Digital Circuit Design Lab2 Report

1. Half Subtractor

(1) Truth Table

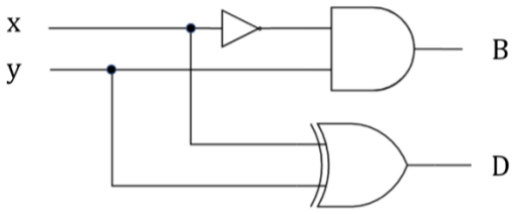
Input		Output	
x	y	B	D
0	0	0	0
0	1	1	1
1	0	0	1
1	1	0	0

(2) Boolean Expressions

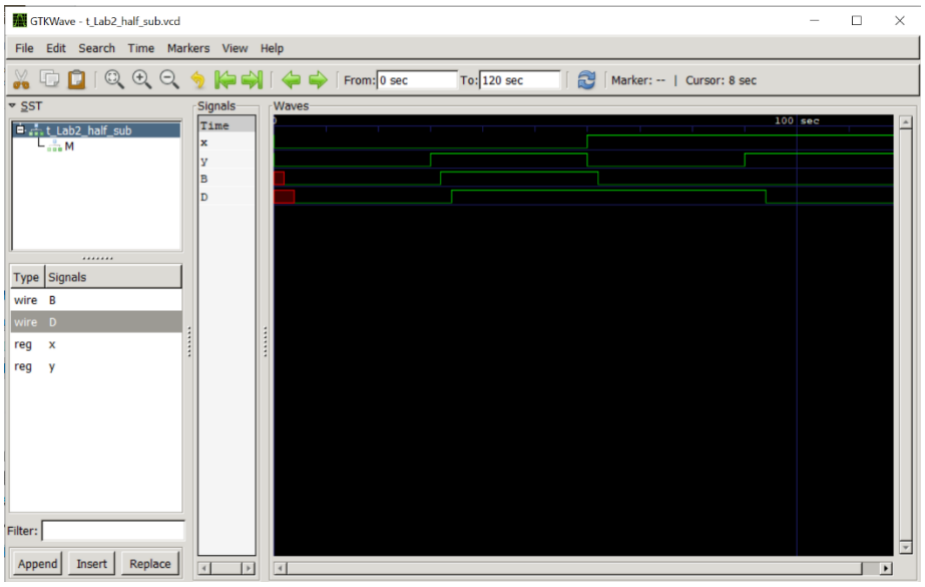
$B = x' y$

$D = x' y + x y' = x \oplus y$

(3) Logic Diagram



(4) Waveform



Derived Truth Table

Input		Output	
x	y	B	D
0	0	0	0
0	1	1	1
1	0	0	1
1	1	0	0

(5) Propagation Delay of the Circuit:

The propagation delay of B is 2 time units, and the propagation delay of D is 4 time units. Therefore, B changes its values 2 time units after the input value change, and D changes 4 time units after the input value change.

(6) Correctness of the Waveform:

The waveform results are correct since the truth table derived from the waveform is equivalent to the truth table we derived at the formulation stage.

2. Full Subtractor

(1) Truth Table

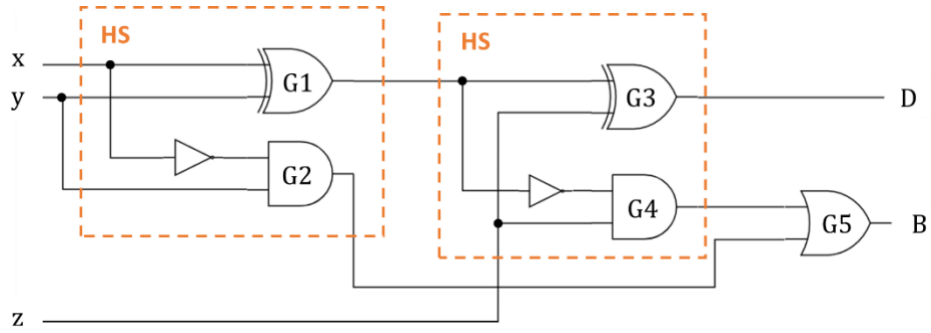
Input			Output	
x	y	z	B	D
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	1	0
1	0	0	0	1
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

(2) Boolean Expressions

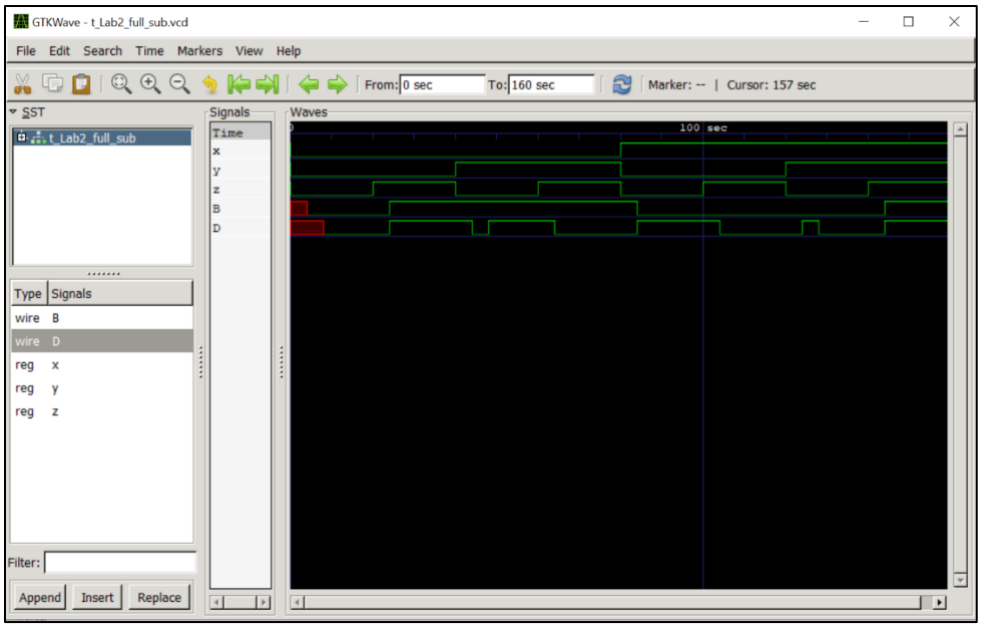
$B = x' y' z + x' y z' + x' y z + x y z$
$= z (x' y' + x y) + x' y$
$= z (x \oplus y)' + x' y$

$D = x \oplus y \oplus z$ (odd function)

(3) Logic Diagram



(4) Waveform



Derived Truth Table

Input			Output	
x	y	z	B	D
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	1	0
1	0	0	0	1
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

(5) Propagation Delay of the Circuit:

The propagation delay of D is 8 time units (G1 + G3). On the other hand, the propagation delay of B is 4 time units only (Max (G2, G4) + G5) since the initial value of y and z are 0, which generates an output of 1 regard less of the other input of G2 and G4. However, if the initial value isn't 0, the delay of B would be longer.

(6) Correctness of the Waveform:

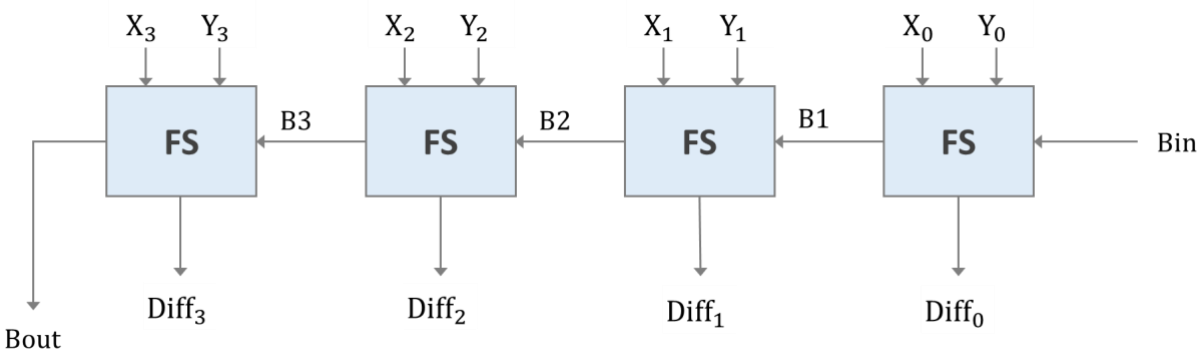
The waveform results are correct since the truth table derived from the waveform is equivalent to the truth table we derived at the formulation stage.

3. 4-bit Ripple Borrow Subtractor (RBS)

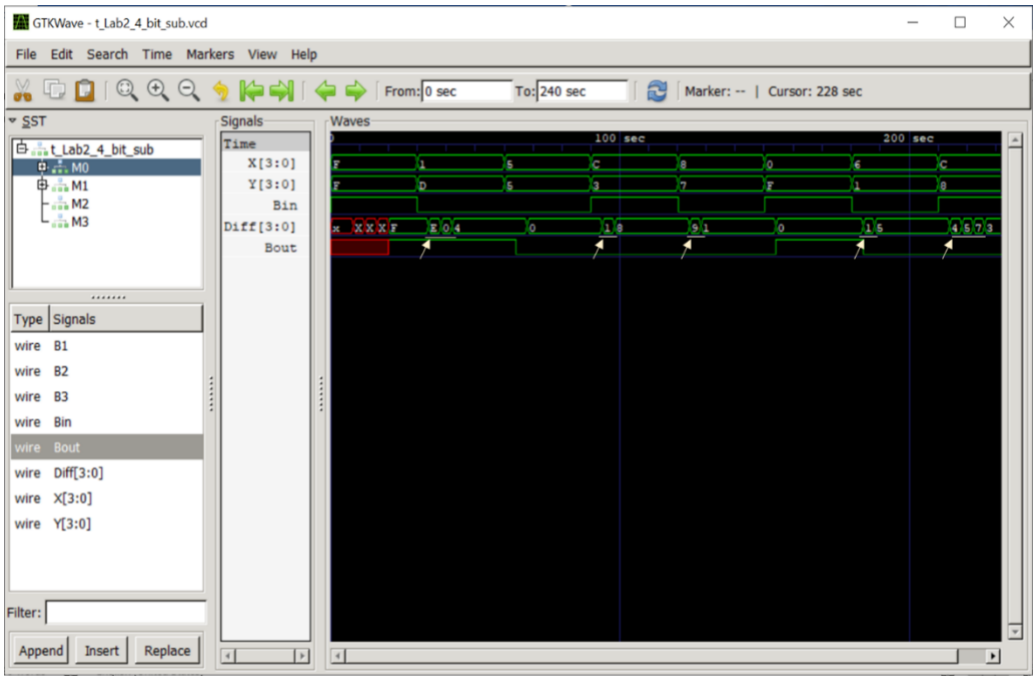
(1) Explanation

I used 4 full subtractors to perform subtraction bit by bit, and the borrow-out of the previous subtractor becomes the borrow-in of the next subtractor. The block diagram is below:

(2) Block Diagram



(3) Waveform



Derived Truth Table

Input			Output	
X	Y	Bin	Bout	Diff
F	F	1	1	F
1	D	0	1	4
5	5	0	0	0
C	3	1	0	8
8	7	0	0	1
0	F	1	1	0
6	1	0	0	5
C	8	1	0	3

(4) Propagation Delay of the Circuit

The propagation delay of Bout is 20 time units:
4 pairs of AND – OR gate and 1 XOR gate = $4 \times (2 + 2) + 4 = 20$

The propagation delay of Diff is 20 time units:
3 pairs of AND – OR gate and 2 XOR gate = $3 \times (2 + 2) + 2 \times 4 = 20$

(5) Correctness of the Waveform

The waveform is correct since Bout equals to 1 whenever X is smaller than (Y + Bin) which means that a borrow occurs. Also, the result of Diff is equivalent to (X – Y – Bin). However, since there are propagation delays, X, Y and Bin don't change their values at the same time, which generates different values for Diff at the beginning of the value change (see the pointed parts on the waveform).

4. 4-bit Borrow Lookahead Subtractor (BLS)

(1) Explanation

To construct a 4-bit BLS, we first define 8 new variables: P_i and G_i for $i = 0, 1, 2, 3$.

P_i : Borrow Propagate Function, which propagates the borrow from borrow-in to borrow-out = $(X_i \oplus Y_i)$'

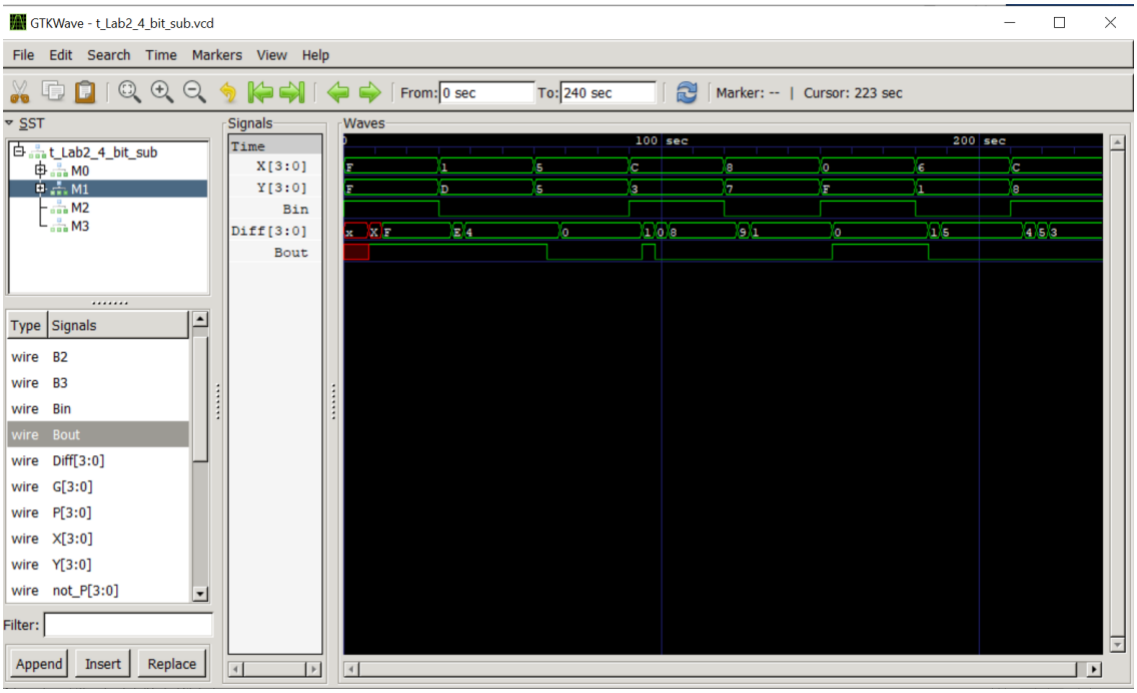
G_i : Borrow Generate Function, which generates a borrow-out regardless of the borrow-in = $X_i' Y_i$

$B_{i+1} = G_i + P_i B_i$

$D_i = P_i' \oplus B_i = (P_i \oplus B_i)'$

(2) Waveforms

Gate Level



Derived Truth Table

Input			Output	
X	Y	Bin	Bout	Diff
F	F	1	1	F
1	D	0	1	4
5	5	0	0	0
C	3	1	0	8
8	7	0	0	1
0	F	1	1	0
6	1	0	0	5
C	8	1	0	3

Propagation Delay of the Circuit:

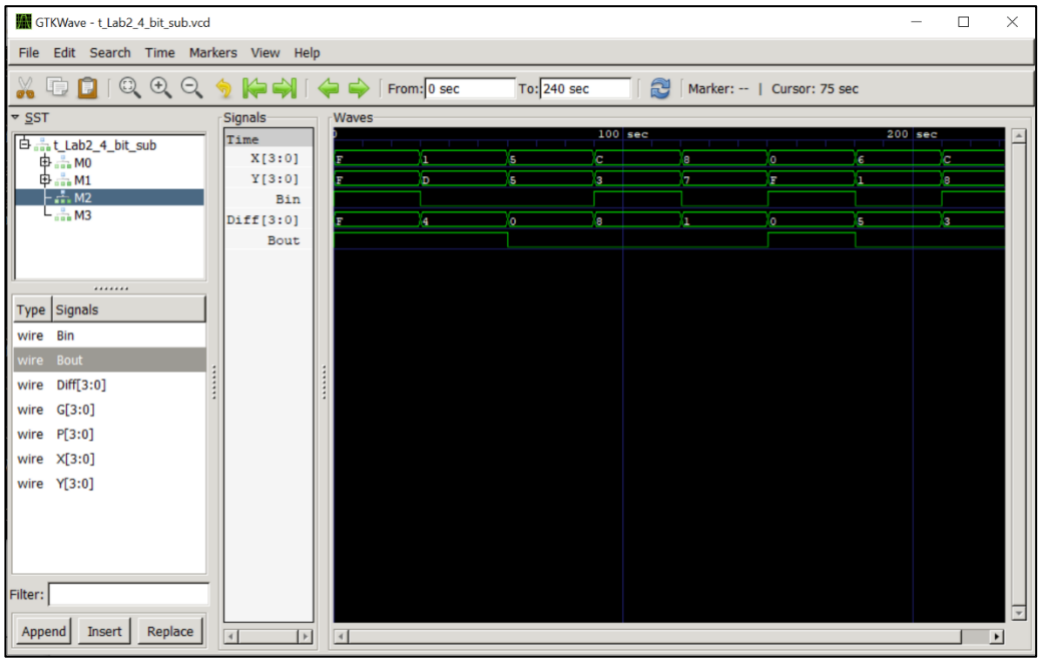
The propagation delay of Diff is 12 time units:
1 XNOR gate and 1 pair AND-OR gate and 1 XNOR gate = $4 + 2 \times 2 + 4 = 12$

The propagation delay of Bout is 8 time units:
1 XNOR gate and 1 pair of AND-OR gate = $4 + 2 \times 2 = 8$

Correctness of the waveform:

The waveform is correct since the derived truth table gives the correct value of Bout and Diff for all inputs. Similar to the previous question, there are propagation delays, X, Y and Bin don't change their values at the same time, which generates different values for Diff at the beginning of the value change.

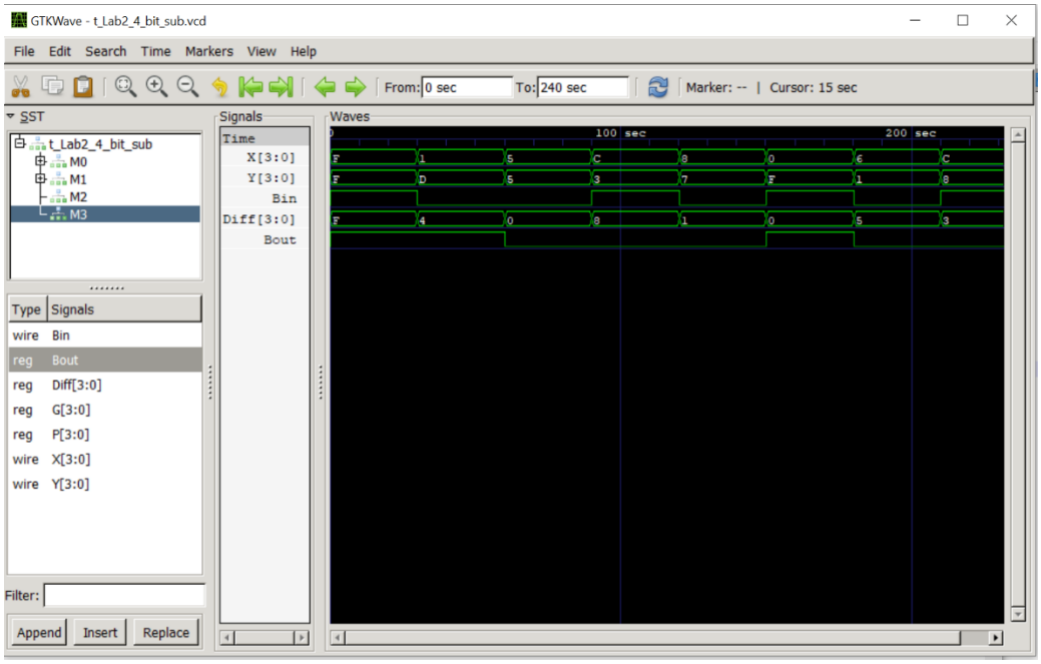
Dataflow



Derived Truth Table

Input			Output	
X	Y	Bin	Bout	Diff
F	F	1	1	F
1	D	0	1	4
5	5	0	0	0
C	3	1	0	8
8	7	0	0	1
0	F	1	1	0
6	1	0	0	5
C	8	1	0	3

Behavioral



Derived Truth Table

Input			Output	
X	Y	Bin	Bout	Diff
F	F	1	1	F
1	D	0	1	4
5	5	0	0	0
C	3	1	0	8
8	7	0	0	1
0	F	1	1	0
6	1	0	0	5
C	8	1	0	3

Correctness of the Waveforms:

The two waveforms above are all correct since the results of the derived truth table of Bout and Diff are all correct. Bout equals to 1 whenever X is smaller than (Y + Bin) which means that a borrow occurs. Also, the result of Diff is equivalent to (X – Y – Bin).

5. Design of Priority Encoder

(1) Truth Table

Input					Output			
D[0]	D[1]	D[2]	D[3]	D[4]	A[2]	A[1]	A[0]	V
0	0	0	0	0	X	x	x	0
1	x	x	x	x	0	0	0	1
0	1	x	x	X	0	0	1	1
0	0	1	x	x	0	1	0	1
0	0	0	1	X	0	1	1	1
0	0	0	0	1	1	0	0	1

(2) Derived Boolean Expressions

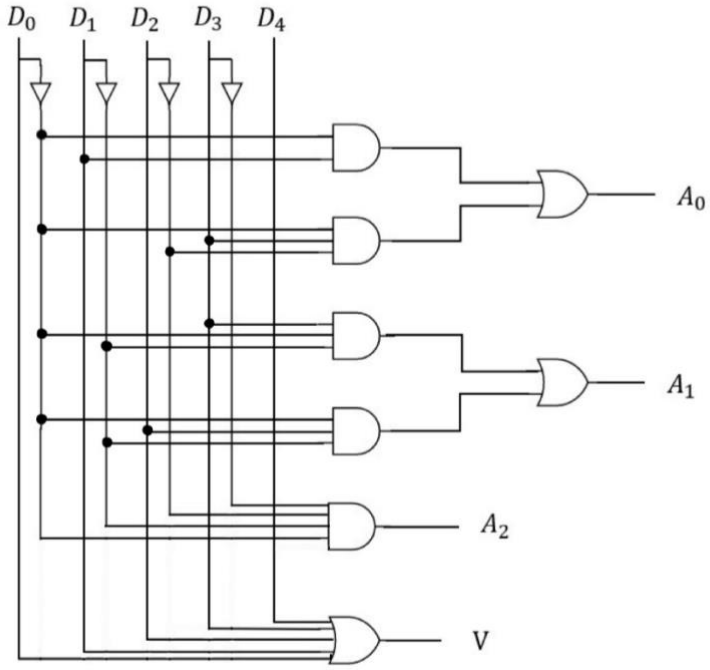
$$A[2] = D[0]'D[1]'D[2]'D[3]'D[4]$$
$$= D[0]'D[1]'D[2]'D[3]'$$

$$A[1] = D[0]'D[1]'D[2] + D[0]'D[1]'D[2]'D[3]$$
$$= D[0]'D[1]'D[2] + D[0]'D[1]'D[3]$$

$$A[0] = D[0]'D[1] + D[0]'D[1]'D[2]'D[3]$$
$$= D[0]'D[1] + D[0]'D[2]'D[3]$$

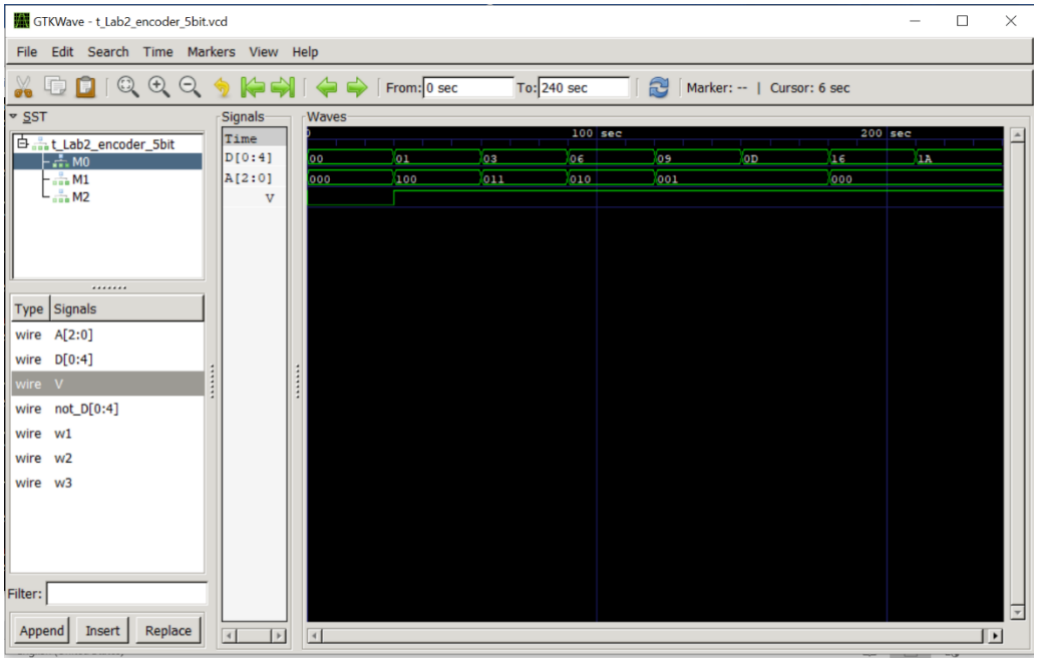
$$V = D[0] + D[1] + D[2] + D[3] + D[4]$$

(3) Block Diagram

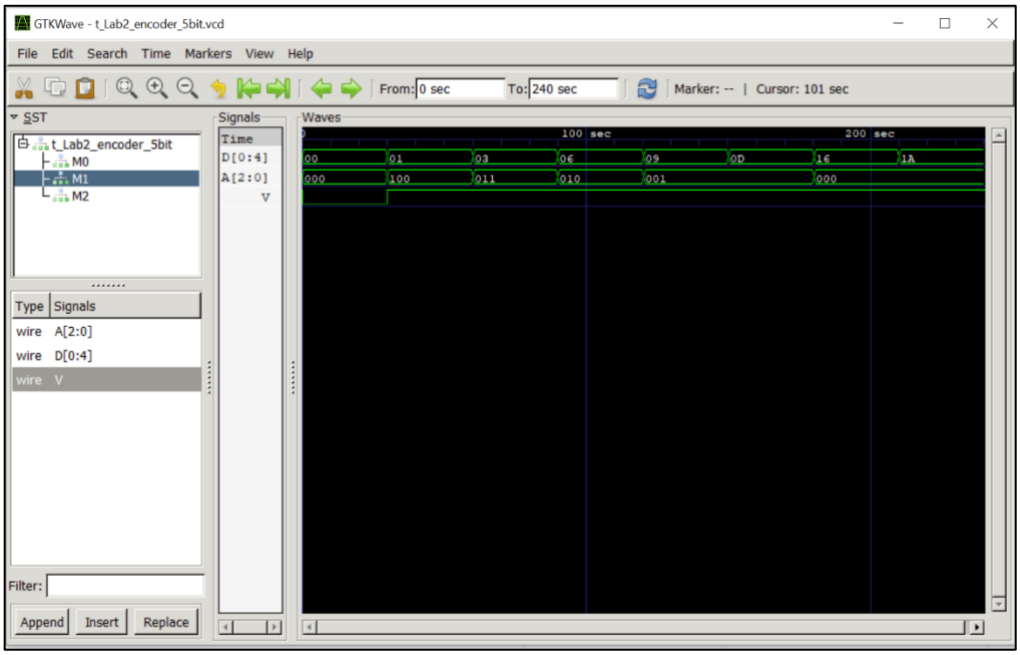


(4) Waveform

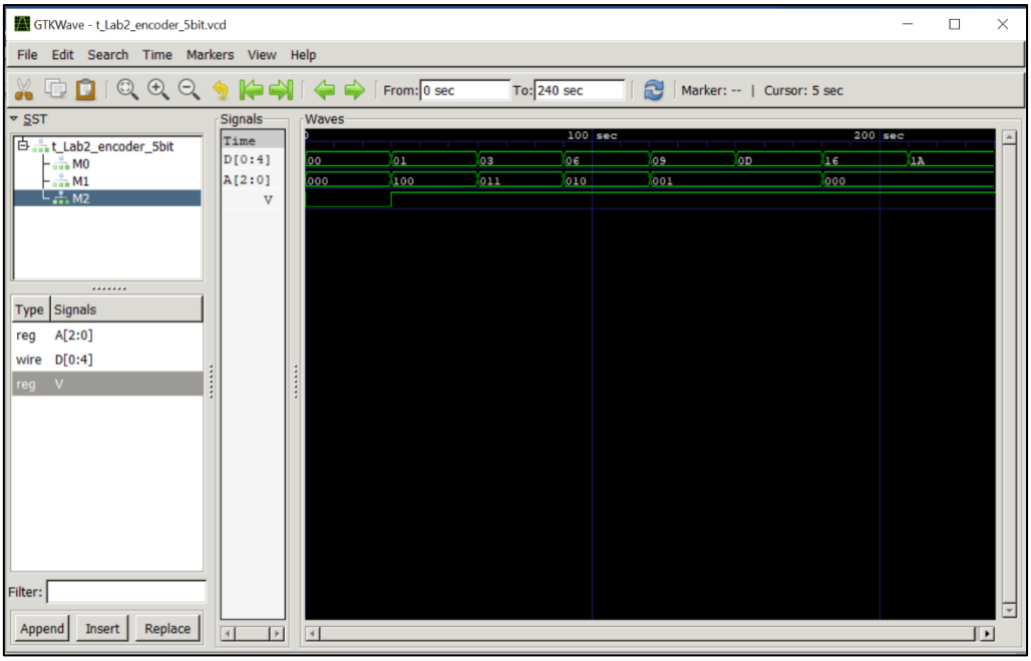
Gate Level



Dataflow



Behavioral



Derived Truth Table

Input					Output			
D[0]	D[1]	D[2]	D[3]	D[4]	A[2]	A[1]	A[0]	V
0	0	0	0	0	0	0	0	0
0	0	0	0	1	1	0	0	1
0	0	0	1	1	0	1	1	1
0	0	1	1	0	0	1	0	1
0	1	0	0	1	0	0	1	1
0	1	1	0	1	0	0	1	1
1	0	1	1	0	0	0	0	1
1	1	0	1	0	0	0	0	1

(5) Correctness of the waveforms:

The three waveforms are all correct since they derive the same truth table above. And the results are correct corresponding to the truth table given in the beginning.

6. Reflection

This assignment took me much more time to finish compared with Lab1 assignment. I spent a lot of time figuring out how to construct the digital circuits using Verilog code since I'm not familiar with the dataflow models and behavioral models. The most difficult part for me is examining the waveform results and check if it is correct because the form becomes more complicated with the propagation delays. Nevertheless, the overall experience is quite beneficial for me since I get to know more about the mechanisms behind combinational circuits as well as how they are built with HDL.