Introduction: This project introduces a voting machine handling a maximum of 60 votes for six participants. The methodology involves six subsystems (Vote Counters), each assigned to a participant (A, B, C, D, E, F). These subsystems utilize six 8-bit counters and five 8-bit comparators, constructed by cascading two 7485IC (4-bit magnitude comparators). The Vote Counter architecture involves cascading two 75190IC (4-bit counters) for accurate vote tallying. The aim is to efficiently determine the Chairman and Vice-Chairman based on the collective votes.

Specification and Requirement:

Requirement:

In this following project to develop a majority voting system for electing the 2023 Executive body of the company where 60 members can cast vote to select 3 participants from 6 participants we have used several types of logical components in the logic circuit.

Fundamental logic gates:

- AND gate
- OR gate

Combinational logic circuits:

- Comparator (cascading two 7485 IC)

Sequential circuit:

- Counter (cascading two 4 bit 74190 IC)

Display section:

- 7 segment display

Additionally

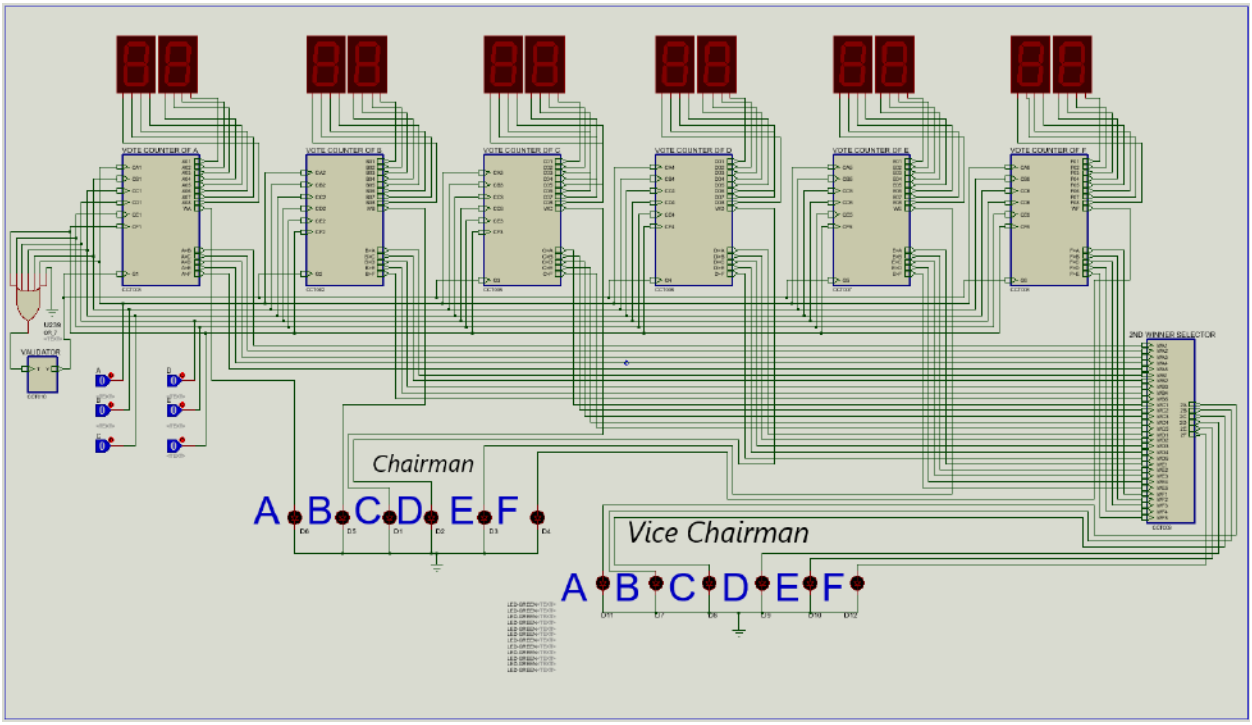
For input section:

- Logic Toggle

For output section:

- LED green

Our main circuit Diagram:



<u>Sub-Circuits:</u> To make our circuit look easy and tidier, we have created some Sub circuits. All the subcircuits and their child sheets are given below.

Vote counter:

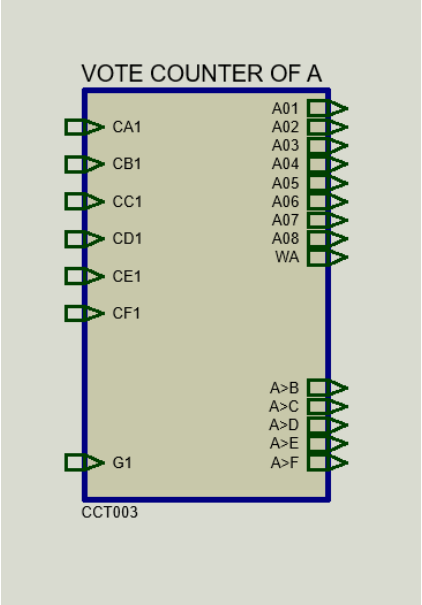


Fig: Vote counter made for A

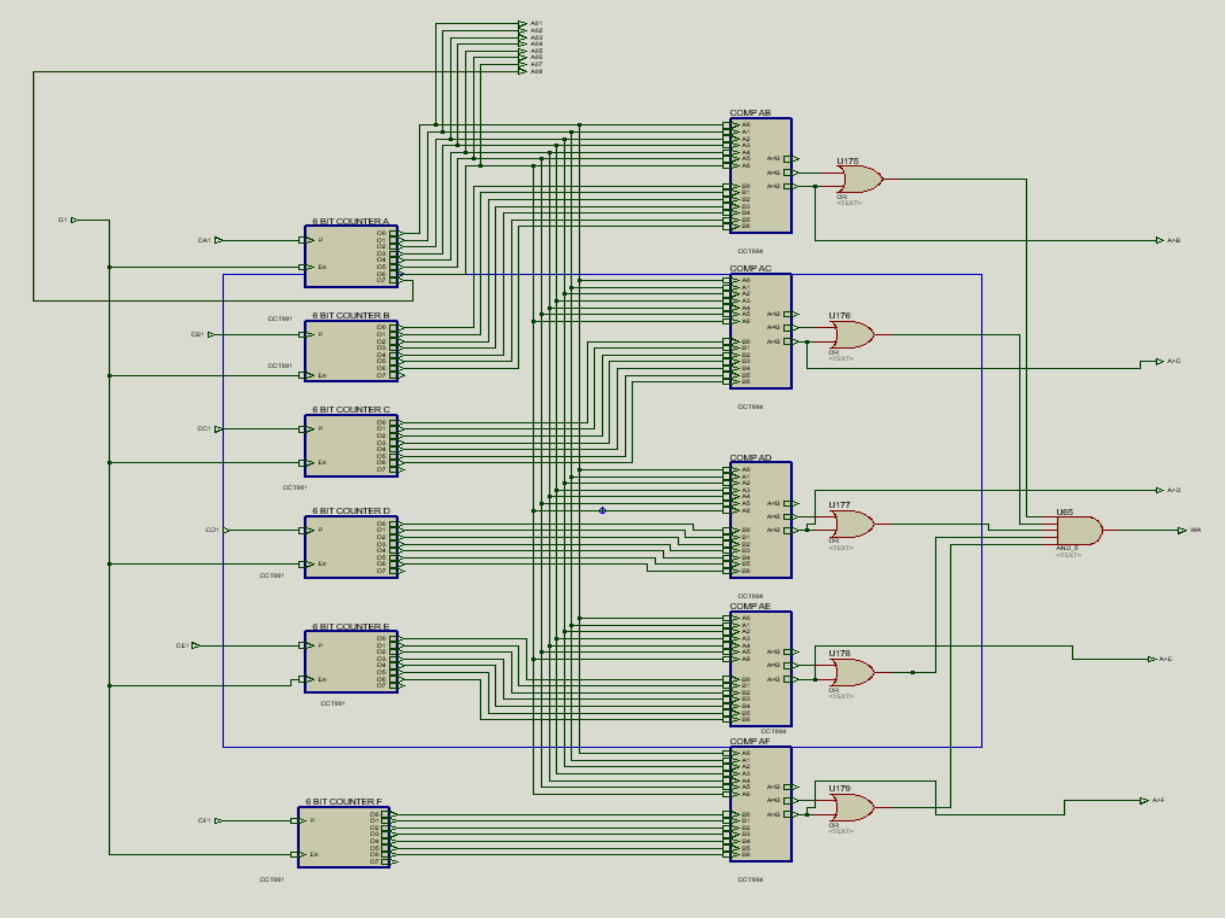


Fig: Child sheet of the Vote counter A

8-Bit Counter:

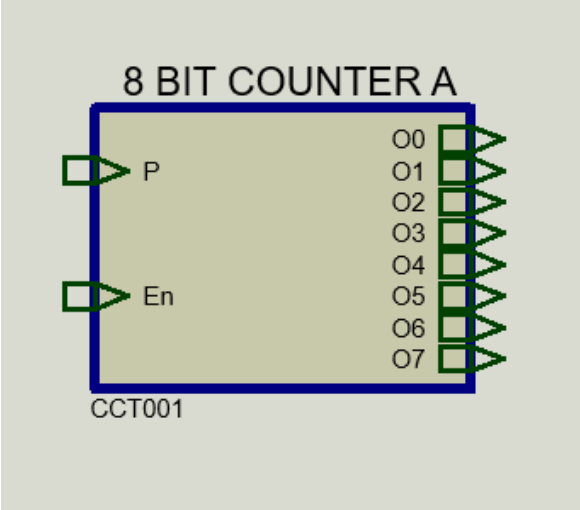


Fig: 8 Bit counter

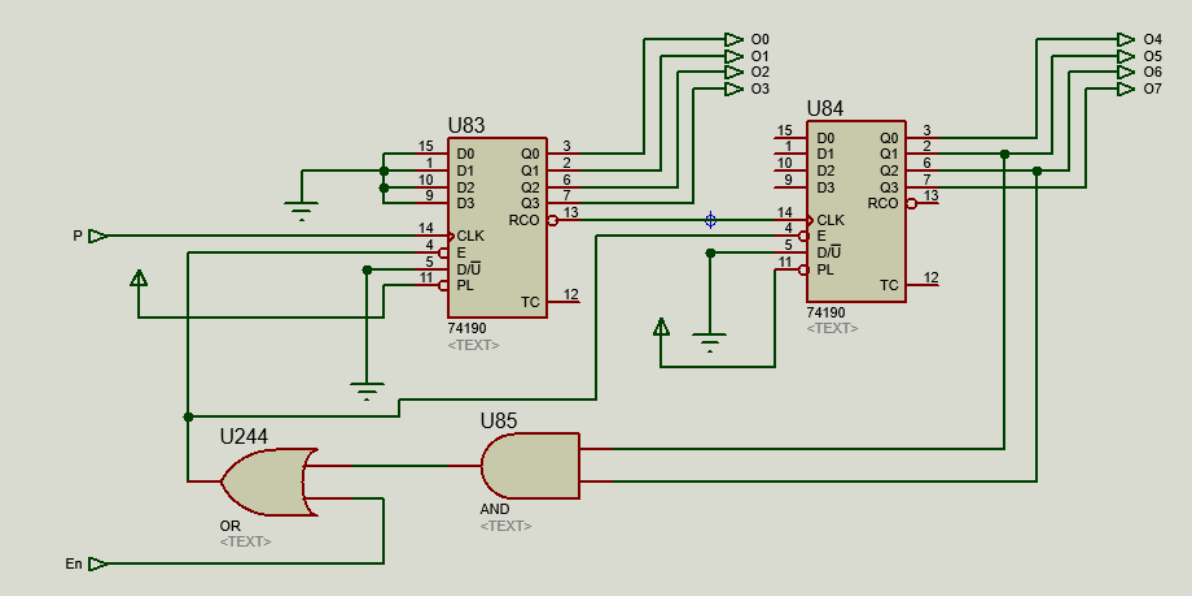


Fig: Childsheet of 8 Bit counter

As each 74190 counter can count up to (2^4)-1= 15, and we need to count up to 60, we have cascaded 2 ICs together. To do so, we give the input pulse to **CLK** only on the first counter and the **RCO** pin from the first counter has been connected with **CLK** of the second counter. And, to stop the counting at the 60th position, we have added an **AND gate** which is connected with the **Q1** and **Q2** outputs of the 2nd counter IC. We have connected specifically on those outputs because, In BCD 60 = 0110 0000 (as the output of the first counters, is less significant than the output of the second counter). So, in the 60th state, only these outputs will remain high, and by entering them with the one AND gate and connecting it with the Enable Pin(**E**) of both counters we can stop counting when 60 votes have been counted.

8-Bit Comparator:

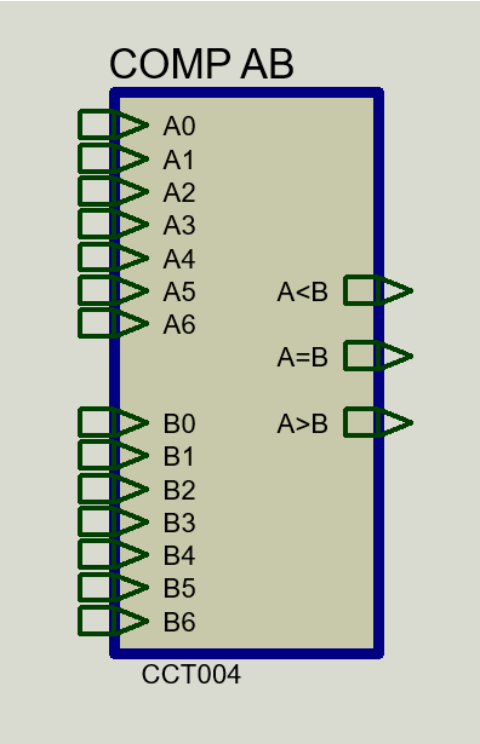


Fig: 8-Bit comparator Sub-circuit.

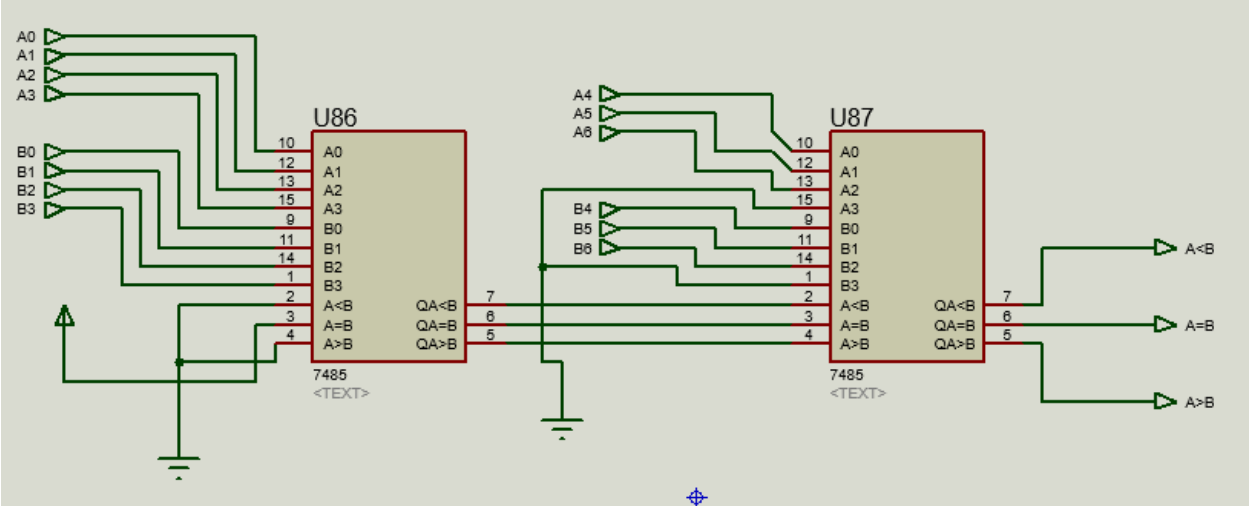


Fig: Childsheet 8 Bit comparator.

As we needed to compare a minimum of 6 bits, we also needed to cascade two 7485ICs. To do so We just needed to connect the outputs of the first comparator to the second comparator.

2nd Winner Selector: This is not a sub-system. We have made this just to tidy up the whole circuit. Mainly, this thing works to identify the 2nd winner. This thing is described in the **Methodology** section.

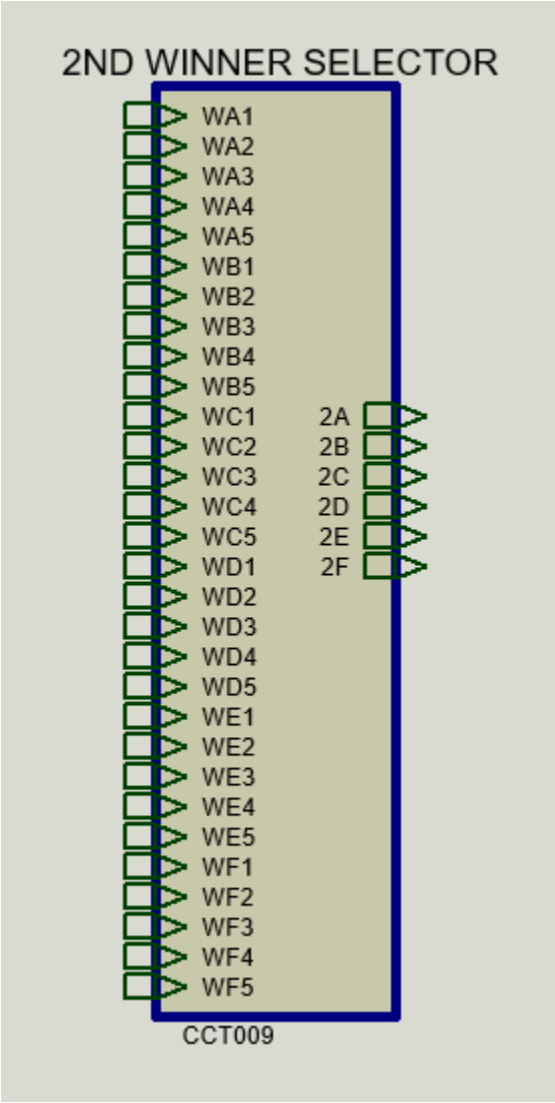


Fig: Sub-circuit of 2nd position selector.

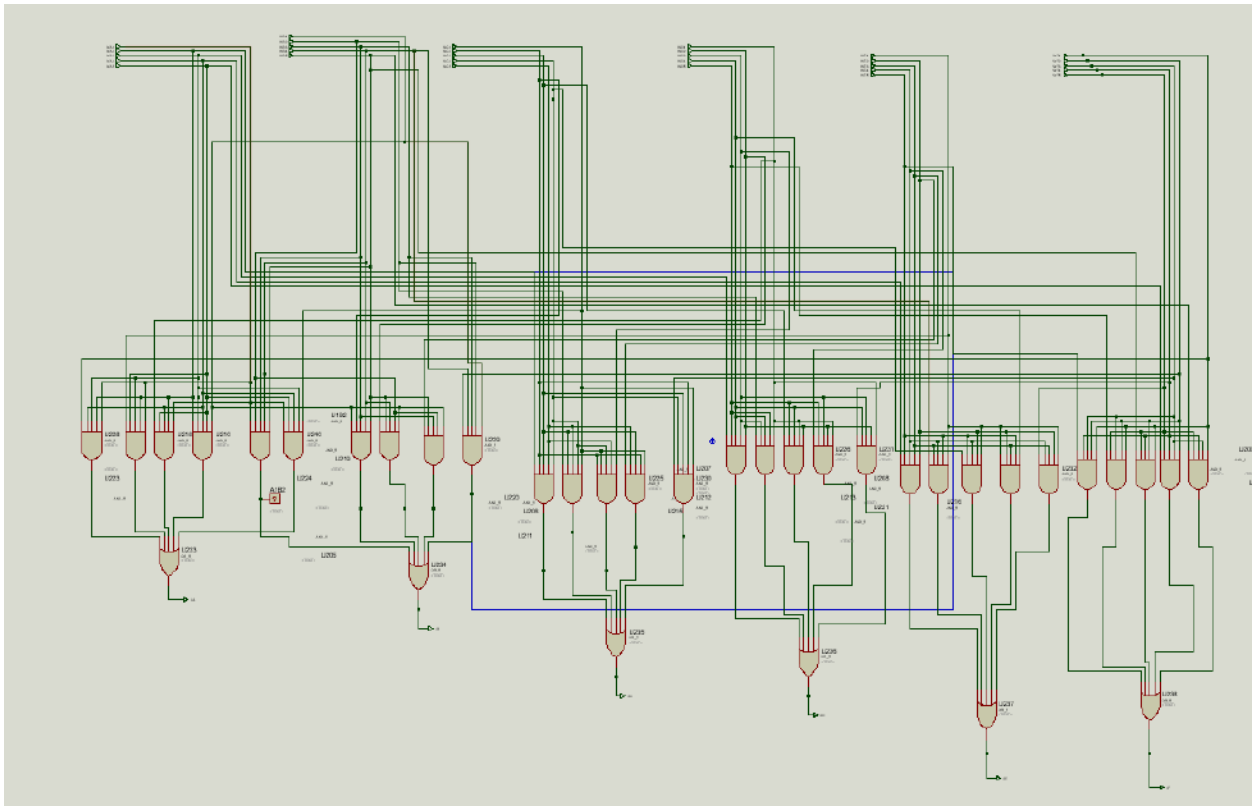
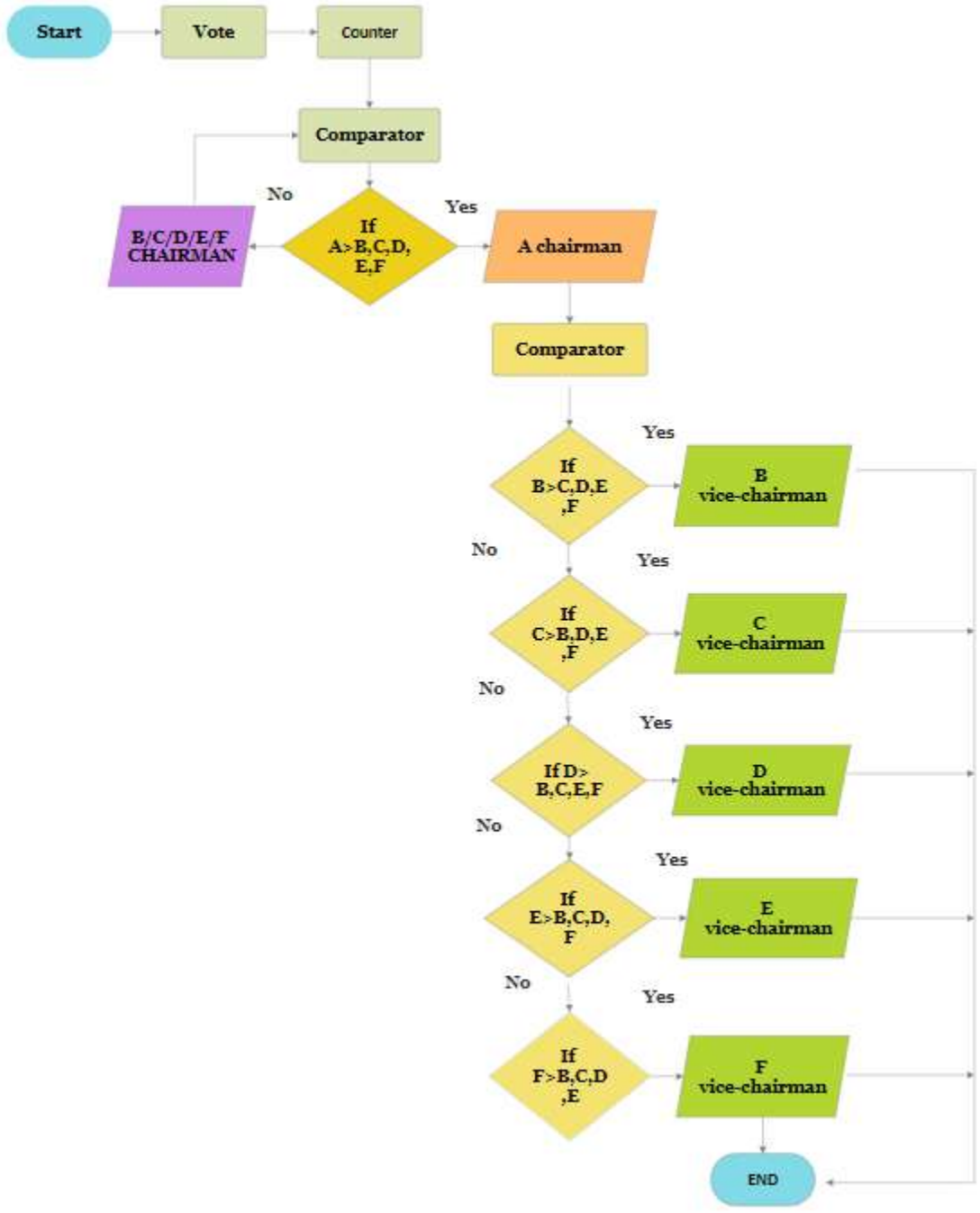


Fig: Childsheet of 2nd position selector.

Flow chart of our voting system:



Algorithm Development:

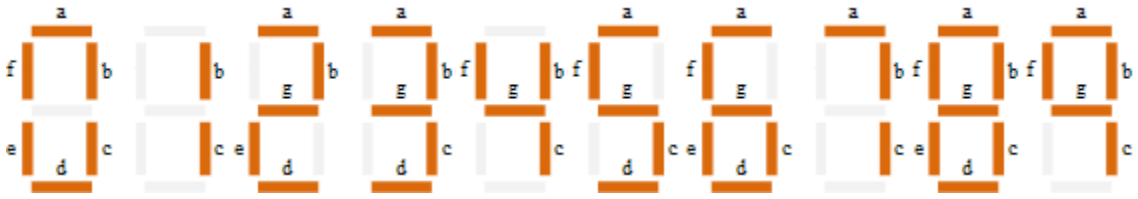
Step 1: start
Step 2: taking votes
Step 3: counting each participant's votes
Step 4: comparing vote numbers of all
Step 5: checking if A>B, C, D, E, F if yes go to step 7
Step 6: B/C/D/E/F chairman
Step 7: A chairman
Step 8: comparing among B, C, D, E, F
Step 9: checking if B>C, D, E, F if "YES" go to step 19 if "NO" go to step 11
Step 10: B vice chairman
Step 11: checking if C>B, D, E, F if "YES" go to step 19 if "NO" go to step 13
Step 12: C vice chairman
Step 13: checking if D>B, C, E, F if "YES" go to step 19 if "NO" go to step 15
Step 14: D vice chairman
Step 15: checking if E>B, C, D, F if "YES" go to step 19 if "NO" go to step 17
Step 16: E vice chairman
Step 17: checking if F>B, C, D, E if "NO" go to step 8
Step 18: F vice chairman
Step 19: End

NB: Algorithm will be similar for selecting B, C, D, E, F as a chairman. Also for the vice chairman position steps will remain the same.

In this way after selecting the Chairman and Vice-Chairman the 3rd highest-voted participant will be the Secretary.

BCD to 7-segment Display:

We have used BCD to 7-segment Display to show the counting votes for each participant.



	Input				Output (Segment)						
Digit	**A**	**B**	**C**	**D**	**a**	**b**	**c**	**d**	**e**	**f**	**g**
0	**0**	**0**	**0**	**0**	**1**	**1**	**1**	**1**	**1**	**1**	**0**
1	**0**	**0**	**0**	**1**	**0**	**1**	**1**	**0**	**0**	**0**	**0**
2	**0**	**0**	**1**	**0**	**1**	**1**	**0**	**1**	**1**	**0**	**1**
3	**0**	**0**	**1**	**1**	**1**	**1**	**1**	**1**	**0**	**0**	**1**
4	**0**	**1**	**0**	**0**	**0**	**1**	**1**	**0**	**0**	**1**	**1**
5	**0**	**1**	**0**	**1**	**1**	**0**	**1**	**1**	**0**	**1**	**1**
6	**0**	**1**	**1**	**0**	**1**	**0**	**1**	**1**	**1**	**1**	**1**
7	**0**	**1**	**1**	**1**	**1**	**1**	**1**	**0**	**0**	**0**	**0**
8	**1**	**0**	**0**	**0**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
9	**1**	**0**	**0**	**1**	**1**	**1**	**1**	**1**	**0**	**1**	**1**

In a seven-segment display, there are 4 inputs and the output number is 16. In the display, it can show 0 to 9.
For example, we can see when A=B=C=D=0 then, a=b=c=d=e=f=1 and g= 0. For this reason, the screen will show ‘0’

From the above truth table, the Boolean expressions of each output function can be written as.,

a = F1 (A, B, C, D) = ∑m (0, 2, 3, 5, 7, 8, 9)

b = F2 (A, B, C, D) = ∑m (0, 1, 2, 3, 4, 7, 8, 9)

c = F3 (A, B, C, D) = ∑m (0, 1, 3, 4, 5, 6, 7, 8, 9)

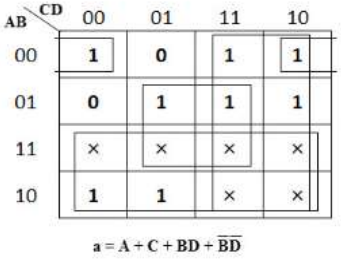
d = F4 (A, B, C, D) = ∑m (0, 2, 3, 5, 6, 8)

e = F5 (A, B, C, D) = ∑m (0, 2, 6, 8)

f = F6 (A, B, C, D) = ∑m (0, 4, 5, 6, 8, 9)

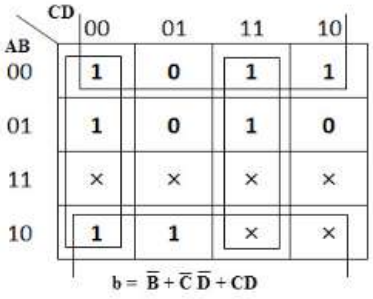
g = F7 (A, B, C, D) = ∑m (2, 3, 4, 5, 6, 8, 9)

For a:



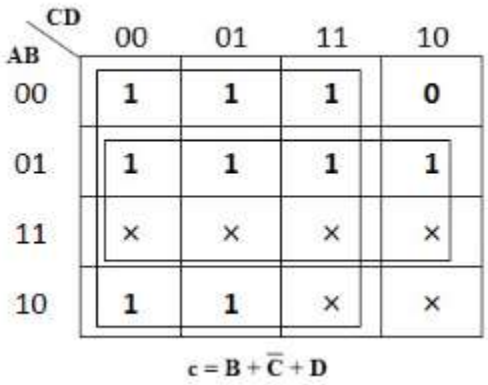
Here, **a= A+C+BD+B'D'**

For b:



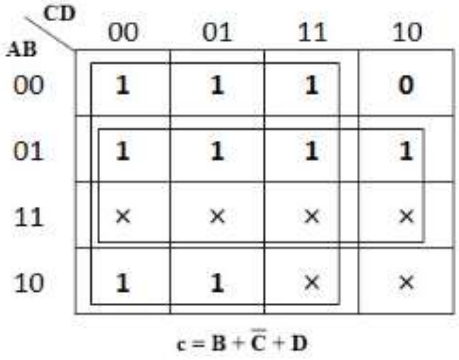
b= B'+C'D'+CD

For c:



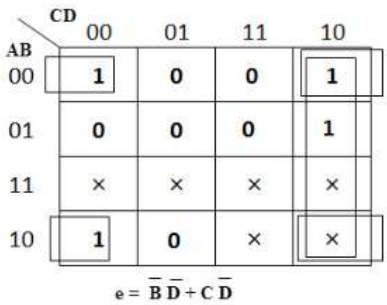
c= B+C'+D

For d:



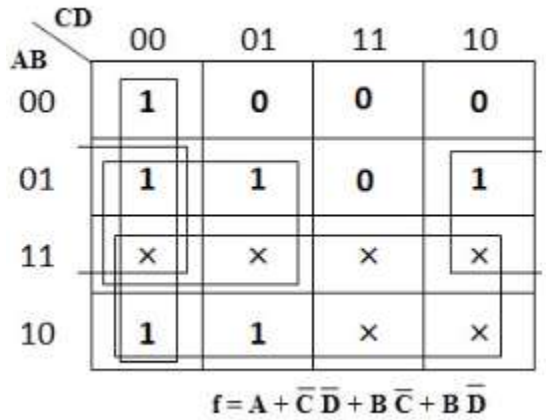
d= B'D'+CD'+BC'D+B'C+A

For e:



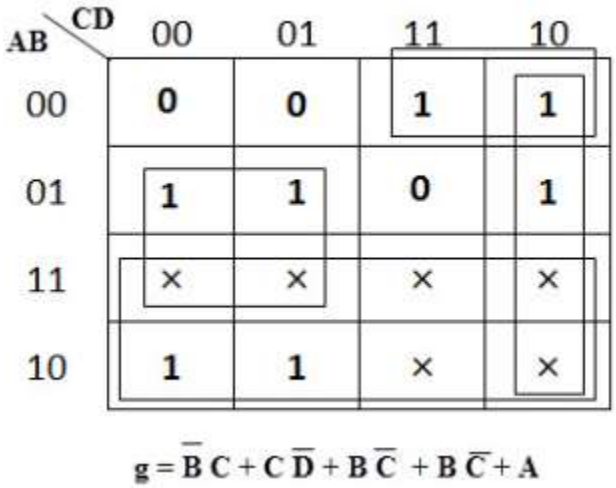
e= B'D+CD'

For f:



f= A+C'D+BC'+BD'

For g:



g= B'C+CD'+BC'+BC'+A

Comparator: 74190 IC is a 4-bit magnitude comparator IC. It determines whether the magnitude of two numbers entered are equal, more than, or less than one another. Comparator works in a bit-by-bit comparison way. It starts comparing from the most significant bit, and it compares al the way down to the least significant bit. The truth table of a 1-bit comparator is given below:

A	B	A=B	A>B	A<B
0	0	1	0	0
0	1	0	0	1
1	0	0	1	0
1	1	1	0	0

From the truth table,

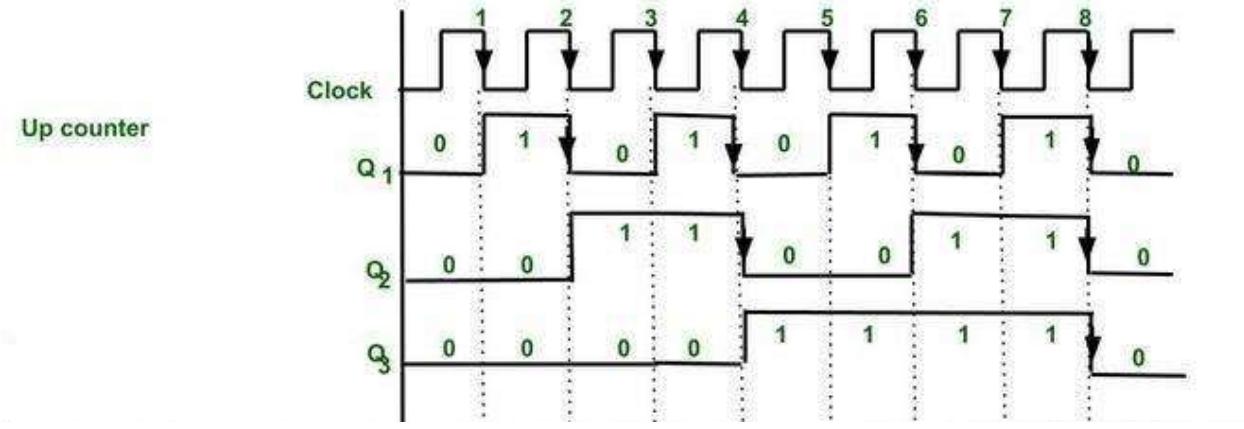
(A=B) = A'B'+AB= (A⊕B)'

A>B= AB'

A<B= A'B

In the very similar way, it counts all the way up to 4 bit. As, in our project, we are dealing with a very big number (60). We know that a MOD 64 (6-bit) counter is required to compare it. So we have cascaded two **7485** ICs to count that number.

Counter:



In a counter clock signal defines when the transition state of output will change. In Our circuit, for every vote count, we have connected the individual vote switch to their designated vote-counting machine.

As a result, if anyone presses a vote, the Output transition will change to the next logic and it will count. As an example, if any voter presses a vote in the particular switch for A person's vote will count for person A.

Truth table for Up Counter:

Clock Signal	A	B	C	D	Decimal value
1st	0	0	0	0	0
2nd	0	0	0	1	1
3rd	0	0	1	0	2
4th	0	0	1	1	3
5th	0	1	0	0	4
6th	0	1	0	1	5
7th	0	1	1	0	6
8th	0	1	1	1	7
9th	1	0	0	0	8
10th	1	0	0	1	9
11th	1	0	1	0	10
12th	1	0	1	1	11
13th	1	1	0	0	12
14th	1	1	0	1	13
15th	1	1	1	0	14
16th	1	1	1	1	15

Number of bit = 4

Total states= 2^4 =16

Maximum Count = (16-1)

= 15 As we have to count upto 60. we have cascaded 2 ICs together.

OR Gate:

We have used OR gates for any of the input is high the output will be high and if all inputs are low the output will be Low.

Truth Table for OR Gate:

A	B	C	F
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

AND Gate:

To select the first winner and second winner we have used AND gate multiple times in our simulation.

In an AND gate when all inputs are high for this case the output will be high and if any of the input is low the output will be low.

Truth table for AND Gate:

A	B	C	D	F
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

Methodology: In our voting system we can take up to 60 votes in total. Based on the total cast votes, our system can declare who is the winner(chairman) and who is the second position winner(vice-chairman). Suppose 6 participants are **A, B, C, D, E,** and **F**. So, based on that we divided our system into 6 subsystems. In every subsystem, only one participant wins. We have named this subsystem as **Vote counters**(such as Vote Counter of A)**.** In every **Vote counters,** there are in total six **8-bit counters** And five **8-bit comparators.** The **Vote counter** is made by cascading two **75190IC** (4-bit counters) and the **8-bit comparator** is made by cascading two **7485IC**(4-bit magnitude comparator)**.**

In the Sub-system where A wins(Vote Counter of A)**.-** the output value of the **8-bit counters** which calculate the vote got by **A** will go to the first half of all five **8-bit comparators.** The second half of the comparators will be connected with the output value of the **8-bit counters** which are calculating the votes of **B**, **C**, **D**, **E**, and **F**. We know that a participant only can win the vote if he/she gets more votes than all other participants. So, When a situation like **A> B, C, D, E, F** arises, participant **A** will be declared as the winner of the vote.

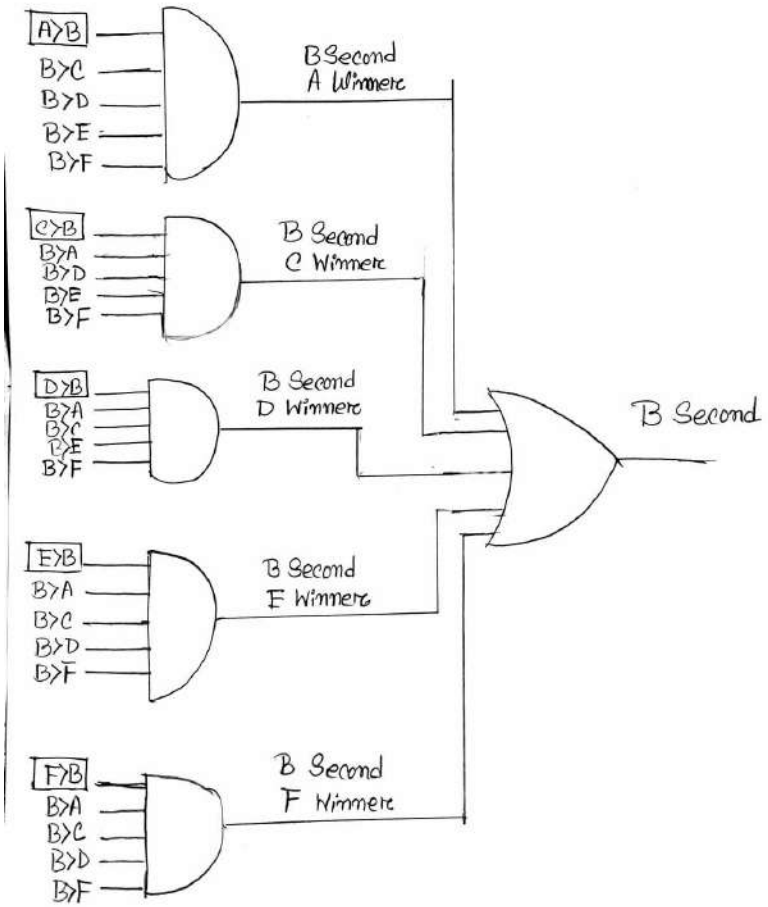
Similarly, In the sub-system where **B** wins(Vote Counter of B), the output value of the **8-bit counters** which calculate the vote got by **B** will go to the first half of all 5 **8-bit comparators.** And, the second half of the comparators will be connected with the output value of the **8-bit counters** which are calculating the votes of **A**, **C**, **D**, **E**, and **F**. So, when a situation like **B> A, C, D, E, F** arises, the participant **B** will be declared as the winner of the vote. Selecting winners of the **C, D, E, and F** also works in a similar process.

(P.T.O)

For the second position:

Now we want to make B second winner For this case, Vote count should be

A>B>(C,D,E,F) if all conditions are met then, B will be the Second winner in which A is winner.
C>B>(A,D,E,F) if all conditions are met then, B will be the Second winner in which C is winner
D>B>(A,C,E,F) if all conditions are met then, B will be the Second winner in which D is winner
E>B>(A,C,D,F) if all conditions are met then, B will be the Second winner in which E is winner
F>B>(A,C,D,E) if all conditions are met then, B will be the Second winner in which F is winner



Now we want to make C second. For this case, the Vote count should be,

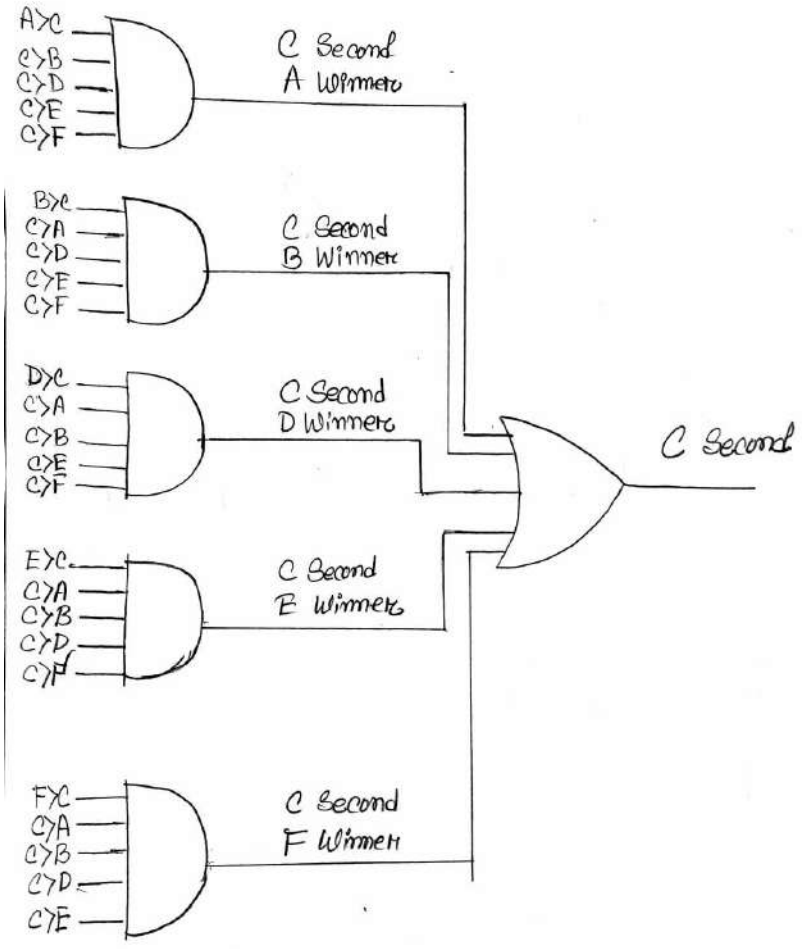
A>C>(B,D,E,F) if all conditions are met then, C will be the Second winner in which A is winner.

B>C>(A,D,E,F) if all conditions are met then, C will be the Second winner in which C is winner

D>C>(A,B,E,F) if all conditions are met then, C will be the Second winner in which D is winner

E>C>(A,B,D,F) if all conditions are met then, C will be the Second winner in which E is winner

F>C>(A,B,D,E) if all conditions are met then, C will be the Second winner in which F is winner



In similar way we have used to determine the second position for A, D, E and F.

Result/Interpretation with test cases: In our system we can always find who has selected as chairman. If there are draw cases, we still can find out the chairman. We can also select a vice chairman when there is no tie among the cast votes of the 6 participants. If there is any tie our system is unable to find the output. Our system can not find out who has been selected for the secretary position. Some test cases are given below.

CASE 01:

A = 17, B= 4, C = 13, D = 8, E = 10, F= 8

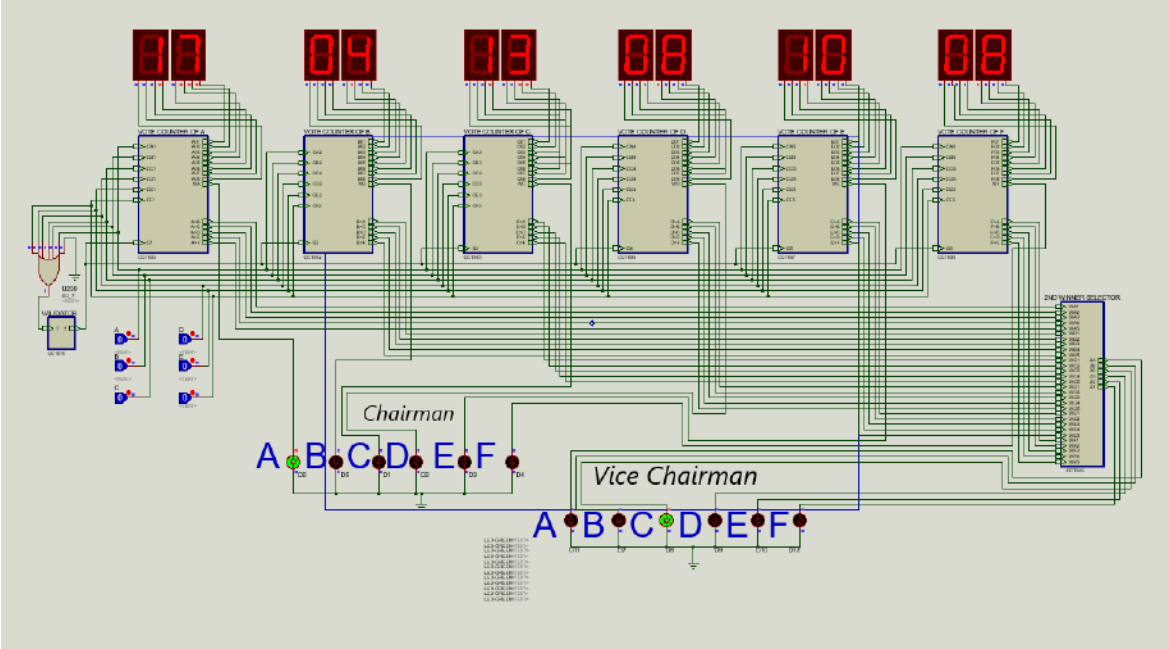


Fig: here the led labeled as A is blinking in the Chairman section, and C has been selected for Vice-chairman

CASE 02:

A = 3, B= 0, C = 17, D = 7, E = 10, F= 23

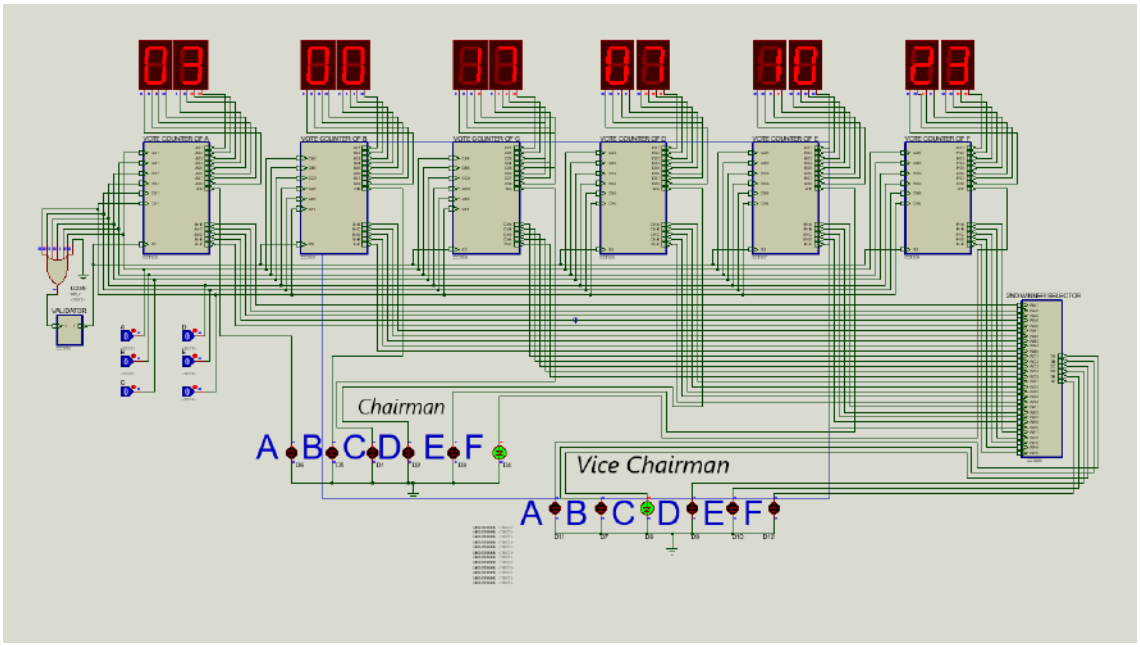


Fig: here the led labeled as F is blinking in the Chairman section, and C has been selected for Vice-chairman

case-3:

A = 1, B= 2, C = 17, D = 17, E = 8, F= 4

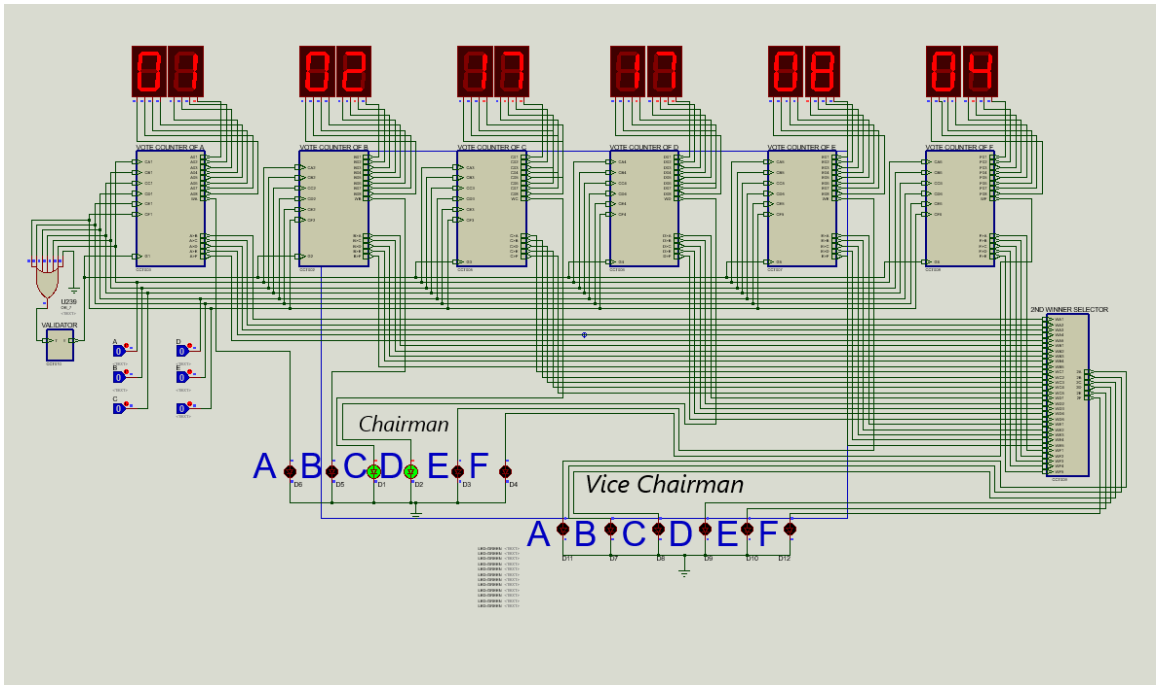


Fig: Since there is a tie, our system could find the champion (here both C and D), but could not find the vice precedent.

Discussion: For our project, we made sure to meet all the requirements laid out in the project description. It was a big learning experience for us because we hadn't worked with such large Binary Coded Decimal (BCD) numbers before. Figuring out how to link different Integrated Circuits (ICs) and making them give us the right results was also new. Dealing with the display system added another layer of challenge. We did achieve some significant results in the project, but there's one thing we couldn't figure out: how to choose the secretary position. Still, we believe what we learned here will be super helpful in the future. Overall, this project taught us a bunch of new stuff and will probably come in handy down the road.

Reference:

1. **BCD Display:**
    - https://www.electronicshub.org/bcd-7-segment-led-display-decoder-circuit/?fbclid=IwAR2Fj7Z4DEWa0RE57LSXywZWYlEsQ5cFI93LAWpkX54QV-X2kWDKFVs6r4w
    - https://www.electronics-tutorials.ws/blog/7-segment-display-tutorial.html

2. **7485 Comparator:**
    - https://youtu.be/wgje9eH3TT0?si=u87X5proXAOmx7Fq

3. **74190 Counter:**
    - https://www.watelectronics.com/bcd-counter-design-operation/?fbclid=IwAR3ygdgWa-Dm3RhSSQWx9RzpuyaAjQEh91z1bjrSZyg5KNjcuCcNCMnO5Gw

4. **AND gate:**
    - https://simply.science/popups/demorgan-mf.html

5. **OR gate:**
    - https://simply.science/popups/demorgan-mf.html