next state for 3'b000.

DECODER






```
# yaa! in = 1, out = 1
# yaa! in = 1, out = 1
# yaa! in = 1, out = 1
# yaa! in = 1, out = 1
# yaa! in = 1, out = 1
# yaa! in = 1, out = 1
# yaa! in = 1, out = 1
# yaa! in = 1, out = 1
# yaa! in = 1, out = 1
# yaa! in = 1, out = 1
# yaa! in = 1, out = 1
# yaa! in = 1, out = 1
# yaa! in = 1, out = 1
# yaa! in = 1, out = 1
# yaa! in = 1, out = 1
# yaa! in = 1, out = 1
# yaa! in = 1, out = 1
# yaa! in = 1, out = 1
# yaa! in = 1, out = 1
# yaa! in = 1, out = 1
# good =         256, bad =           0
# ** Note: $stop    : C:/Users/Venz/Desktop/ECE 111/HW8/vitassign/viterbi_tx_rx_tb.sv(371)
#    Time: 1381700 ps  Iteration: 0  Instance: /viterbi_tx_rx_tb
# Break in Module viterbi_tx_rx_tb at C:/Users/Venz/Desktop/ECE 111/HW8/vitassign/viterbi_tx_rx_tb.sv line 371
```

Explanation (Decoder)-

Here we can see that our viterbi decoder was able to get 0 bad bits showcasing that our code works perfectly. Our expected values were correct.

e. **Verilog Design Code**

Github Repository - https://github.com/Vnz01/ECE-111-HW-Assignments.git