Giaan Nguyen

Lab 6: Viterbi Decoder on FPGAs

ECEN 689-600: FPGA Information Processing Systems
Wednesday, March 24, 2021

Simulation

Figure 1 shows the results for simulating the Viterbi decoder, given an input bitstream of 1111010001. We can see that the output code is 10110. When computing the code by hand using a trellis diagram as seen in Figure 2, we see that the codes in both figures match, confirming that the design works.

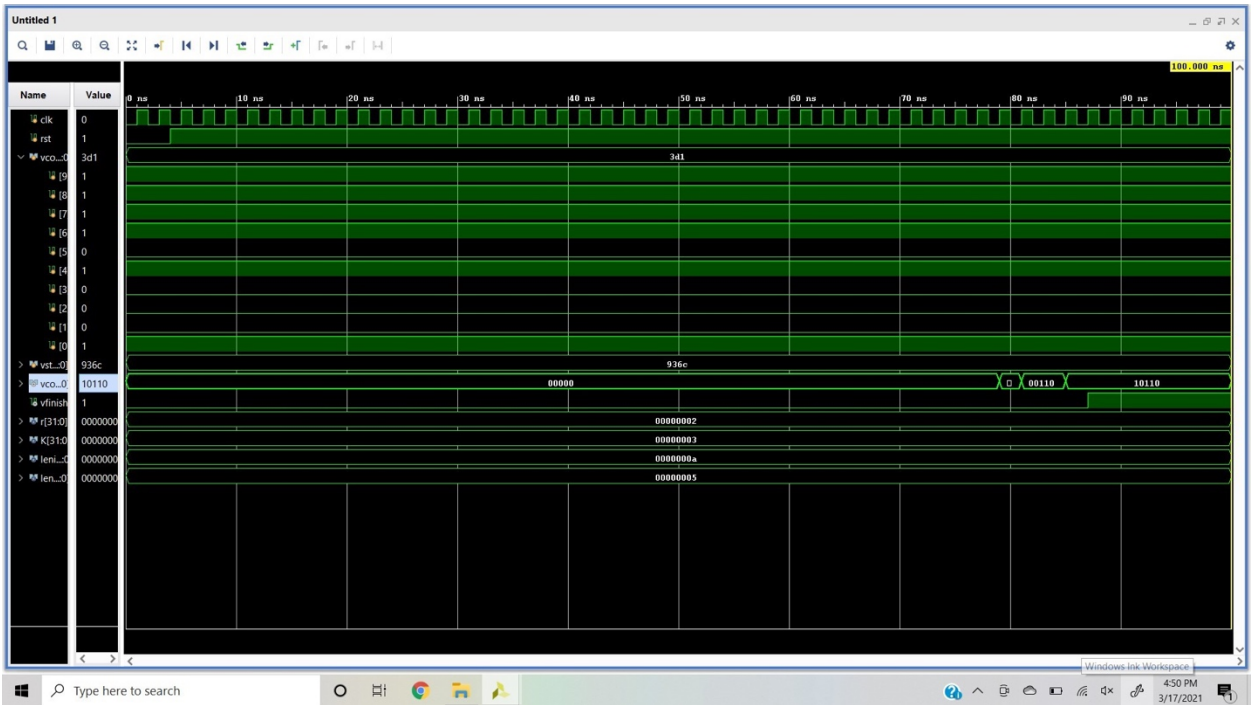


Figure 1. Simulation for the Viterbi decoder.

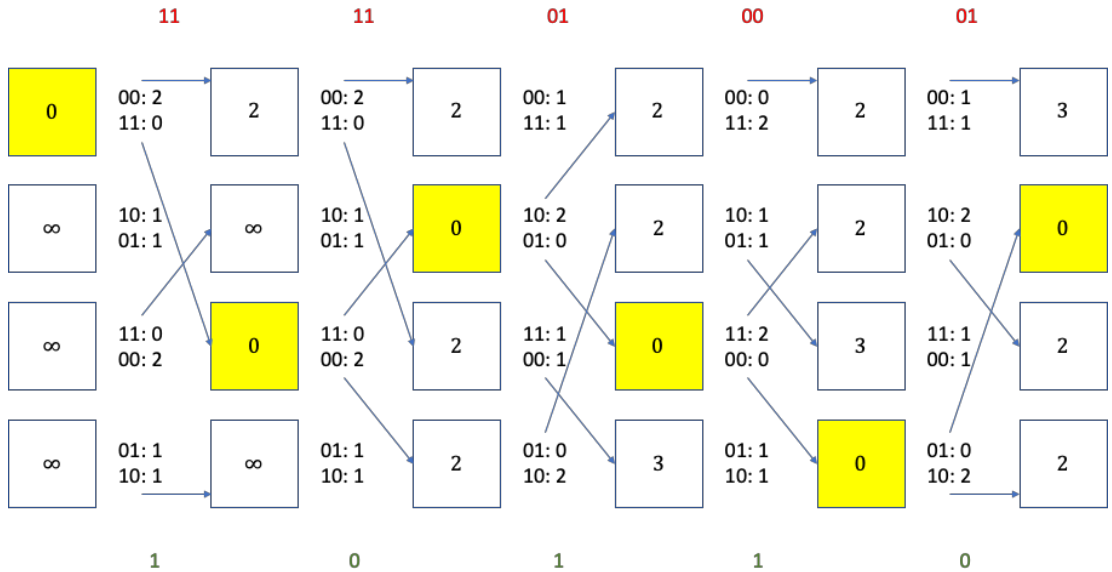


Figure 2. Trellis diagram for simulation. The top red values represent the bitstream, inputted two bits at a time. The bottom green values represent the decoded values corresponding to the minimal path highlighted in yellow. The four rows represent states 00, 01, 10, and 11 in consecutive order.

FPGA Outputs

The terminal outputs for the Viterbi decoder can be seen in Figure 3. We can observe that for all three test inputs, we get the same output code as the printed correct code, which means our design works; additionally, each test case undergoes 251 clock cycles.

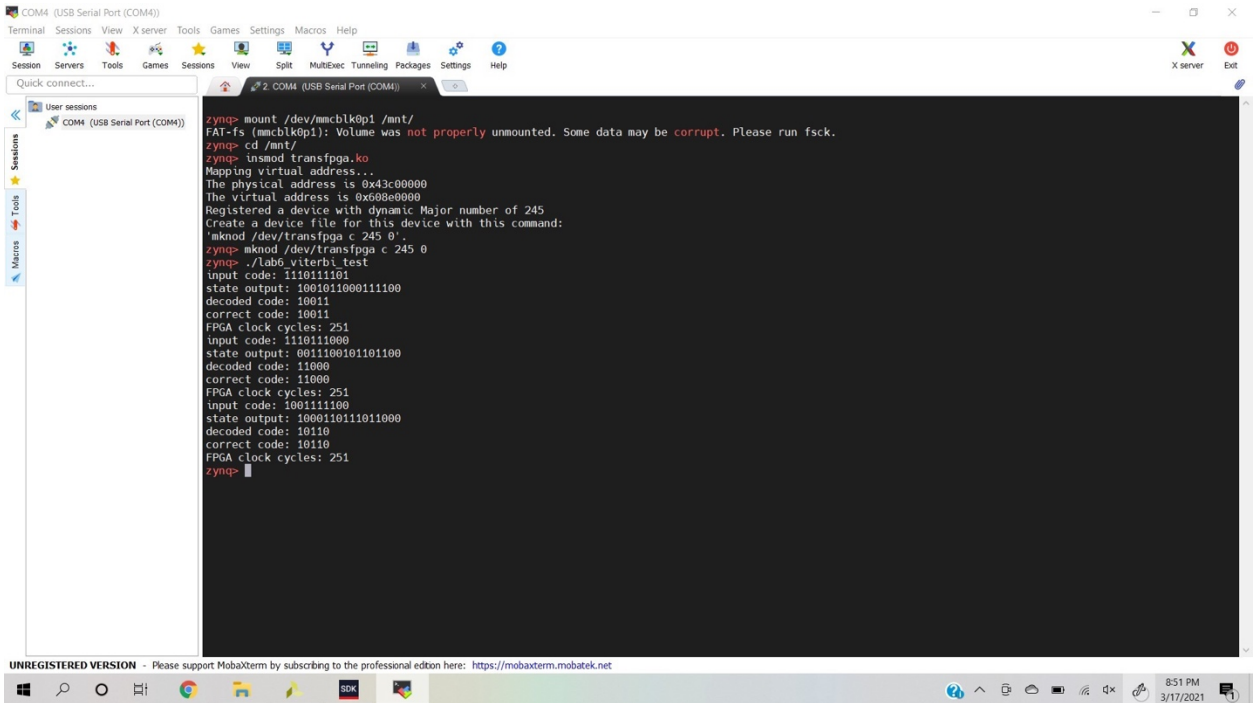
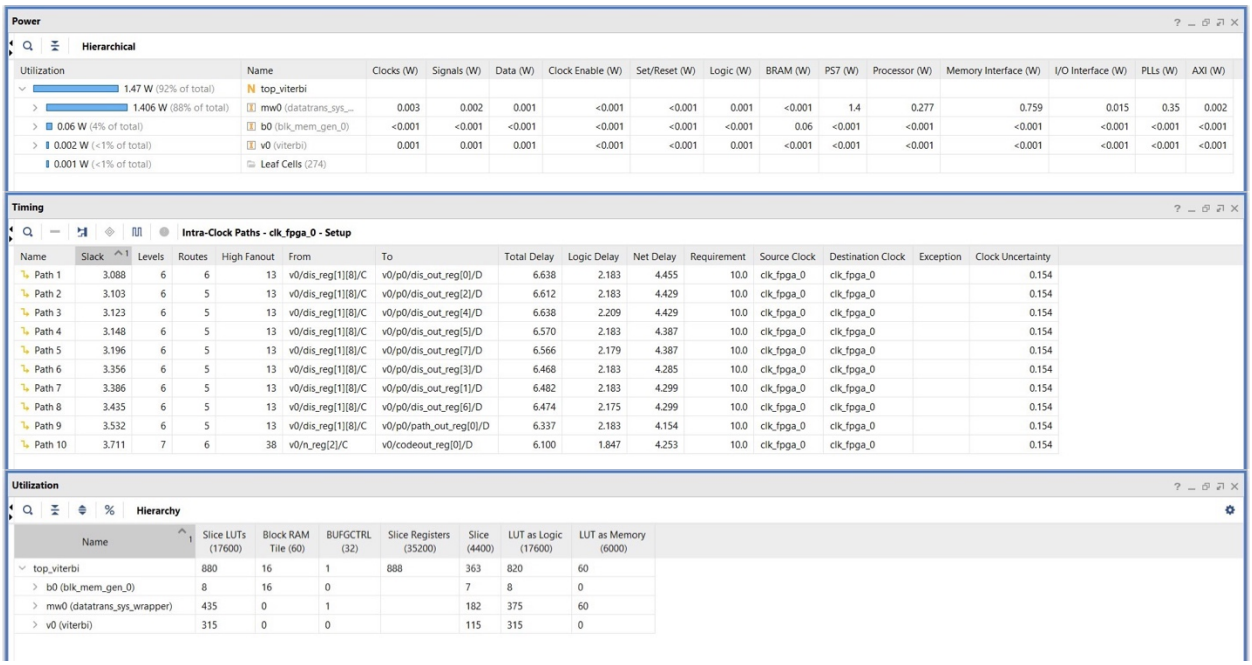


Figure 3. FPGA output for the Viterbi decoder, with the terminal displaying input codes and their respective decoded codes.

Implementation Summary

Figure 4 shows the power, timing, and utilization summaries for the implementation of the Viterbi decoder.



Power

Hierarchical

Utilization	Name	Clocks (W)	Signals (W)	Data (W)	Clock Enable (W)	Set/Reset (W)	Logic (W)	BRAM (W)	PS7 (W)	Processor (W)	Memory Interface (W)	I/O Interface (W)	PLLs (W)	AXI (W)
1.47 W (92% of total)	top_viterbi													
1.406 W (88% of total)	mw0 (datatrans_sys_...	0.003	0.002	0.001	<0.001	<0.001	0.001	<0.001	1.4	0.277	0.759	0.015	0.35	0.002
0.06 W (4% of total)	b0 (blk_mem_gen_0)	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001	0.06	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001
0.002 W (<1% of total)	v0 (viterbi)	0.001	0.001	0.001	<0.001	<0.001	0.001	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001
0.001 W (<1% of total)	Leaf Cells (274)													

Timing

Intra-Clock Paths - clk_fpga_0 - Setup

Name	Slack	Levels	Routes	High Fanout	From	To	Total Delay	Logic Delay	Net Delay	Requirement	Source Clock	Destination Clock	Exception	Clock Uncertainty
Path 1	3.088	6	6	13	v0/dis_reg[1][8]/C	v0/p0/dis_out_reg[0]/D	6.638	2.183	4.455	10.0	clk_fpga_0	clk_fpga_0		0.154
Path 2	3.103	6	5	13	v0/dis_reg[1][8]/C	v0/p0/dis_out_reg[2]/D	6.612	2.183	4.429	10.0	clk_fpga_0	clk_fpga_0		0.154
Path 3	3.123	6	5	13	v0/dis_reg[1][8]/C	v0/p0/dis_out_reg[4]/D	6.638	2.209	4.429	10.0	clk_fpga_0	clk_fpga_0		0.154
Path 4	3.148	6	5	13	v0/dis_reg[1][8]/C	v0/p0/dis_out_reg[5]/D	6.570	2.183	4.387	10.0	clk_fpga_0	clk_fpga_0		0.154
Path 5	3.196	6	5	13	v0/dis_reg[1][8]/C	v0/p0/dis_out_reg[7]/D	6.566	2.179	4.387	10.0	clk_fpga_0	clk_fpga_0		0.154
Path 6	3.356	6	5	13	v0/dis_reg[1][8]/C	v0/p0/dis_out_reg[3]/D	6.468	2.183	4.285	10.0	clk_fpga_0	clk_fpga_0		0.154
Path 7	3.386	6	5	13	v0/dis_reg[1][8]/C	v0/p0/dis_out_reg[1]/D	6.482	2.183	4.299	10.0	clk_fpga_0	clk_fpga_0		0.154
Path 8	3.435	6	5	13	v0/dis_reg[1][8]/C	v0/p0/dis_out_reg[6]/D	6.474	2.175	4.299	10.0	clk_fpga_0	clk_fpga_0		0.154
Path 9	3.532	6	5	13	v0/dis_reg[1][8]/C	v0/p0/path_out_reg[0]/D	6.337	2.183	4.154	10.0	clk_fpga_0	clk_fpga_0		0.154
Path 10	3.711	7	6	38	v0/n_reg[2]/C	v0/codeout_reg[0]/D	6.100	1.847	4.253	10.0	clk_fpga_0	clk_fpga_0		0.154

Utilization

Hierarchy

Name	Slice LUTs (17600)	Block RAM Tile (60)	BUFGCTRL (32)	Slice Registers (35200)	Slice (4400)	LUT as Logic (17600)	LUT as Memory (6000)
top_viterbi	880	16	1	888	363	820	60
b0 (blk_mem_gen_0)	8	16	0		7	8	0
mw0 (datatrans_sys_wrapper)	435	0	1		182	375	60
v0 (viterbi)	315	0	0		115	315	0

Figure 4. Implementation summary for the Viterbi decoder: power (top), timing (middle), and utilization (bottom).

Questions

1. **Is Viterbi decoder guaranteed to decode the original data correctly? Why?**

While the Viterbi decoder minimizes the probability of error, it does not necessarily remove error completely; that is, the Viterbi decoder does not always decode correctly. For instance, consider a noisy analog signal value of 0.49 versus 0.51. When applying hard-decision decoding, digitization will truncate 0.49 to 0 and round 0.51 to 1; while 0.49 and 0.51 are close in value, this quantization error would not be picked up by the decoder under hard-decision decoding.
Additionally, there may be times when branch metrics for multiple states at a certain time instance are equal and minimal; then the problem introduces multiple different paths to take. Naturally, with more samples, a rule for tiebreaking can be formulated, but the initial tiebreaks may give error. That said, the Viterbi errors are usually minimal (but not completely zero).