EE243 Project-2 Report

Boran Aybak Kılıç
2022401207

Overview of the Project and Methodology:

In this project we will design a digital clock. This clock will be in (hh.mm.ss) format and it will display each digit of the current second, minute, hour values using 7-segment display in BCD format. We are first required to design a 6-bit gray code counter and convert the outputs of the gray-code counter to 7-segment BCD format. We've already covered this part in the previous project, so the real challenge is to design the clock outputting numbers in the gray code format.
We will be given a 1Hz clock which sends positive signals every second. We need to utilize this clock to increment time while holding the state of each bit in memory to make the transition to the next state. In contrast with the first project where we had some inputs and a combinational circuit which manipulates these inputs to extract an output using Boolean algebra; in this project we need to use a sequential circuit in which outputs are dependent on the previous state of the outputs. To remember the previous state, we should utilize a common memory element known as flip-flop. In particular, we need to use D flip-flops which stores whatever data they're inputted in until the next data input.
We should also have a control input R which satisfies the following conditions:
- counter functions as described above if R=1 and returns to (00:00:00) if R=0.
- the counter is reset at the rising edge of the clock cycle. So, it's a synchronous reset.

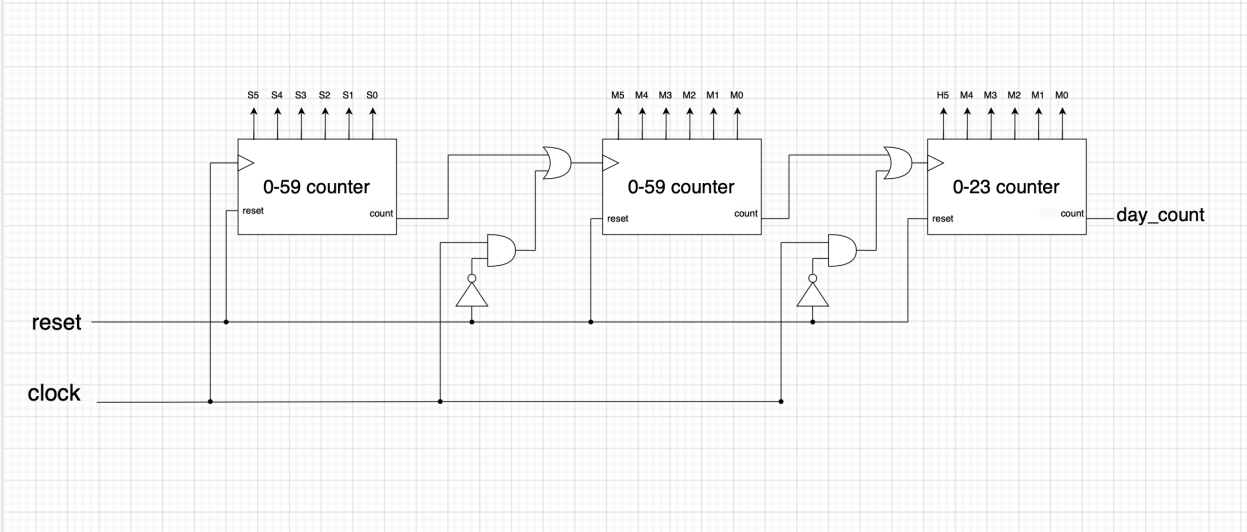
In order to reset each DFF synchronously with the clock we should somehow find a way of delivering the clock signal to each DFF. That means a ripple counter will not work in this case. Rather, we should design synchronous counters.

General Schematic:

Digital clock will have 3 segments:

- A 0-59 counter for counting seconds,
- A 0-59 counter for counting minutes,
- A 0-23 counter for counting hours.

Our reset input should go to each counters reset input uninterruptedly. However, since minute and hour counters need a clock signal for reset to be activated, we need to deliver a clock signal to them whenever reset is 0. We can do that by connecting clock input and inverted reset input by an AND gate. Moreover, for minute and hour counters to be updated. We should give them a clock signal whenever either second counter reaches 59 or minute counter reaches 59. Finally, should connect them by and OR gate. So, when either the prior counter reaches 59 or when reset signal is zero clock signal should be delivered to minute and hour counters. This is required because we are essentially designing a kind of ripple counter and in order to have a synchronous reset, we should deliver the clock signal to minute and hour counters.
You can see the general schematic below. Count output of second and minute counters are 1 only when output equals 60. *Day count output is only 1 when 24 hours have passed. This output can be used to design other projects such as a digital calendar. * This is a very nice option for modularity and scalability.



[1]

PART 1: Constructing the D flip-flops:

In this part we need to design a positive edge data flip-flop. Since our reset will be activated at the positive edge of the clock signal, it makes sense that our flip-flops are updated at the positive edges.

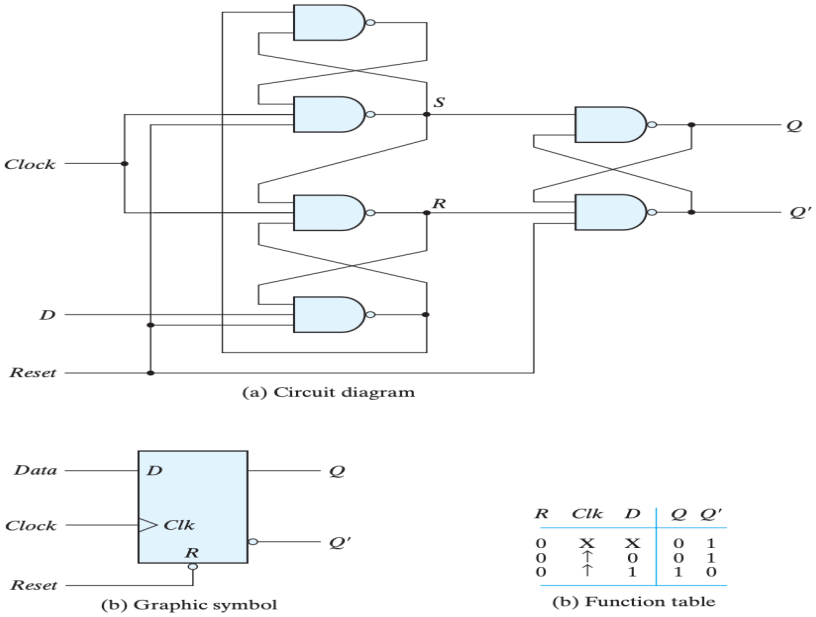
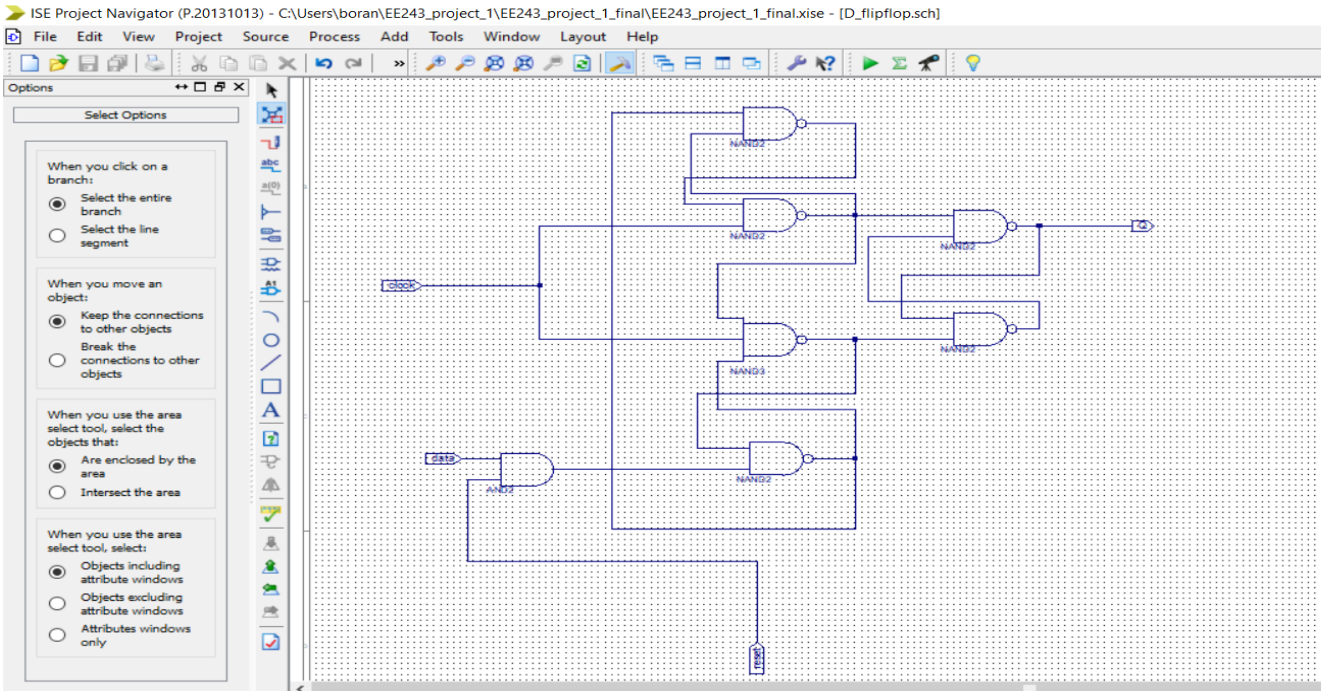


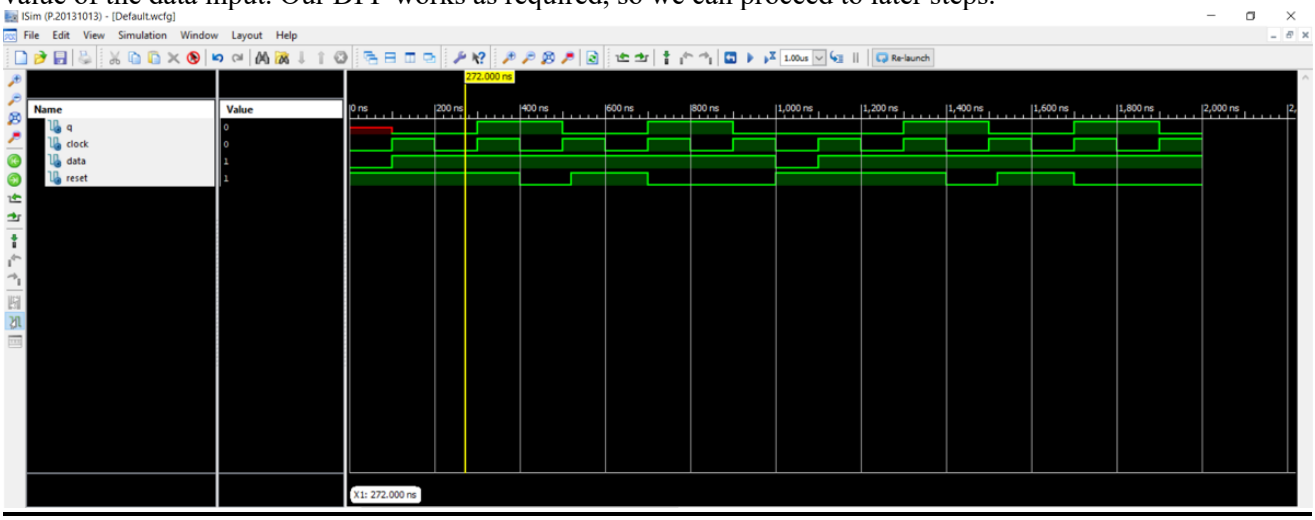
FIGURE 5.14
D flip-flop with asynchronous reset

[2]

For reset, we can alternatively connect data and reset inputs by an AND gate as in the schematic below. When reset=0, output Q=0 regardless of the value of data. When reset = 1, output equals to data input. Also in my design I have omitted the Q' output since it's just the complement of Q, we can obtain it by connecting Q with a NOT gate if required.



As you can see below if reset=1, q input gets updated at the rising edge of each clock signal based on the value of the data input. If reset=0, q output gets updated to 0 at the next positive edge regardless of the value of the data input. Our DFF works as required, so we can proceed to later steps.



PART 2: 0-59 and 0-23 Counters

Now for our 59 and 23 counters we need to design a sequential circuit. Each DFF output should give the next gray number based on the current state of the DFF. In order to design this circuit, we should create a state table as in [3]. Next state value corresponding to each present state is the next gray number on the count sequence. Each flip-flop input value is the next state value of the corresponding bit in significance order.

Present State						Next State						Flip-flop Inputs					
G5	G4	G3	G2	G1	G0	G5	G4	G3	G2	G1	G0	D5	D4	D3	D2	D1	D0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1
0	0	0	0	0	1	0	0	0	0	1	1	0	0	0	0	1	1
0	0	0	0	1	1	0	0	0	0	1	0	0	0	0	0	1	0
0	0	0	0	1	0	0	0	0	0	1	1	0	0	0	1	1	0
0	0	0	1	1	0	0	0	0	1	1	1	0	0	0	1	1	1
0	0	0	1	1	1	0	0	0	1	0	1	0	0	0	1	0	1
0	0	0	1	0	1	0	0	0	1	0	0	0	0	0	1	0	0
0	0	0	1	0	0	0	0	0	1	1	0	0	0	1	1	0	0
0	0	1	1	0	0	0	0	1	1	0	1	0	0	1	1	0	1
0	0	1	1	0	1	0	0	1	1	1	1	0	0	1	1	1	1
0	0	1	1	1	1	0	0	1	1	1	0	0	0	1	1	1	0
0	0	1	1	1	0	0	0	1	0	1	0	0	0	1	0	1	0
0	0	1	0	1	0	0	0	1	0	1	1	0	0	1	0	1	1
0	0	1	0	1	1	0	0	1	0	0	1	0	0	1	0	0	1
0	0	1	0	0	1	0	0	1	0	0	0	0	0	1	0	0	0
0	0	1	0	0	0	0	1	1	0	0	0	0	1	1	0	0	0
0	1	1	0	0	0	0	1	1	0	0	1	0	1	1	0	0	1
0	1	1	0	0	1	0	1	1	0	1	1	0	1	1	0	1	1
0	1	1	0	1	1	0	1	1	0	1	0	0	1	1	0	1	0
0	1	1	0	1	0	0	1	1	1	1	0	0	1	1	1	1	0
0	1	1	1	1	0	0	1	1	1	1	1	0	1	1	1	1	1
0	1	1	1	1	1	0	1	1	1	1	0	1	0	1	1	0	1
0	1	1	1	0	1	0	1	1	1	0	0	0	1	1	1	0	0
0	1	1	1	0	0	0	1	1	0	1	0	0	1	0	1	0	0
0	1	0	1	0	0	0	1	0	1	0	1	0	1	0	1	0	1
0	1	0	1	0	1	0	1	0	1	1	1	0	1	0	1	1	1
0	1	0	1	1	1	0	1	0	1	1	0	0	1	0	1	1	0
0	1	0	1	1	0	0	1	0	0	1	0	0	1	0	0	1	0
0	1	0	0	1	0	0	1	0	0	1	1	0	1	0	0	1	1
0	1	0	0	1	1	0	1	0	0	0	1	0	1	0	0	0	1
0	1	0	0	0	1	0	1	0	0	0	0	0	1	0	0	0	0
0	1	0	0	0	0	1	1	0	0	0	0	1	1	0	0	0	0
1	1	0	0	0	0	1	1	0	0	0	1	1	1	0	0	0	1
1	1	0	0	0	1	1	1	0	0	1	1	1	1	0	0	1	1
1	1	0	0	1	1	1	1	0	0	1	0	1	1	0	0	1	0
1	1	0	0	1	0	1	1	0	1	1	0	1	1	0	1	1	0
1	1	0	1	1	0	1	1	0	1	1	1	1	1	0	1	1	1
1	1	0	1	1	1	1	1	0	1	0	1	1	1	0	1	0	1
1	1	0	1	0	1	1	1	0	1	0	0	1	1	0	1	0	0
1	1	0	1	0	0	1	1	1	1	0	0	1	1	1	1	0	0
1	1	1	1	0	0	1	1	1	1	0	1	1	1	1	1	0	1
1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	0
1	1	1	1	1	0	1	1	1	1	0	1	1	1	1	1	0	1
1	1	1	1	0	1	1	1	1	1	0	1	1	1	1	1	0	1
1	1	1	0	1	1	1	1	1	1	0	0	1	1	1	1	0	1
1	1	1	0	0	1	1	1	1	1	0	0	0	1	1	1	0	0
1	1	1	0	0	0	1	0	1	1	0	0	1	0	1	1	0	0
1	0	1	0	0	0	1	0	1	0	0	1	1	0	1	0	0	1
1	0	1	0	0	1	1	0	1	0	1	1	1	0	1	0	1	1
1	0	1	0	1	1	1	0	1	0	1	0	1	0	1	0	1	0
1	0	1	0	1	0	1	0	1	1	1	0	1	0	1	1	1	0
1	0	1	1	1	0	1	0	1	1	1	1	1	0	1	1	1	1
1	0	1	1	1	1	1	0	1	1	0	1	1	0	1	1	0	1
1	0	1	1	0	1	1	0	1	1	0	0	1	0	1	1	0	0
1	0	1	1	0	0	1	0	0	1	0	0	1	0	0	1	0	0
1	0	0	1	0	0	1	0	0	1	0	1	1	0	0	1	0	1
1	0	0	1	0	1	1	0	0	1	1	1	1	0	0	1	1	1
1	0	0	1	1	1	1	0	0	1	1	0	1	0	0	1	1	0
1	0	0	1	1	0	1	0	0	0	1	0	1	0	0	0	1	0
1	0	0	0	1	0	1	0	0	0	0	1	1	0	0	0	1	1
1	0	0	0	1	1	1	0	0	0	0	1	1	0	0	0	0	1
1	0	0	0	0	1	1	0	0	0	0	0	1	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

I have created a 6-bit state table for each 64 number in gray code. Colors highlight the states when each of DFF's inputs are 1. Using the states when flip-flop inputs are 1, we should construct K-maps. We will need 6 flip-flops because we have 6 state variables: D5, D4, D3, D2, D1, D0 (D5 being the most significant and D0 being the least significant).

PART 2.1: Next State Circuitry

For constructing 6 K-maps each of which has 64 elements, I have used an online K-map solver tool. Here A corresponds to G5, and F corresponds to G0. All other outputs between are in correspondence with alphabetical order.

	$\overline{D}.\overline{E}.\overline{F}$	$\overline{D}.\overline{E}.F$	$\overline{D}.E.\overline{F}$	$\overline{D}.E.F$	$D.\overline{E}.\overline{F}$	$D.\overline{E}.F$	$D.E.\overline{F}$	$D.E.F$
$\overline{A}.\overline{B}.\overline{C}$	1	1	0	0	0	0	1	1
$\overline{A}.\overline{B}.C$	0	0	1	1	1	1	0	0
$\overline{A}.B.C$	1	1	0	0	0	0	1	1
$\overline{A}.B.\overline{C}$	0	0	1	1	1	1	0	0
$A.\overline{B}.\overline{C}$	0	0	1	1	1	1	0	0
$A.\overline{B}.C$	1	1	0	0	0	0	1	1
$A.B.C$	0	0	1	1	1	1	0	0
$A.B.\overline{C}$	1	1	0	0	0	0	1	1

Feedback

Email

D0

Map Layout

	$\overline{D}.\overline{E}.\overline{F}$	$\overline{D}.\overline{E}.F$	$\overline{D}.E.\overline{F}$	$\overline{D}.E.F$	$D.\overline{E}.\overline{F}$	$D.\overline{E}.F$	$D.E.\overline{F}$	$D.E.F$
$\overline{A}.\overline{B}.\overline{C}$	0	1	3	2	4	5	7	6
$\overline{A}.\overline{B}.C$	8	9	11	10	12	13	15	14
$\overline{A}.B.C$	24	25	27	26	28	29	31	30
$\overline{A}.B.\overline{C}$	16	17	19	18	20	21	23	22
$A.\overline{B}.\overline{C}$	32	33	35	34	36	37	39	38
$A.\overline{B}.C$	40	41	43	42	44	45	47	46
$A.B.C$	56	57	59	58	60	61	63	62
$A.B.\overline{C}$	48	49	51	50	52	53	55	54

Groups

(0,1)	$\overline{A}.\overline{B}.\overline{C}.\overline{D}.E$
(6,7)	$\overline{A}.\overline{B}.\overline{C}.D.E$
(10,11)	$\overline{A}.\overline{B}.C.\overline{D}.E$
(12,13)	$\overline{A}.\overline{B}.C.D.\overline{E}$
(18,19)	$\overline{A}.B.\overline{C}.D.\overline{E}$
(20,21)	$\overline{A}.B.C.\overline{D}.E$
(24,25)	$\overline{A}.B.C.\overline{D}.\overline{E}$
(30,31)	$\overline{A}.B.C.D.E$
(34,35)	$A.\overline{B}.\overline{C}.\overline{D}.\overline{E}$
(36,37)	$A.\overline{B}.\overline{C}.D.E$
(40,41)	$A.\overline{B}.C.\overline{D}.E$
(46,47)	$A.\overline{B}.C.D.\overline{E}$
(48,49)	$A.B.\overline{C}.\overline{D}.E$
(54,55)	$A.B.\overline{C}.D.\overline{E}$
(58,59)	$A.B.C.\overline{D}.E$
(60,61)	$A.B.C.D.\overline{E}$

y = A'B'C'D'E' + A'B'C'DE' + A'B'CD'E' + A'B'CDE' + A'BC'D'E' + A'BC'DE' + A'BCD'E' + A'BCDE' + AB'C'D'E' + AB'C'DE' + AB'CD'E' + AB'CDE' + ABC'D'E' + ABC'DE' + ABCD'E' + ABCDE'

Map

	$\overline{D}.\overline{E}.\overline{F}$	$\overline{D}.\overline{E}.F$	$\overline{D}.E.\overline{F}$	$\overline{D}.E.F$	$D.\overline{E}.\overline{F}$	$D.\overline{E}.F$	$D.E.\overline{F}$	$D.E.F$
$\overline{A}.\overline{B}.\overline{C}$	0	0	0	1	1	1	1	1
$\overline{A}.\overline{B}.C$	0	0	0	0	1	1	1	0
$\overline{A}.B.C$	0	0	0	1	1	1	1	1
$\overline{A}.B.\overline{C}$	0	0	0	0	1	1	1	0
$A.B.\overline{C}$	0	0	0	0	1	1	1	0
$A.\overline{B}.C$	0	0	0	1	1	1	1	1
$A.B.C$	0	0	0	0	1	1	1	0
$A.B.\overline{C}$	0	0	0	1	1	1	1	

D2

Map Layout

	$\overline{D}.\overline{E}.\overline{F}$	$\overline{D}.\overline{E}.F$	$\overline{D}.E.\overline{F}$	$\overline{D}.E.F$	$D.\overline{E}.\overline{F}$	$D.\overline{E}.F$	$D.E.\overline{F}$	$D.E.F$
$\overline{A}.\overline{B}.\overline{C}$	0	1	3	2	4	5	7	6
$\overline{A}.\overline{B}.C$	8	9	11	10	12	13	15	14
$\overline{A}.B.C$	24	25	27	26	28	29	31	30
$\overline{A}.B.\overline{C}$	16	17	19	18	20	21	23	22
$A.B.\overline{C}$	32	33	35	34	36	37	39	38
$A.\overline{B}.C$	40	41	43	42	44	45	47	46
$A.B.C$	56	57	59	58	60	61	63	62
$A.B.\overline{C}$	48	49	51	50	52	53	55	54

Groups

(4,5,12,13,20,21,28,29,36,37,44,45,52,53,60,61)	$D.\overline{E}$
(5,7,13,15,21,23,29,31,37,39,45,47,53,55,61,63)	D.F
(2,6)	$\overline{A}.\overline{B}.\overline{C}.E.\overline{F}$
(26,30)	$\overline{A}.B.C.E.\overline{F}$
(42,46)	$A.\overline{B}.C.E.\overline{F}$
(50,54)	$A.B.\overline{C}.E.\overline{F}$

y = DE' + DF + A'B'C'EF' + A'BCEF' + AB'CEF' + ABC'EF'

Map

	$\overline{D}.\overline{E}.\overline{F}$	$\overline{D}.\overline{E}.F$	$\overline{D}.E.\overline{F}$	$\overline{D}.E.F$	$D.\overline{E}.\overline{F}$	$D.\overline{E}.F$	$D.E.\overline{F}$	$D.E.F$
$\overline{A}.\overline{B}.\overline{C}$	0	0	0	0	0	0	0	0
$\overline{A}.\overline{B}.C$	1	0	0	0	0	0	0	0
$\overline{A}.B.C$	1	1	1	1	1	1	1	1
$\overline{A}.B.\overline{C}$	1	1	1	1	1	1	1	1
$A.B.\overline{C}$	0	0	0	0	0	0	0	0
$A.\overline{B}.C$	0	0	0	0	0	0	0	0
$A.B.C$	0	1	1	1	1	1	1	1
$A.B.\overline{C}$	1	1	1	1	1	1	1	1

D4

Map Layout

	$\overline{D}.\overline{E}.\overline{F}$	$\overline{D}.\overline{E}.F$	$\overline{D}.E.\overline{F}$	$\overline{D}.E.F$	$D.\overline{E}.\overline{F}$	$D.\overline{E}.F$	$D.E.\overline{F}$	$D.E.F$
$\overline{A}.\overline{B}.\overline{C}$	0	1	3	2	4	5	7	6
$\overline{A}.\overline{B}.C$	8	9	11	10	12	13	15	14
$\overline{A}.B.C$	24	25	27	26	28	29	31	30
$\overline{A}.B.\overline{C}$	16	17	19	18	20	21	23	22
$A.B.\overline{C}$	32	33	35	34	36	37	39	38
$A.\overline{B}.C$	40	41	43	42	44	45	47	46
$A.B.C$	56	57	59	58	60	61	63	62
$A.B.\overline{C}$	48	49	51	50	52	53	55	54

Groups

(16,17,18,19,20,21,22,23,48,49,50,51,52,53,54,55)	$B.\overline{C}$
(17,19,21,23,25,27,29,31,49,51,53,55,57,59,61,63)	B.F
(18,19,22,23,26,27,30,31,50,51,54,55,58,59,62,63)	B.E
(20,21,22,23,28,29,30,31,52,53,54,55,60,61,62,63)	B.D
(8,24)	$\overline{A}.C.\overline{D}.\overline{E}.\overline{F}$

y = BC' + BF + BE + BD + A'CD'E'F'

SUM of PRODUCTS

Map

	$\overline{D}.\overline{E}.\overline{F}$	$\overline{D}.\overline{E}.F$	$\overline{D}.E.\overline{F}$	$\overline{D}.E.F$	$D.\overline{E}.\overline{F}$	$D.\overline{E}.F$	$D.E.\overline{F}$	$D.E.F$
$\overline{A}.\overline{B}.\overline{C}$	0	1	1	1	0	0	0	1
$\overline{A}.\overline{B}.C$	0	0	0	1	0	1	1	1
$\overline{A}.B.C$	0	1	1	1	0	0	0	1
$\overline{A}.B.\overline{C}$	0	0	0	1	0	1	1	1
$A.B.\overline{C}$	0	0	0	1	0	1	1	1
$A.\overline{B}.C$	0	1	1	1	0	0	0	1
$A.B.C$	0	0	0	1	0	1	1	1
$A.B.\overline{C}$	0	1	1	1	0	0	0	1

Feedback

Email

D1

Map Layout

	$\overline{D}.\overline{E}.\overline{F}$	$\overline{D}.\overline{E}.F$	$\overline{D}.E.\overline{F}$	$\overline{D}.E.F$	$D.\overline{E}.\overline{F}$	$D.\overline{E}.F$	$D.E.\overline{F}$	$D.E.F$
$\overline{A}.\overline{B}.\overline{C}$	0	1	3	2	4	5	7	6
$\overline{A}.\overline{B}.C$	8	9	11	10	12	13	15	14
$\overline{A}.B.C$	24	25	27	26	28	29	31	30
$\overline{A}.B.\overline{C}$	16	17	19	18	20	21	23	22
$A.B.\overline{C}$	32	33	35	34	36	37	39	38
$A.\overline{B}.C$	40	41	43	42	44	45	47	46
$A.B.C$	56	57	59	58	60	61	63	62
$A.B.\overline{C}$	48	49	51	50	52	53	55	54

Groups

(2,6,10,14,18,22,26,30,34,38,42,46,50,54,58,62)	$E.\overline{F}$
(1,3)	$\overline{A}.\overline{B}.\overline{C}.D.F$
(13,15)	$\overline{A}.\overline{B}.C.D.F$
(21,23)	$\overline{A}.B.\overline{C}.D.F$
(25,27)	$\overline{A}.B.C.\overline{D}.F$
(37,39)	$A.\overline{B}.\overline{C}.D.F$
(41,43)	$A.\overline{B}.C.\overline{D}.F$
(49,51)	$A.B.\overline{C}.\overline{D}.F$
(61,63)	$A.B.C.D.F$

y = EF' + A'B'C'D'F + A'B'CDF + A'BC'DF + A'BCD'F + AB'C'DF + AB'CD'F + ABC'D'F + ABCDF

Map

	$\overline{D}.\overline{E}.\overline{F}$	$\overline{D}.\overline{E}.F$	$\overline{D}.E.\overline{F}$	$\overline{D}.E.F$	$D.\overline{E}.\overline{F}$	$D.\overline{E}.F$	$D.E.\overline{F}$	$D.E.F$
$\overline{A}.\overline{B}.\overline{C}$	0	0	0	0	1	0	0	0
$\overline{A}.\overline{B}.C$	1	1	1	1	1	1	1	1
$\overline{A}.B.C$	1	1	1	1	0	1	1	1
$\overline{A}.B.\overline{C}$	0	0	0	0	0	0	0	0
$A.B.\overline{C}$	0	0	0	0	0	0	0	0
$A.\overline{B}.C$	1	1	1	1	0	1	1	1
$A.B.C$	1	1	1	1	1	1	1	1
$A.B.\overline{C}$	0	0	0	0	1	0	0	0

D3

Map Layout

	$\overline{D}.\overline{E}.\overline{F}$	$\overline{D}.\overline{E}.F$	$\overline{D}.E.\overline{F}$	$\overline{D}.E.F$	$D.\overline{E}.\overline{F}$	$D.\overline{E}.F$	$D.E.\overline{F}$	$D.E.F$
$\overline{A}.\overline{B}.\overline{C}$	0	1	3	2	4	5	7	6
$\overline{A}.\overline{B}.C$	8	9	11	10	12	13	15	14
$\overline{A}.B.C$	24	25	27	26	28	29	31	30
$\overline{A}.B.\overline{C}$	16	17	19	18	20	21	23	22
$A.B.\overline{C}$	32	33	35	34	36	37	39	38
$A.\overline{B}.C$	40	41	43	42	44	45	47	46
$A.B.C$	56	57	59	58	60	61	63	62
$A.B.\overline{C}$	48	49	51	50	52	53	55	54

Groups

(8,9,10,11,24,25,26,27,40,41,42,43,56,57,58,59)	$C.\overline{D}$
(9,11,13,15,25,27,29,31,41,43,45,47,57,59,61,63)	C.F
(10,11,14,15,26,27,30,31,42,43,46,47,58,59,62,63)	C.E
(4,12)	$\overline{A}.\overline{B}.D.\overline{E}.\overline{F}$
(52,60)	$A.B.D.\overline{E}.\overline{F}$

y = CD' + CF + CE + A'B'DE'F' + ABDE'F'

Map

	$\overline{D}.\overline{E}.\overline{F}$	$\overline{D}.\overline{E}.F$	$\overline{D}.E.\overline{F}$	$\overline{D}.E.F$	$D.\overline{E}.\overline{F}$	$D.\overline{E}.F$	$D.E.\overline{F}$	$D.E.F$
$\overline{A}.\overline{B}.\overline{C}$	0	0	0	0	0	0	0	0
$\overline{A}.\overline{B}.C$	0	0	0	0	0	0	0	0
$\overline{A}.B.C$	0	0	0	0	0	0	0	0
$\overline{A}.B.\overline{C}$	1	0	0	0	0	0	0	0
$A.B.C$	0	1	1	1	1	1	1	1
$A.\overline{B}.C$	1	1	1	1	1	1	1	1
$A.B.C$	1	1	1	1	1	1	1	1
$A.B.\overline{C}$	1	1	1	1	1	1	1	1

D5

Map Layout

	$\overline{D}.\overline{E}.\overline{F}$	$\overline{D}.\overline{E}.F$	$\overline{D}.E.\overline{F}$	$\overline{D}.E.F$	$D.\overline{E}.\overline{F}$	$D.\overline{E}.F$	$D.E.\overline{F}$	$D.E.F$
$\overline{A}.\overline{B}.\overline{C}$	0	1	3	2	4	5	7	6
$\overline{A}.\overline{B}.C$	8	9	11	10	12	13	15	14
$\overline{A}.B.C$	24	25	27	26	28	29	31	30
$\overline{A}.B.\overline{C}$	16	17	19	18	20	21	23	22
$A.B.\overline{C}$	32	33	35	34	36	37	39	38
$A.\overline{B}.C$	40	41	43	42	44	45	47	46
$A.B.C$	56	57	59	58	60	61	63	62
$A.B.\overline{C}$	48	49	51	50	52	53	55	54

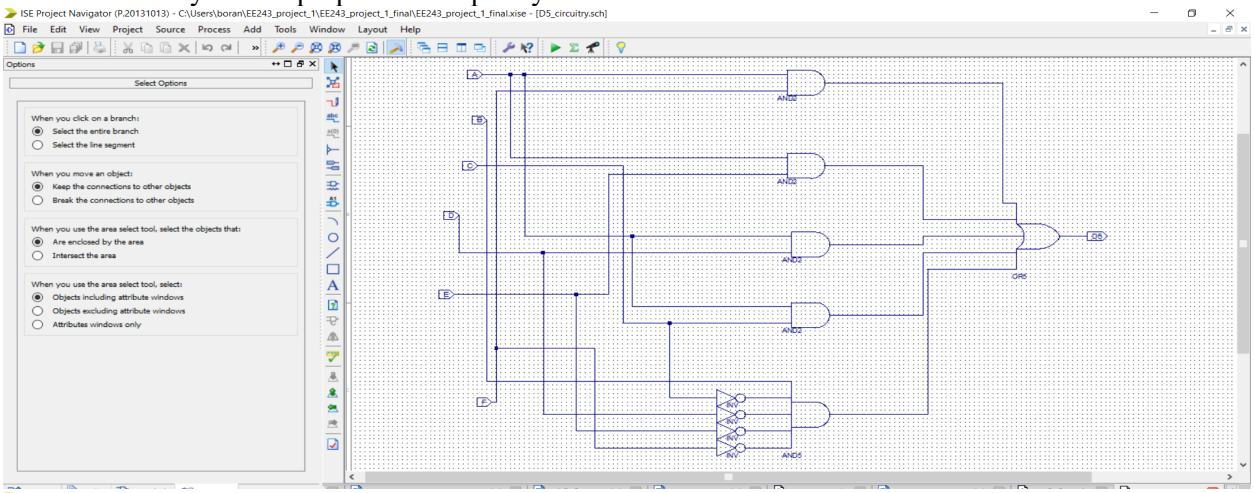
Groups

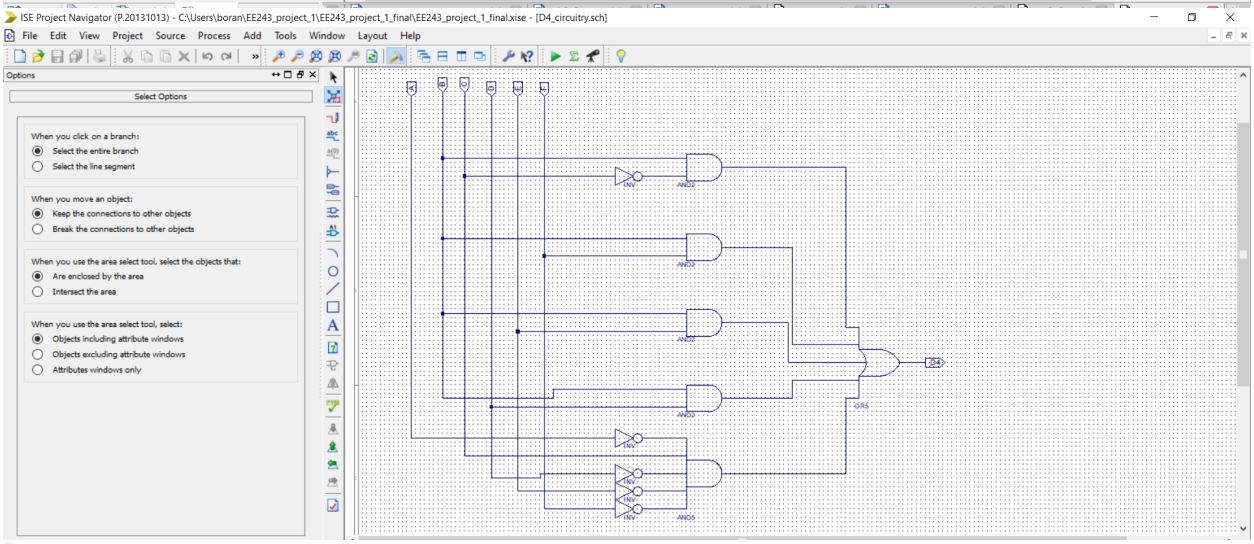
(33,35,37,39,41,43,45,47,49,51,53,55,57,59,61,63)	A.F
(34,35,38,39,42,43,46,47,50,51,54,55,58,59,62,63)	A.E
(36,37,38,39,44,45,46,47,52,53,54,55,60,61,62,63)	A.D
(40,41,42,43,44,45,46,47,56,57,58,59,60,61,62,63)	A.C
(16,48)	$B.\overline{C}.\overline{D}.\overline{E}.\overline{F}$

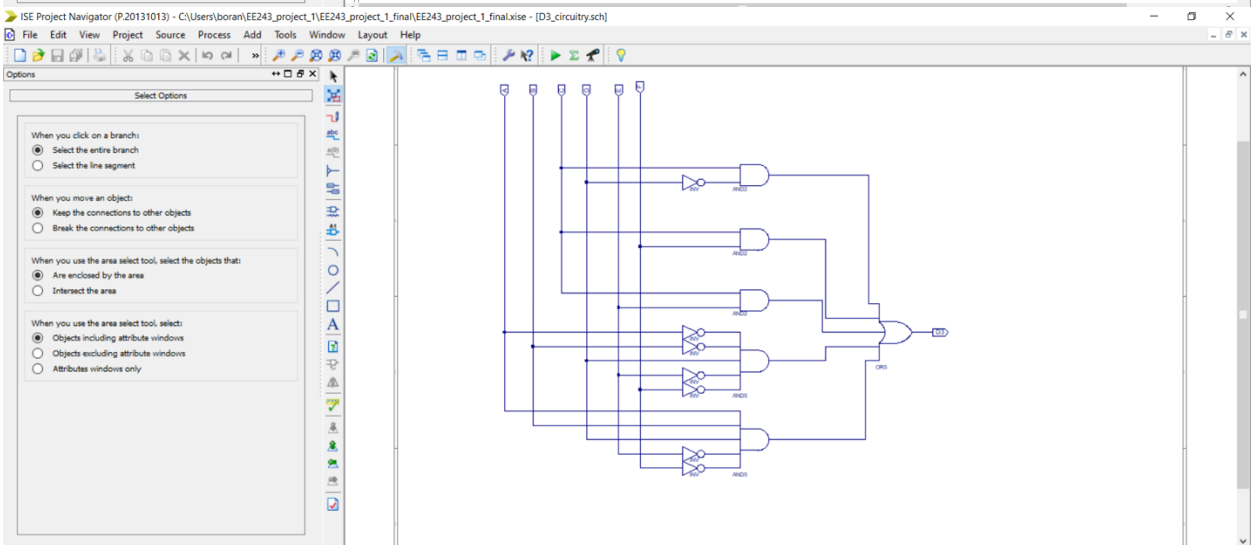
y = AF + AE + AD + AC + BC'D'E'F'

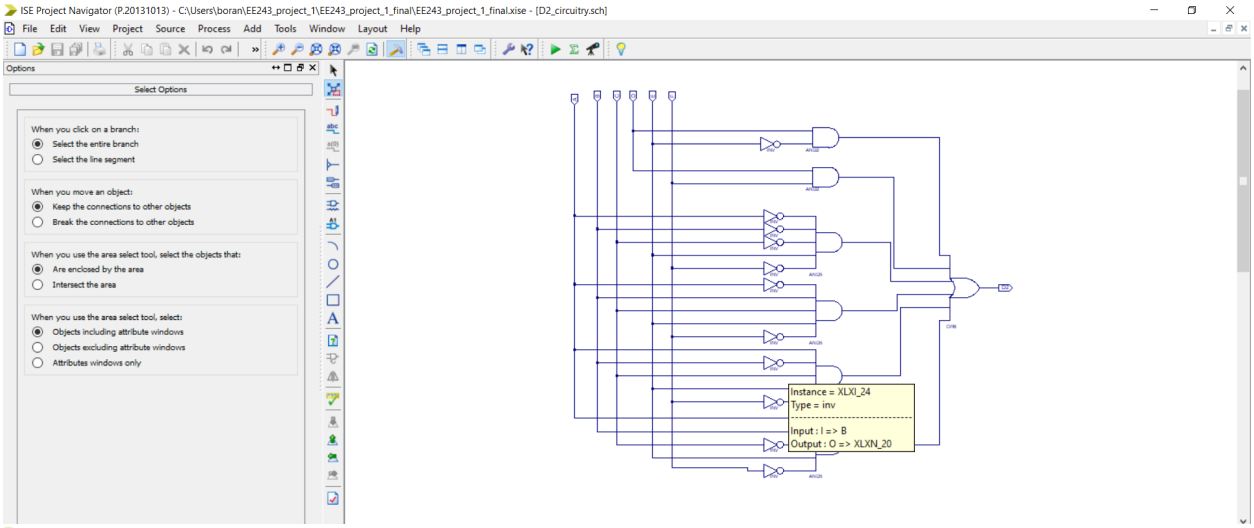
[4]

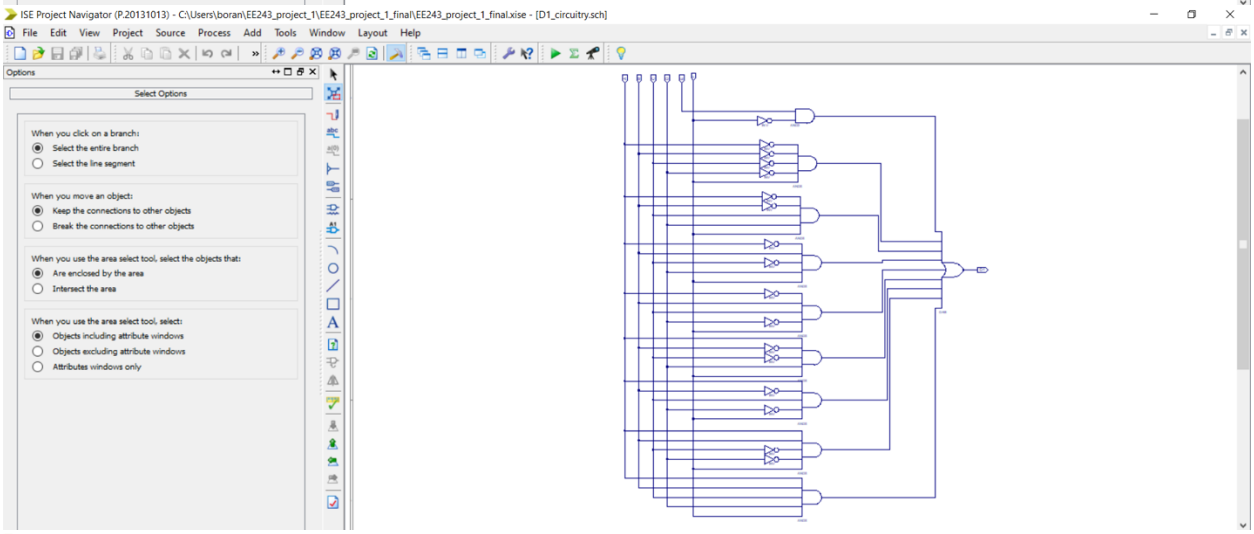
Now as we've determined the inputs of flip-flop, we can create separate circuitries for each DFF. You can see below circuitries of each DFF; D5 at the top and D0 at the bottom. A, B, C, D, E, F are inputs corresponding to G5, G4,…,G0 with A being the most significant bit. I have created a schematic symbol for each circuitry for the purpose of simplicity.

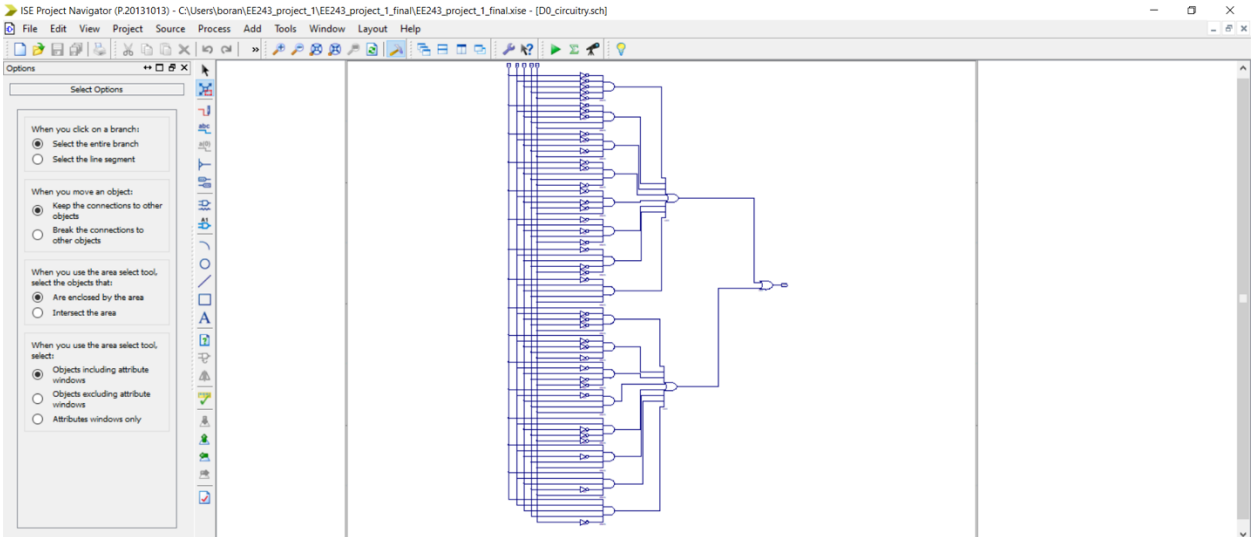






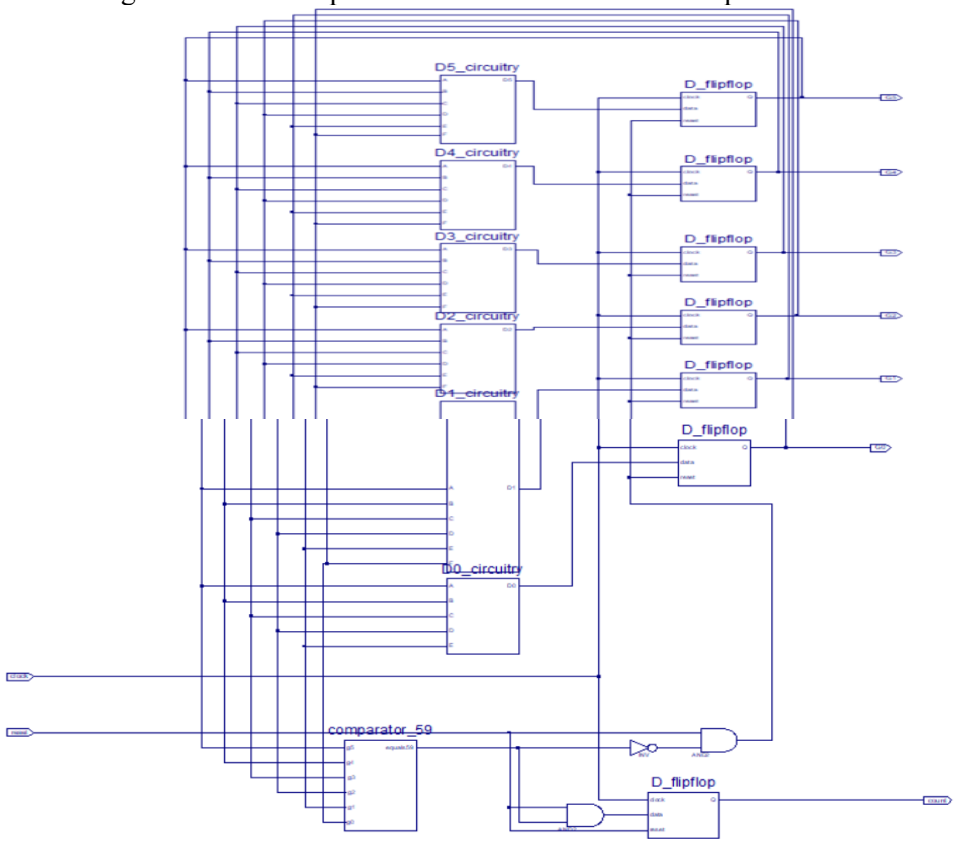

ISE Project Navigator (P.20131013) - C:\Users\boran\EE243_project_1\EE243_project_1_final\EE243_project_1_final.xise - [D2_circuitry.sch]
File Edit View Project Source Process Add Tools Window Layout Help
Options
Select Options
When you click on a branch:
Select the entire branch
Select the line segment
When you move an object:
Keep the connections to other objects
Break the connections to other objects
When you use the area select tool, select the objects that:
Are enclosed by the area
Intersect the area
When you use the area select tool, select:
Objects including attribute windows
Objects excluding attribute windows
Attributes windows only
Instance = XLXI_24
Type = inv
Input : I => B
Output : O => XLXN_20


ISE Project Navigator (P.20131013) - C:\Users\boran\EE243_project_1\EE243_project_1_final\EE243_project_1_final.xise - [D1_circuitry.sch]
File Edit View Project Source Process Add Tools Window Layout Help
Options
Select Options
When you click on a branch:
Select the entire branch
Select the line segment
When you move an object:
Keep the connections to other objects
Break the connections to other objects
When you use the area select tool, select the objects that:
Are enclosed by the area
Intersect the area
When you use the area select tool, select:
Objects including attribute windows
Objects excluding attribute windows
Attributes windows only


ISE Project Navigator (P.20131013) - C:\Users\boran\EE243_project_1\EE243_project_1_final\EE243_project_1_final.xise - [D0_circuitry.sch]
File Edit View Project Source Process Add Tools Window Layout Help
Options
Select Options
When you click on a branch:
Select the entire branch
Select the line segment
When you move an object:
Keep the connections to other objects
Break the connections to other objects
When you use the area select tool, select the objects that:
Are enclosed by the area
Intersect the area
When you use the area select tool, select:
Objects including attribute windows
Objects excluding attribute windows
Attributes windows only

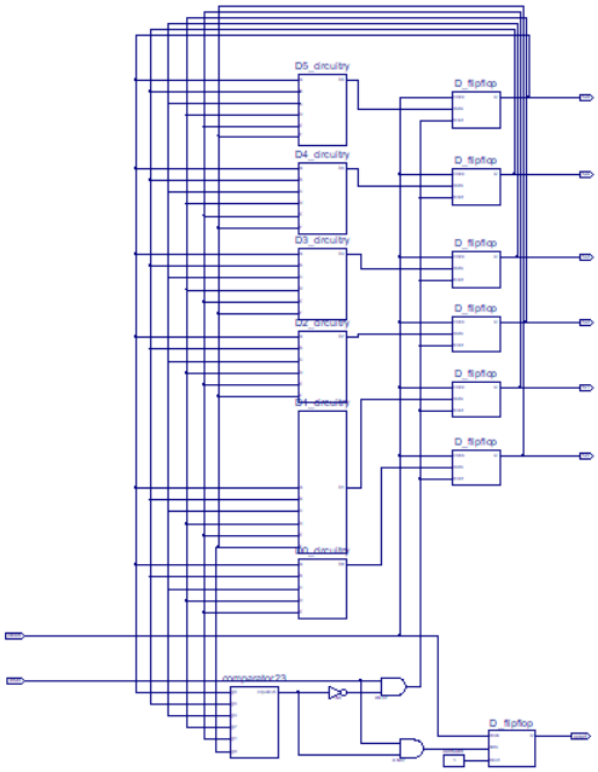
PART 2.2: Adding the reset and count

You can see below the complete design of the 0-59 counter. Each flip-flop circuitry I have constructed is connected to the corresponding flip-flop. But K-maps I have created were for a 0-63 counter. So, we need to reset them when counter reaches 59. I have created a 59 comparator which is fed by flip-flop outputs. This comparator outputs 1 when we're on 59 and otherwise 0. When the output of this comparator is 1, it gets inverted and resets each flip-flop to 0. Also, since we have a general reset input, we should connect the inverted comparator output and reset input by an AND gate. Our flip-flops will be reset when either reset input is zero or when counter has reached 59. It important that our counter is not reset exactly when counter hits 59. Rather we should deliver the reset signal at 59th second but flip-flops should be updated at the next clock signal which corresponds to the 60th second as we require.



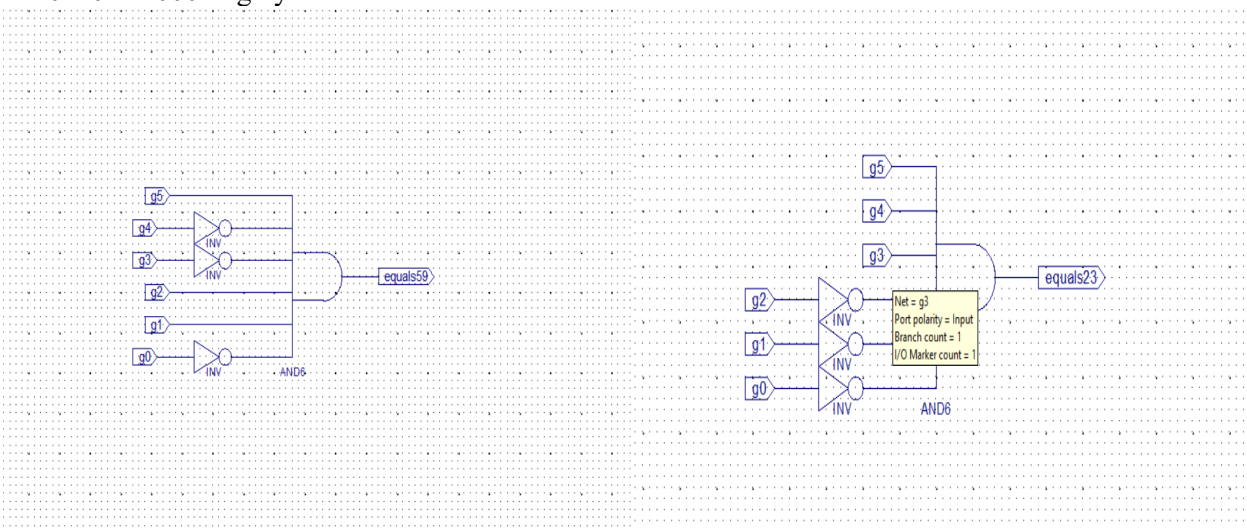
Moreover, we should find a way of indicating a full minute cycle (60 seconds) has passed. In order to achieve this, we should deliver the output of 59_comparator to an external output "count". However directly connecting them doesn't work and we should modify this a little bit. Firstly, for our minute clock to update not on 59th second but the 60th second, we should introduce a delay of 1 clock cycle. This can be achieved with a DFF. DFF stores the value of the comparator and outputs it in the next clock cycle. Secondly, when I first run my first test simulations, I have encountered a problem. Initial state of the minute and hour counters were undetermined, and this resulted in a 'X' value in my simulation. In order to prevent this from happening, we can connect the output of comparator and reset signals by an AND gate. This way, when reset=0, data input of the first flip-flop is assured to be 0 and we can safely reset all counters at the beginning of our simulation while avoiding undetermined states. After the 59th second, each DFF turns back to 0 and count signal goes back to 0 again.

Design of the 0-23 counter is pretty similar to the 0-59 counter. Only difference is the comparator being used which should be the 23 comparator for 0-23 counter.

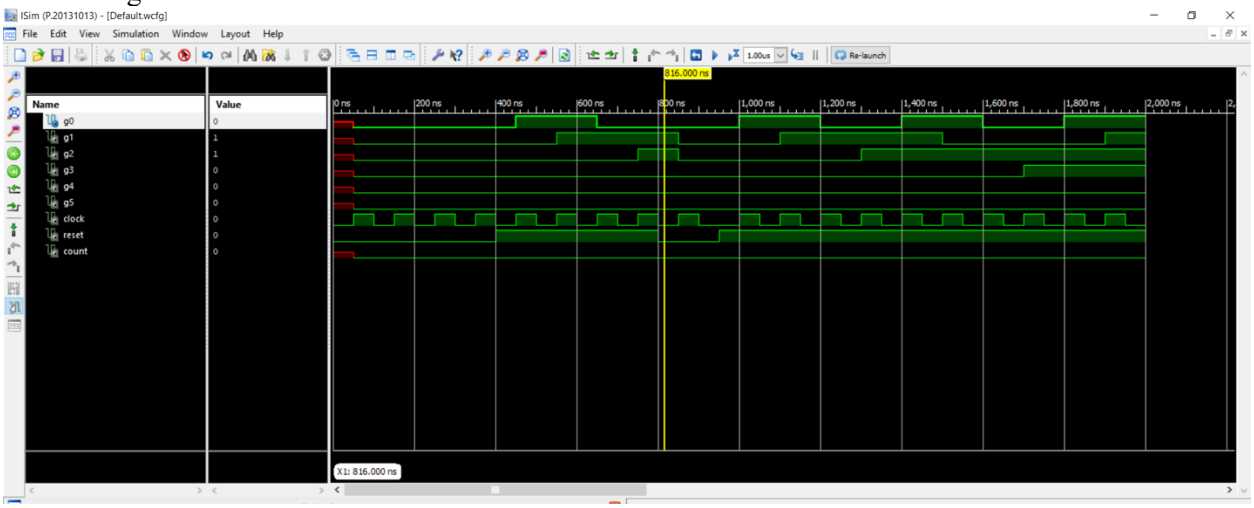


PART 2.3: Comparator

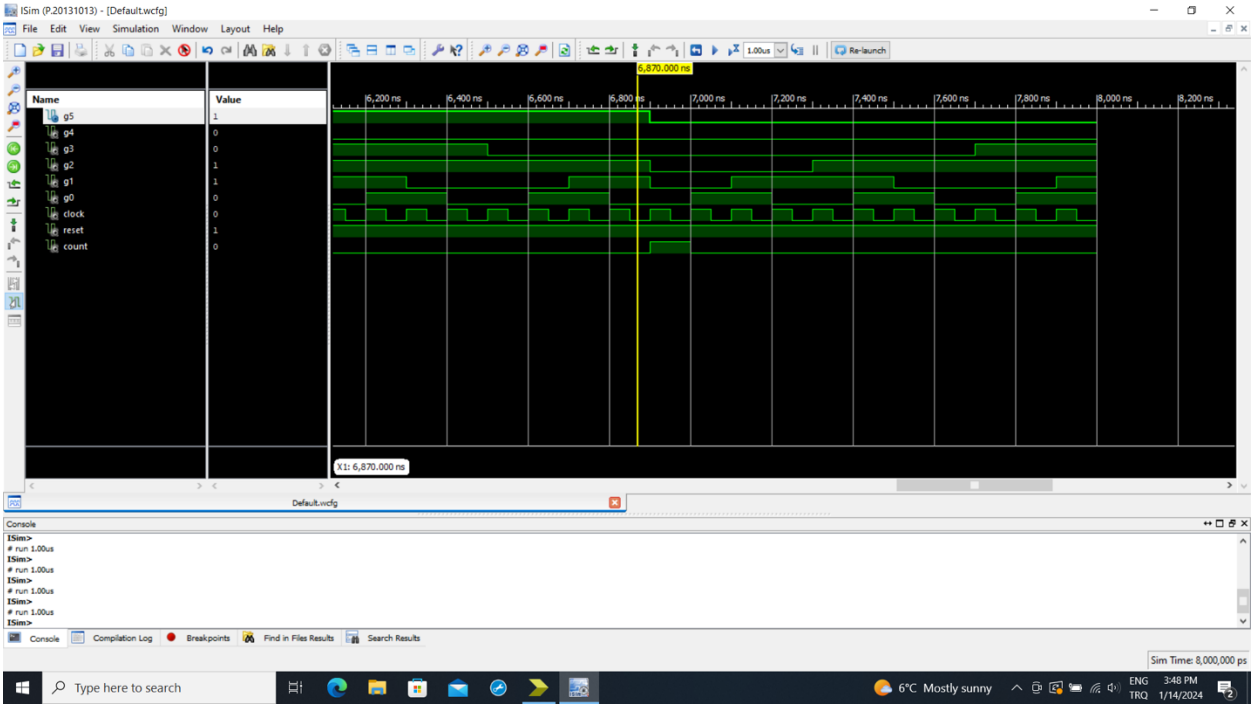
You can see the 59 and 23 comparators below. They output zero unless input is 59/23 which corresponds to 10110/111000 in gray code.



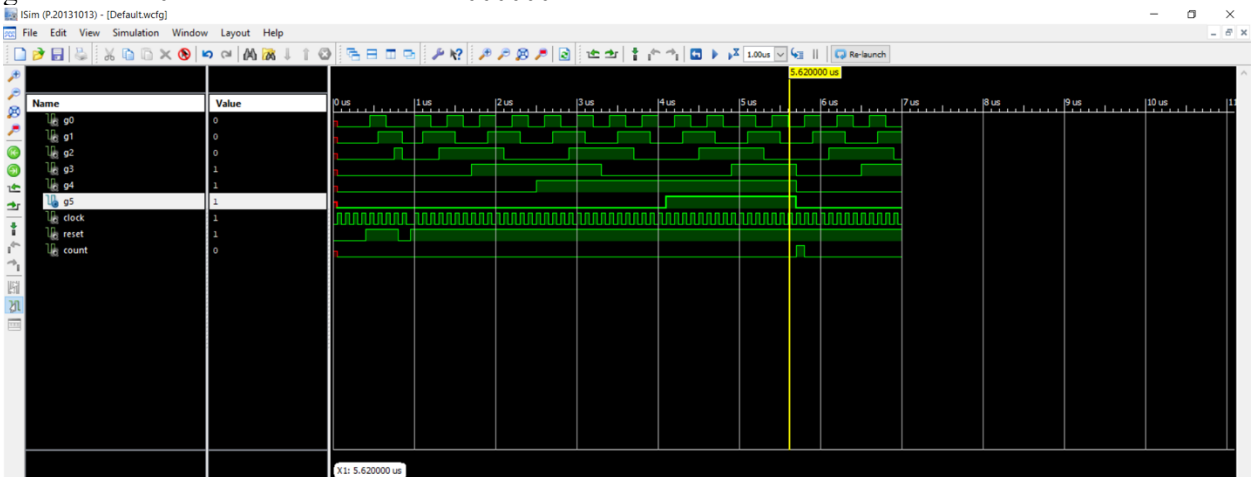
As you can see below when reset=1, counter starts counting. If reset is switched to 0 counter goes back to 000000 and stays like this until reset is switched to 1 again. Here g5 is the most significant bit and g0 is the least significant bit.



If we fast forward to 59th second, we can see after 100110 (59 in decimal), counter goes back to 000000 and starts counting again. You can also see count signal is updated at the 60th second and goes back to 0 when counter is reset to 000000.

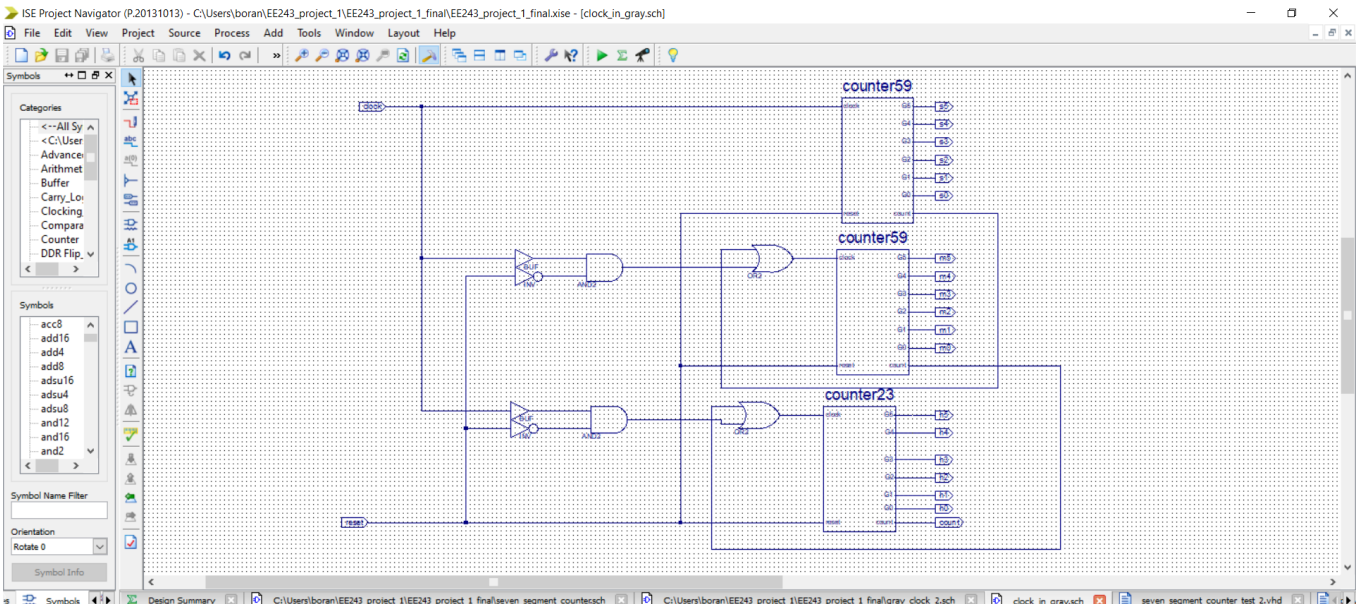


We can see the familiar gray code pattern for the 0-23 counter as well. After 111000 (23 in decimal), counter goes back to 000000 as expected. You can also see count signal is updated at the 24th second and goes back to 0 when counter is reset to 000000.

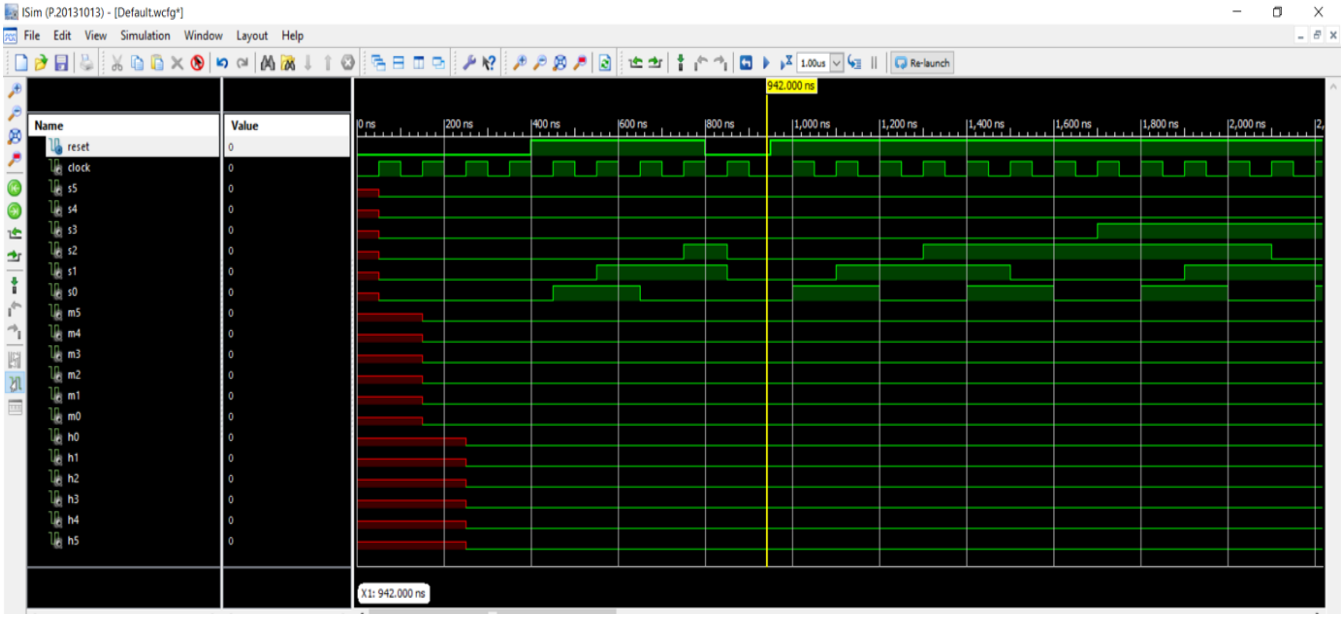


PART 3: Complete gray clock

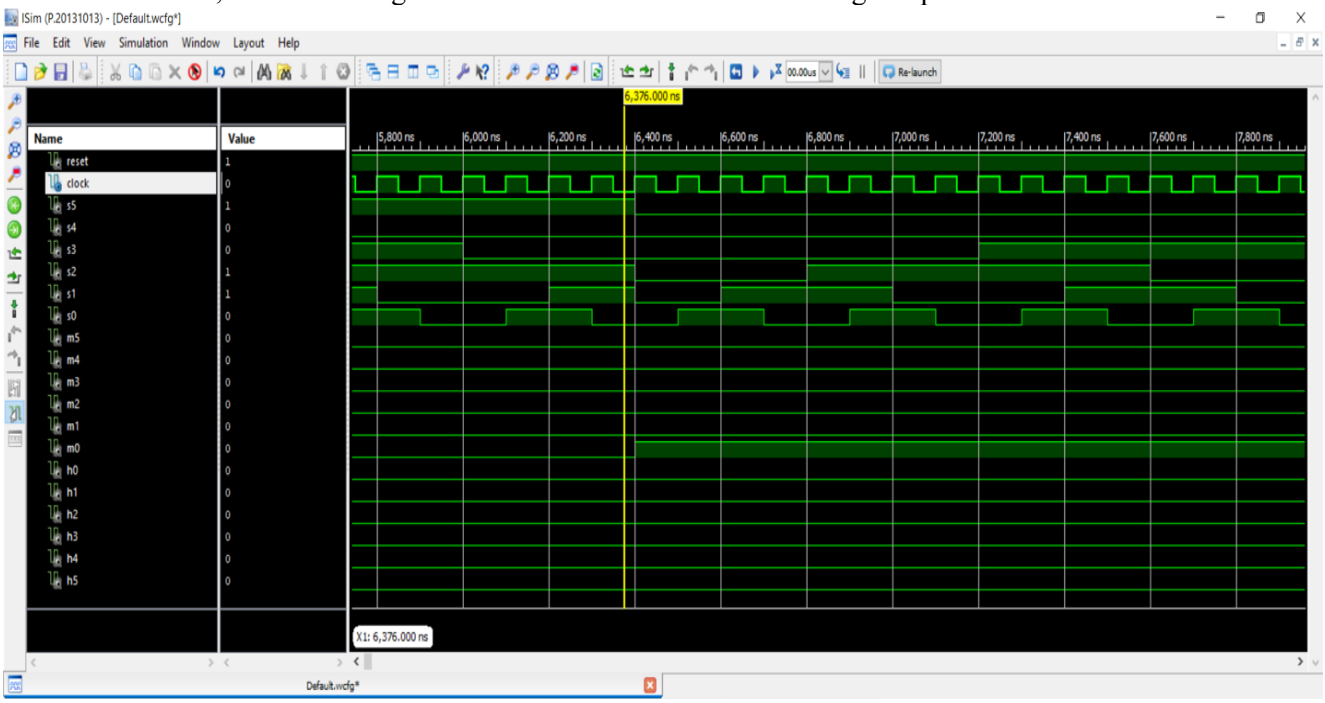
As we've created the 0-59 and 0-23 counter, we can combine them all as shown in 'General Schematics' part. Here s5, s4,…, s0 are second outputs, m5, m4,…, m0 are minute outputs and h5, h4,…, h0 are hour outputs. Higher number indicates a more significant bit.



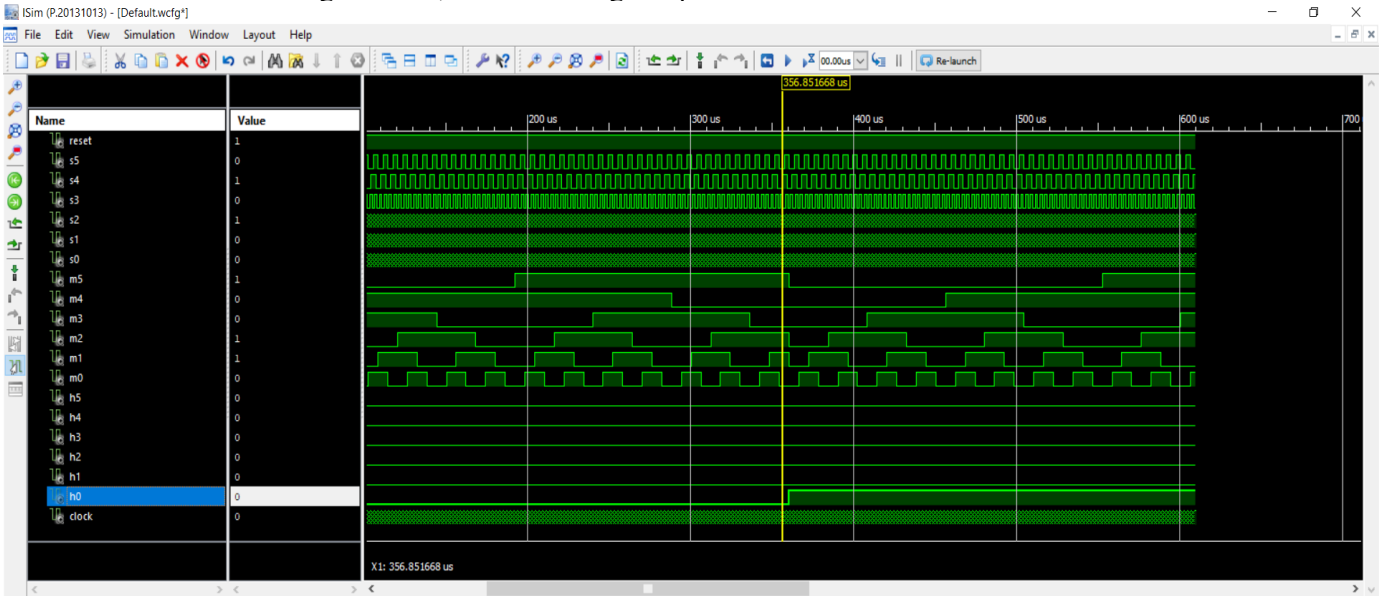
Red parts in the simulation indicates that state of outputs is undetermined. In order to tackle this problem, we should reset each DFF in each counter at the beginning. You can see undetermined states gradually returning to normal. And you can see that when reset=0, counter resets to 000000 and stays like that until reset is switched to 1.



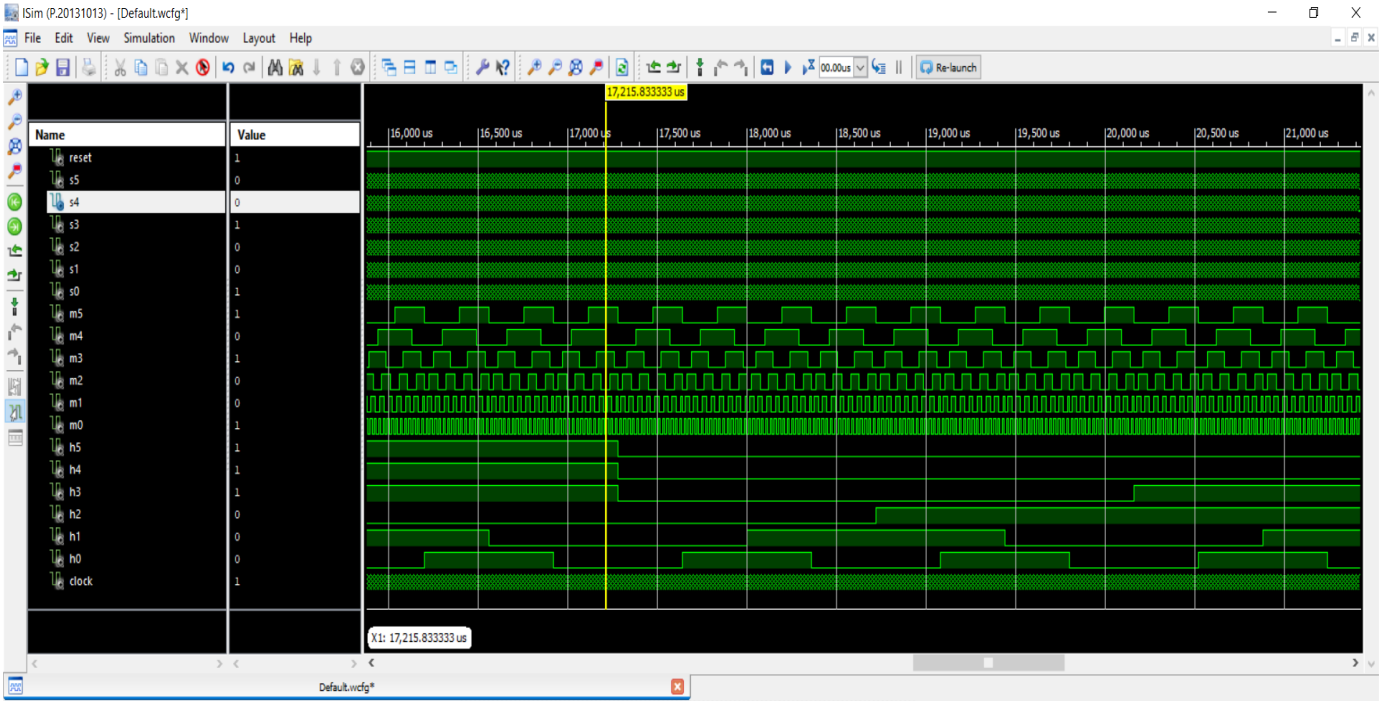
At the 60th second, second count goes back to 000000 and minute count gets updated to 000001.



And after minute counter reaches 59 which means m5=1, m4=0, m3=0, m2=1, m1=1, m0=0; minute counter goes back to 0 at the next cycle of minute counter. In this case next cycle corresponds to next minute and not the clock signal. Also, hour counter gets updated to 000001.

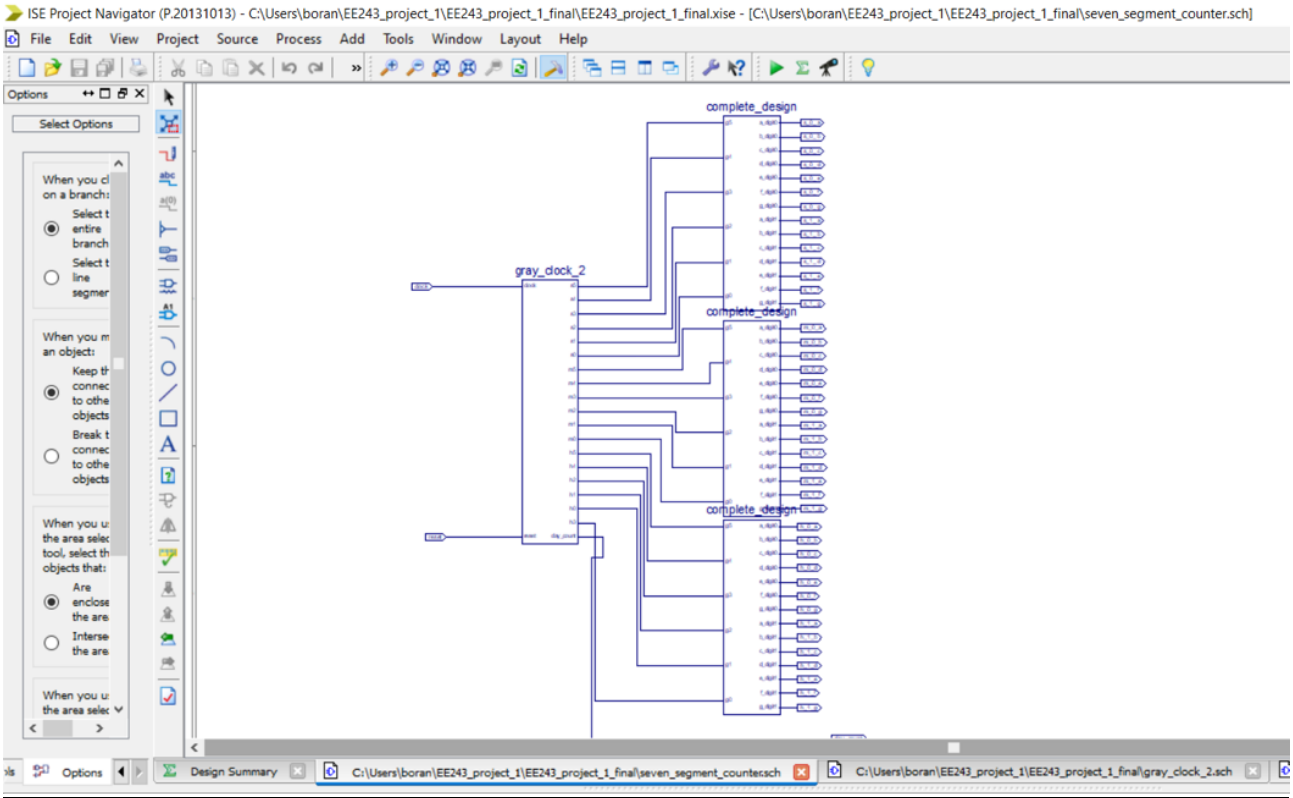


Finally, after hour counter reaches 23 which means h5=1, h4=0, h3=0, h2=1, h1=1, h0=0; hour counter goes back to 0 at the next cycle of hour counter. In this case next cycle corresponds next hour. Also, hour counter goes back to 000000 as expected.



PART 4: Converting Gray Code to 7-segment Display

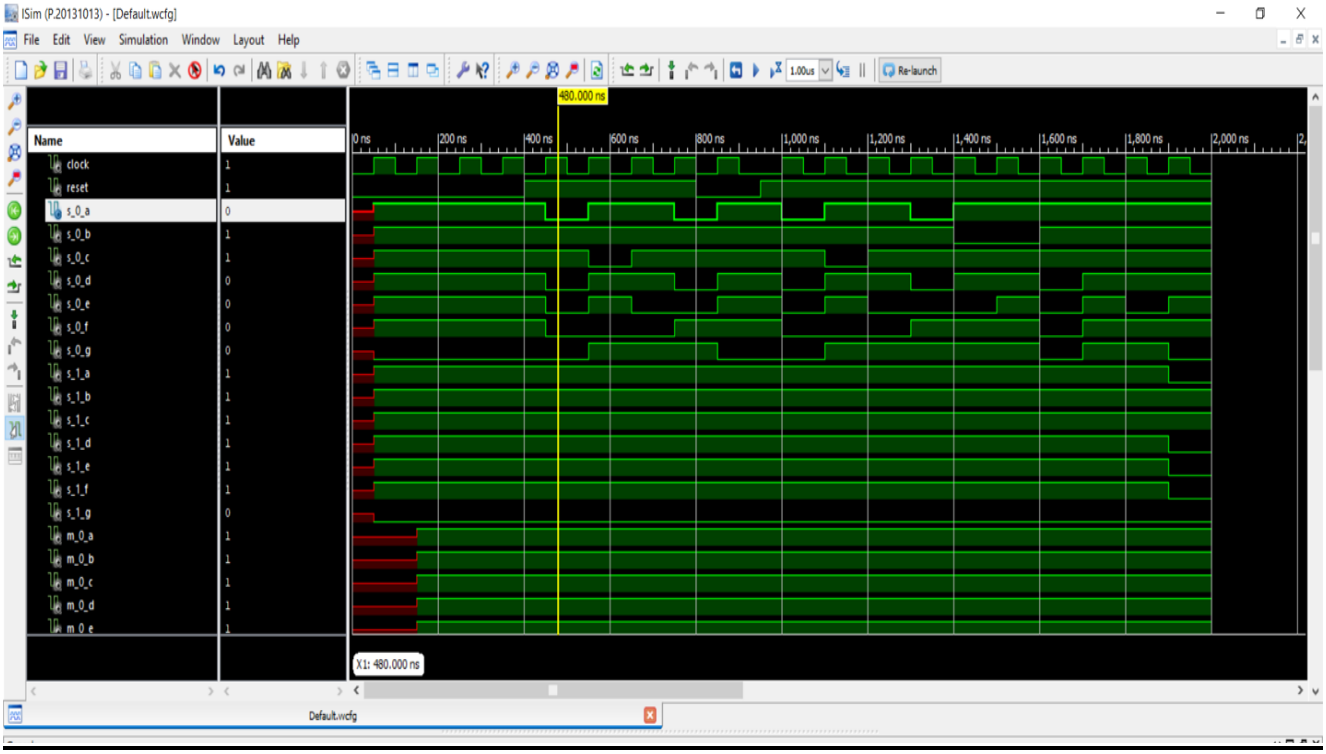
So far, we've constructed a digital clock in gray code which can be reset with a reset input. This clock should be fed by a 1 Hz external clock. Now we should convert gray code outputs of the digital clock to seven-segment display. We have already designed a combinational circuit that converts a 6-bit number in gray code to segment display in the first project. Second, minute and hour segments are each represented with a 2 digit number in seven-segment display. All we need to do is combine the digital clock with 3 gray to seven-segment converters representing seconds, minutes and hours respectively. Note that our gray clock schematic symbol has a day_count output indicating a full 24 hours have passed.



Here outputs are named in the following way:
hour/minute/second_digit number (0 or 1)_ part of the seven-segment(a,b,c,d,e,f,g)

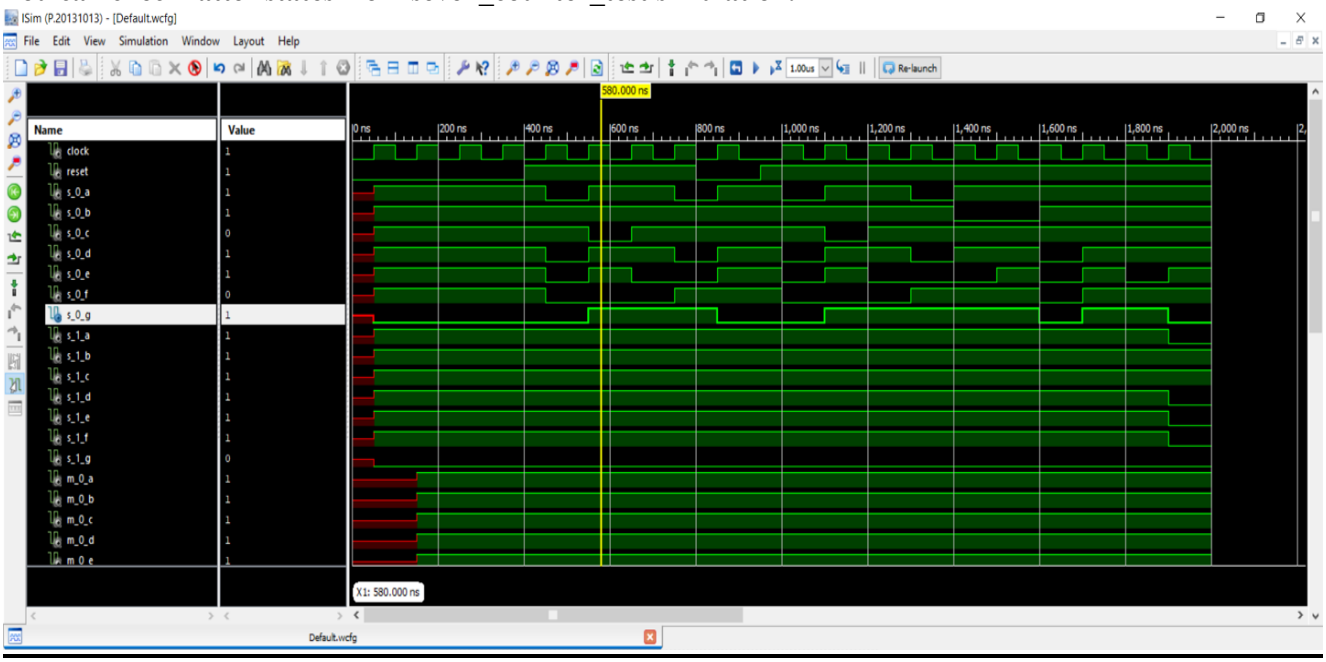
For example, m_0_b output is the b segment of the first digit corresponding to the minute part of the clock.

As you can see at the beginning, s_0_a, s_0_b, …, s_0_f are 1 and s_0_e is 0. This corresponds to number 0 as expected. After reset=1 and our clock start counting at the first rising edge. Only s_0_b, s_0_c are 1 which corresponds to number 1.



Here minute and hour segments stays at number 0 until 59 seconds or minutes have passed.
Since our clock in gray code works perfectly fine, we can expect it to work smoothly when we connect a combinational circuit to the end. Since it will not intervene the states of the sequential circuit by any means.

You can check latter states from seven_counter_test simulation.



References:

- [1]: This schematic is created using: https://app.diagrams.net/#G1E3smVA9Il5r9WBE06bxPMam-QGJHj1Xa
- [2]: Digital Design, M. Morris Mano, page 204
- [3]: Digital Design, M. Morris Mano, page 210
- [4]: K-maps are constructed using: http://www.32x8.com/index.html