



REAL TIME DENSITY BASED TRAFFIC LIGHT CONTROLLER USING VERILOG & MICROCONTROLLER

A project for EEE 304 (Digital Electronics Laboratory)



BANGLADESH UNIVERSITY OF ENGINEERING AND TECHNOLOGY



Submitted by-

Md. Adnan Faisal Hossain (1606063)

Mohammad Ali Kazmi (1606061)

Mumtahina Islam Sukanya (1606039)

DECEMBER 21, 2020

One Drive Link for Demonstration Video & files:

https://buetedu-my.sharepoint.com/:f:/g/personal/1606063_eee_buet_ac_bd/EvmlqHiP8wZOiqRJf4giMu8Blw_bDg_sBu3AIB1OvIUAQw?e=bizyah

Table of Contents

I.	System Overview & Algorithms:	3
A.	Four-way traffic structure:	3
B.	State description:	4
(i)	Real time density feature:	4
(ii)	Allowing emergency vehicles to pass:	4
(iii)	Camera module to detect traffic violation:	5
(iv)	Priority Condition:	5
(v)	State Table:	6
(vi)	State Diagram:	7
II.	Quartus Simulation Results & Discussion	8
A.	Case Study 1:	8
B.	Case Study 2:	9
C.	Case Study 3:	10
D.	Case Study 4:	11
E.	Case Study 5:	12
III.	Real Time simulation on Proteus	13
A.	Components used from Proteus library in the simulation:	13
B.	Arduino Board description:	13
C.	Connections to the Arduino board:	13
D.	Description of the model circuit:	15
E.	Real time simulation examples:	16
IV.	Conclusion	22

List of Figures:

Figure 1: Overall Traffic System with Sensors	3
Figure 2: State Diagram of the Traffic Light Controller System	7
Figure 3: Quartus Simulation for first Case Study.....	8
Figure 4: Quartus Simulation for second Case Study.....	9
Figure 5: Quartus Simulation for third Case Study	10
Figure 6: Quartus Simulation for fourth Case Study.....	11
Figure 7: Quartus Simulation for fifth Case Study	12
Figure 8: I/O pin connections of the rest of the circuit to the Arduino board	14
Figure 9: Entire circuit model (Arduino board + traffic junction setup)	15
Figure 10: Proteus Simulation for Vehicle Density Detection System.....	16
Figure 11: Proteus Simulation for Priority Imposition System	17
Figure 12: Proteus Simulation for Emergency Vehicle Detection Feature.....	18
Figure 13: Proteus Simulation for Traffic Infringement Detection feature	19
Figure 14: Proteus Simulation for Yellow State Instant.....	20
Figure 15: Proteus Simulation for Clear Pin Functionalities Display	21

I. SYSTEM OVERVIEW & ALGORITHMS:

A. Four-way traffic structure:

Our traffic system is based on a 4-way junction. Each road has three sensors to detect vehicles on them which would indicate traffic pressure on that particular road. Sensors a1, a2, a3 are there to detect vehicle density on road A. These 3 sensors are placed at a distance of 15 feet, 30 feet and 45 feet from the stoppage line respectively. Similarly, we have sensors b1, b2, b3 for road B, c1, c2, c3 for road C and d1, d2, d3 for road D. We are expressing these 12 sensors as density tracking sensors. From these 12 sensors, it is possible to determine that how many vehicles are on each road and which road needs to be able to have moving vehicles before the other roads. There are also 4 sound sensors (ss1, ss2, ss3, ss4) on four roads to detect the presence of any emergency vehicle. In an event of any emergency vehicle being present in any of the four roads, their particular sound sensor will detect their sound and pass that information along. Also, there are 4 sensors (rc1, rc2, rc3, rc4) to detect vehicles which are violating traffic rules. When any of these sensors are activated, they will trigger a camera module which will eventually capture pictures of the corresponding vehicle which is violating traffic rule by crossing permissible stopping line.

The overall traffic system can be observed on figure 1.

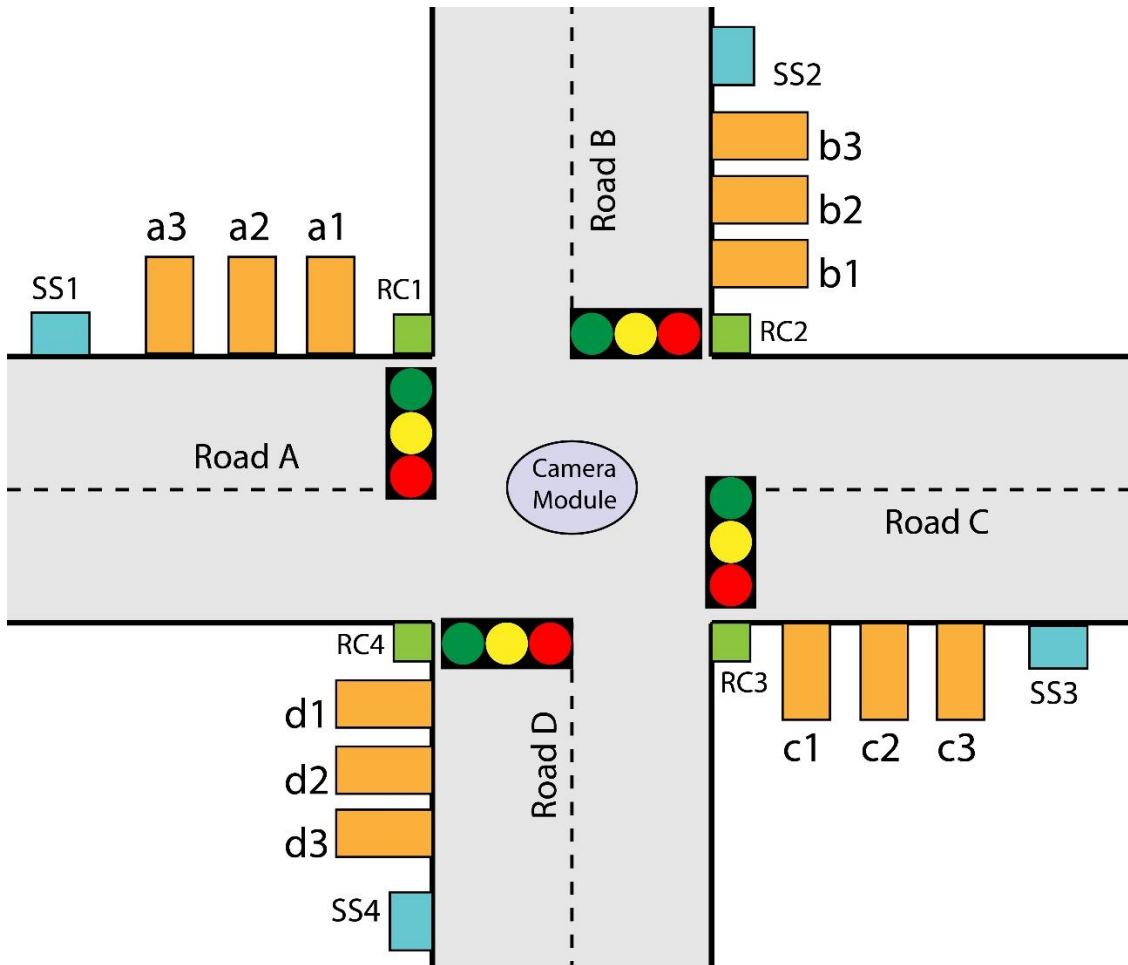


Figure 1: Overall Traffic System with Sensors

B. State description:

Our traffic controller system has a total of 27 states. The system starts or resets from s0 state. After that, it checks the outputs of density tracking sensors & sound sensors. If there is any emergency vehicle present on the junction, then sound sensor is activated & we move to state s22. Otherwise, density tracking sensors are activated & we move to state s1.

In state s1, we search for the road that has greater vehicle density. The maxoutput function is for this purpose. Our system moves to s2, s3, s4, s5 state depending on whether there is greater vehicle density on road A, B, C & D respectively. In all these 6 states, ID will be 100 100 100 100 which means that all of the roads will have red signal.

From state s2, the system can move to three different states, s6, s7 or s8 depending on the output of density tracking sensors. If there is a huge amount of traffic on road A, all the three sensors, a1, a2 & a3 will be active. Thus, we will move to s8. If there is a moderate amount of traffic on road A, only two sensors, a1 & a2 will be active and so we will move to s7. In case of slight amount of traffic on road A, only sensor a1 will be active. So, we will move to s6. Also, it is possible to move to s6 & s7 from s8 and to s6 from s7, if traffic is being cleared on road A. In these three states, ID will be 001 100 100 100 which means only road A will have green signal and the rest will have red signal.

Similarly, s10, s11, s12 are for road B, s14, s15, s16 are for road C and s18, s19, s20 are for road D. Similar logic applies for these states like the ones for road A. The reason for three different states for a single road is to realize the density of vehicles on each road.

(i) Real time density feature:

This intelligent traffic system is also a real time density-based system. Suppose at one moment, road A has green signal but traffic density in road C is greater than road A. So, our traffic system will be able to realize this & give green signal to road C turning road A's signal to yellow, then red.

For example, if we are in s6 state & only a1 is active. At the same time, c1 & c2 turns on. It means now we have greater vehicle density on road C. So, our system will move to s9, then s0, then s1, s4 and sequentially to s15. Thus, the green signal of road A will turn into yellow, then red, then road C will have green signal. This system ensures a proper management of traffic on all the four roads.

(ii) Allowing emergency vehicles to pass:

For emergency vehicles, s22 will be active. After that, the system will check for the road that has the emergency vehicle. Depending on the output of the sound sensors, we will move to s23, s24, s25 and s26 for presence of emergency vehicle on road A, B, C and D respectively. These states will be active as long as the emergency vehicle is present. After that, we will move to s9, s13, s17 or s21 for road A, B, C or D respectively depending on which road had the emergency vehicle on it.

Also, our system can move to s9 from s6, s7 or s8 if the corresponding sensor outputs are low. It means that the traffic on road A has been cleared and so road A will be given yellow signal, then red. This is the case for the other roads as well. In these four states, ID will be 010100100100, 100010100100, 100100010100, 100100100010 for states s9, s13, s17 and s21 respectively. In this way, the corresponding roads will have yellow signal for their corresponding state.

After that, the system will reset & move to s0 and the traffic cycle can start once again.

(iii) *Camera module to detect traffic violation:*

Our system has a total of four (rc1, rc2, rc3, rc4) violation detector sensors. These sensors are present in front of the stoppage line of each road. Thus, during a red signal, all of the vehicles should be behind the stoppage line and the sensor value should be low. But if any vehicle tries to violate the rule and move through, the sensor will be active and the camera module will turn on. The camera will capture the image of the vehicle and send it to traffic road authority for further investigation.

(iv) *Priority Condition:*

In case of equal traffic congestion or emergency vehicle on multiple roads, the priority would be: Road A > Road B > Road C > Road D

(v) *State Table:*

All the states for this system and their overall scenarios can be observed in table 1 as provided below:

Table 1: State conditions with sensor activation and traffic signal indication

STATE	TRIGGERING CONDITION	ID	ROAD 1	ROAD 2	ROAD 3	ROAD 4
S0	Starting condition	100 100 100 100	red	red	red	red
S1	a1 or b1 or c1 or d1	100 100 100 100	red	red	red	red
S2	Maxoutput = 1	100 100 100 100	red	red	red	red
S3	Maxoutput = 2	100 100 100 100	red	red	red	red
S4	Maxoutput = 3	100 100 100 100	red	red	red	red
S5	Maxoutput = 4	100 100 100 100	red	red	red	red
S6	Only a1	001 100 100 100	green	red	red	red
S7	Only a1 & a2	001 100 100 100	green	red	red	red
S8	a1, a2 & a3	001 100 100 100	green	red	red	red
S9	a1 = 0 or ss2, ss3, ss4 = 1	010 100 100 100	yellow	red	red	red
S10	Only b1	100 001 100 100	red	green	red	red
S11	Only b1 & b2	100 001 100 100	red	green	red	red
S12	b1, b2 & b3	100 001 100100	red	green	red	red
S13	b1 = 0 or ss1, ss3, ss4 = 1	100 010 100 100	red	yellow	red	red
S14	Only c1	100 100 001 100	red	red	green	red
S15	Only c1 & c2	100 100 001 100	red	red	green	red
S16	c1, c2 & c3	100 100 001 100	red	red	green	red
S17	c1 = 0 or ss1, ss2, ss4 = 1	100 100 010 100	red	red	yellow	red
S18	Only d1	100 100 100 001	red	red	red	green
S19	Only d1 & d2	100 100 100 001	red	red	red	green
S20	d1, d2 & d3	100 100 100 001	red	red	red	green
S21	d1 = 0 or ss1, ss2, ss3 = 1	100 100 100 010	red	red	red	yellow
S22	ss1 or ss2 or ss3 or ss4	100 100 100 100	red	red	red	red
S23	ss1 = 1	001 100 100 100	green	red	red	red
S24	ss2 = 1	100 001 100 100	red	green	red	red
S25	ss3 = 1	100 100 001 100	red	red	green	red
S26	ss4 = 1	100 100 100 001	red	red	red	green

Here, ID is a 12-bit output variable of our Verilog code which indicates the signal of all the four roads. The first 3 bits are for road A, second 3 bits are for road B and so on. Among the 3 bits, the first bit (LSB) indicates green signal, second bit indicates yellow and third bit (MSB) indicates red signal. So, a value of ID 001 100 100 100 means that road A has green signal, and the other roads have red signal.

(vi) *State Diagram:*

The whole network between the 26 states and how the transition from one state to other takes place can be found in figure 2.

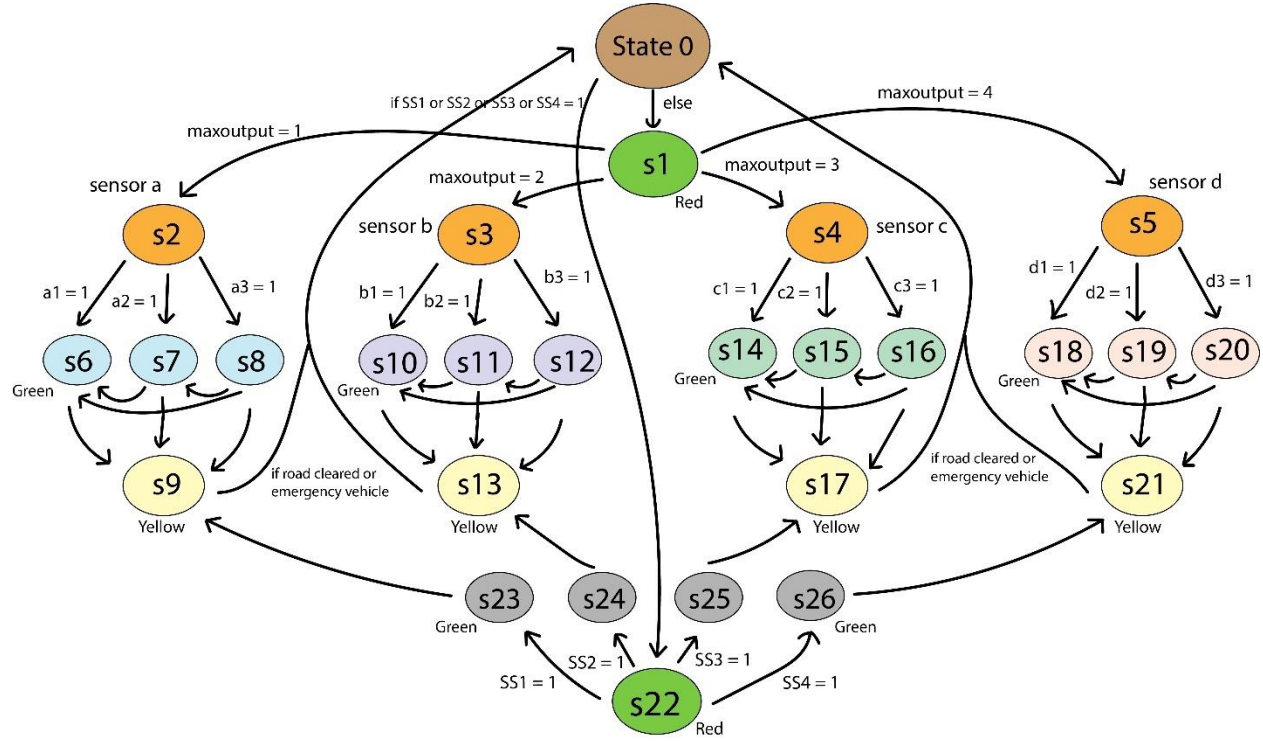


Figure 2: State Diagram of the Traffic Light Controller System

II. QUARTUS SIMULATION RESULTS & DISCUSSION

A. Case Study 1:

In this case, at first a1 sensor is activated. So, system moves to s1, then s2 and get green signal in Road A. When c1 & c2 sensors are both activated, Road A's yellow signal turns on and then goes red. Road C gets green signal as there's more traffic in road C now. After road C is cleared, sensor b1 is activated and so Road B gets green signal. Road B gets yellow signal, then red when traffic of road B is cleared.

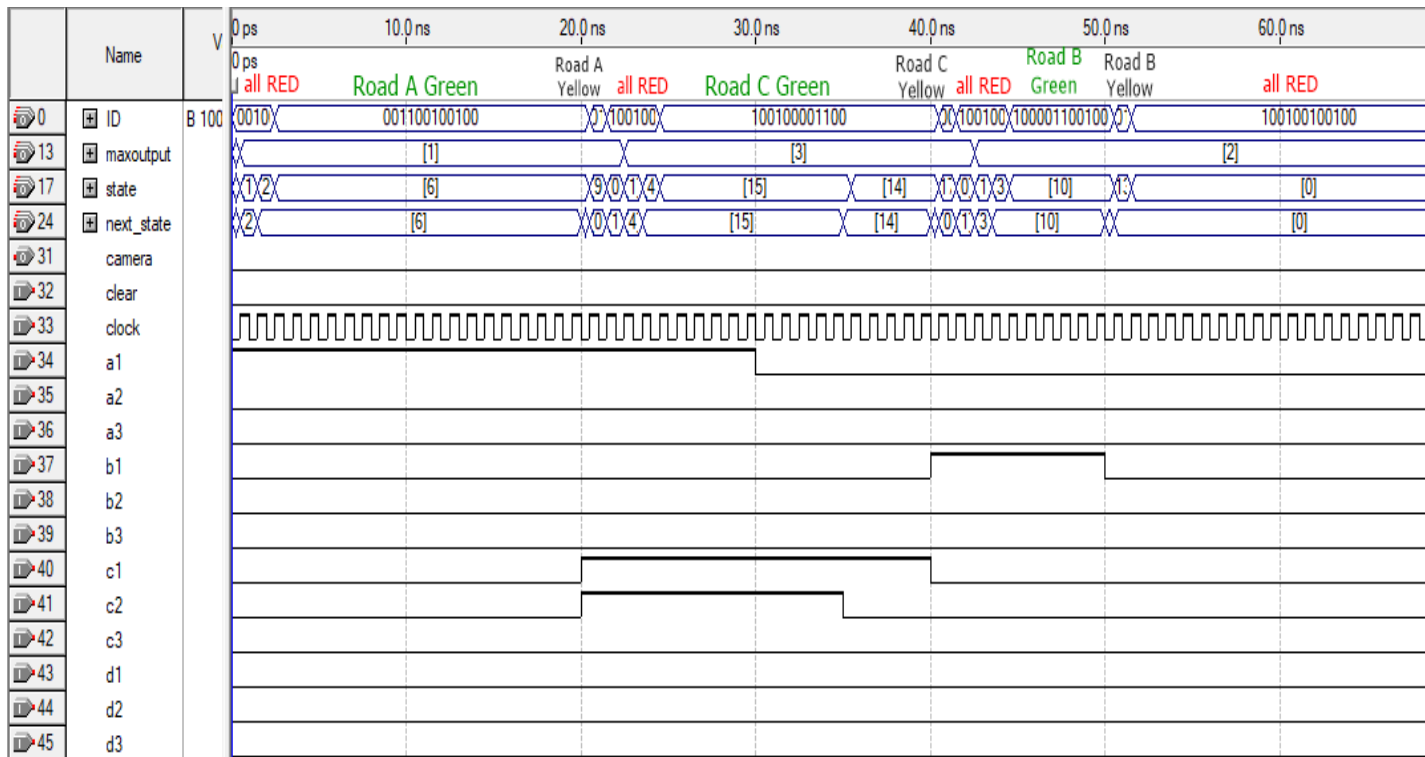


Figure 3: Quartus Simulation for first Case Study

B. Case Study 2:

In this case, at first sensor d1 is activated. So, road D gets green signal. After some time, b1, c1 and c2 are activated. But road C gets green signal as there is more traffic density on road C now. During this time, rc2 sensor is activated which means that a vehicle is violating the traffic rule on road B. So, our camera module is activated to catch the violator. After traffic is cleared on road C, sensor b1 is sensed once again and this time road B gets green signal.

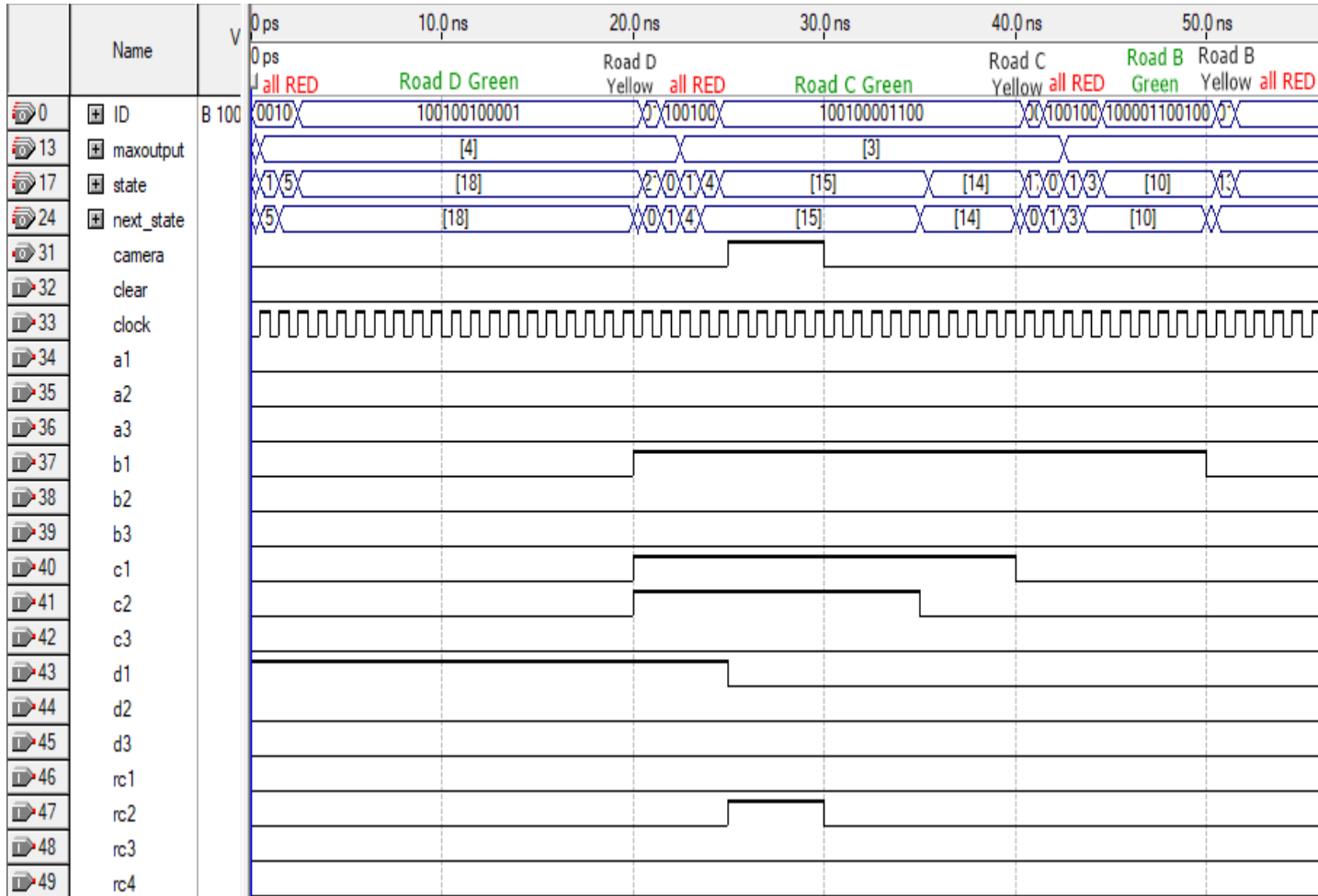


Figure 4: Quartus Simulation for second Case Study

In this case, at first sensor a1 & d1 both are activated. Because of the priority rule, road A gets green signal. After road A gets cleared, road D gets green signal. After that, sensor b1 is activated and road B gets green signal. Suddenly an emergency vehicle arrives on road C and so sensor ss3 is activated. Now, road B gets yellow signal, then red. Then, road C gets green signal and the system allows the emergency vehicle to pass. After that, road B gets green signal once again as sensor b1 is still activated.



D. Case Study 4:

In this case, at first sensors b1, b2, d1 are activated. As road B has more traffic, road B gets green signal. During this, rc4 is activated and so camera module is activated. After road B is cleared, road D gets green signal as d1 is still active. After that, road A gets green signal as a1 is active at that moment. During this, sensor ss2 is activated and so road B gets green signal to allow the emergency vehicle to pass. After that, road A gets green signal once again.

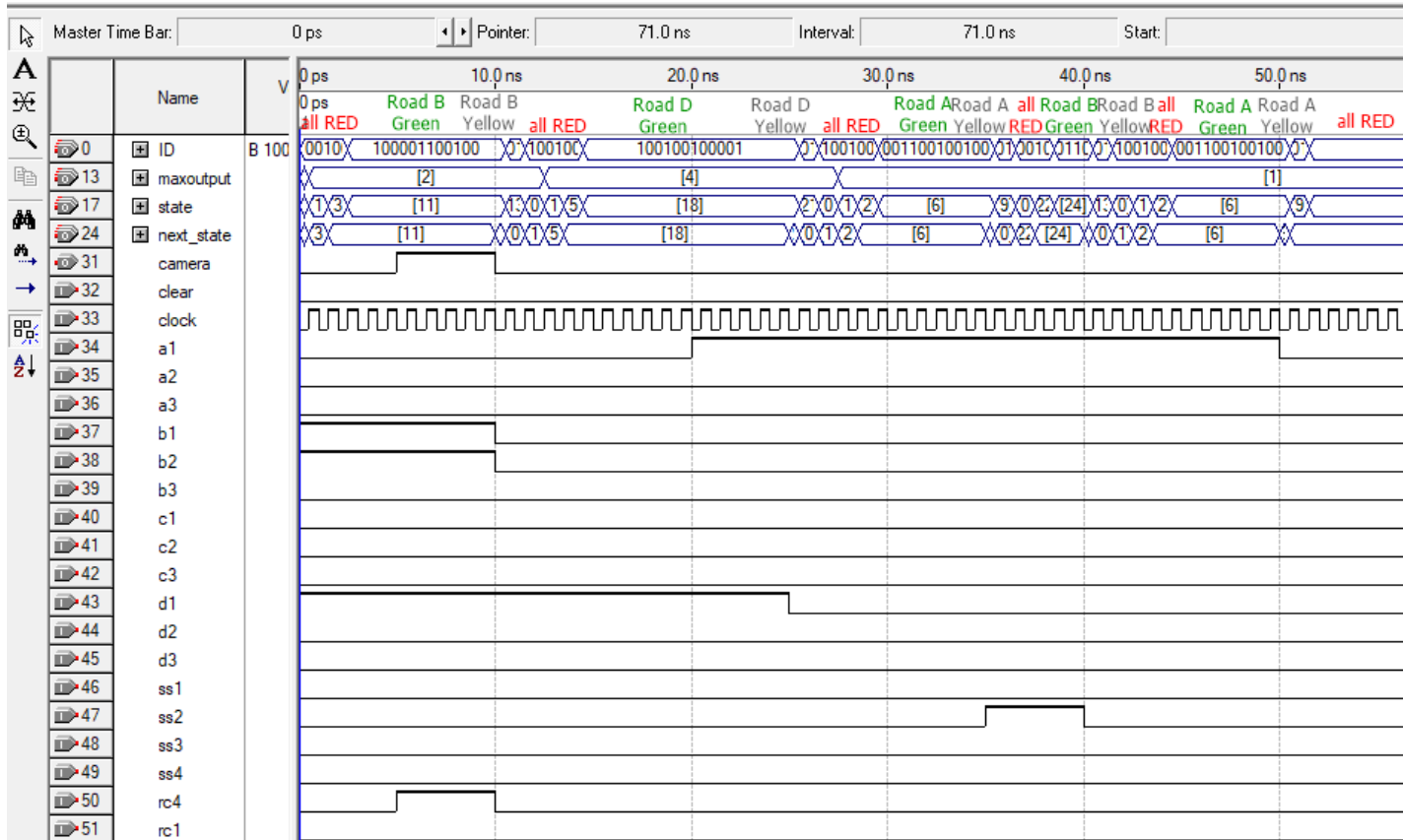


Figure 6: Quartus Simulation for fourth Case Study

E. Case Study 5:

In this case, at first d1 & d2 are active. So, road D gets green signal. After that, c1, c2 and c3 all are activated. It means that road C has a huge amount of traffic right now. So, road C gets green signal & road D gets yellow signal, then red. When rc1 is activated, camera module is activated. After road A is cleared, both a1 and d1 are activated. But road A gets green signal because of priority rule. After that, road D gets green signal. But an emergency vehicle appears on road B and so ss2 is activated. So, road B gets green signal. After the emergency vehicle passes, road D gets green signal once again.

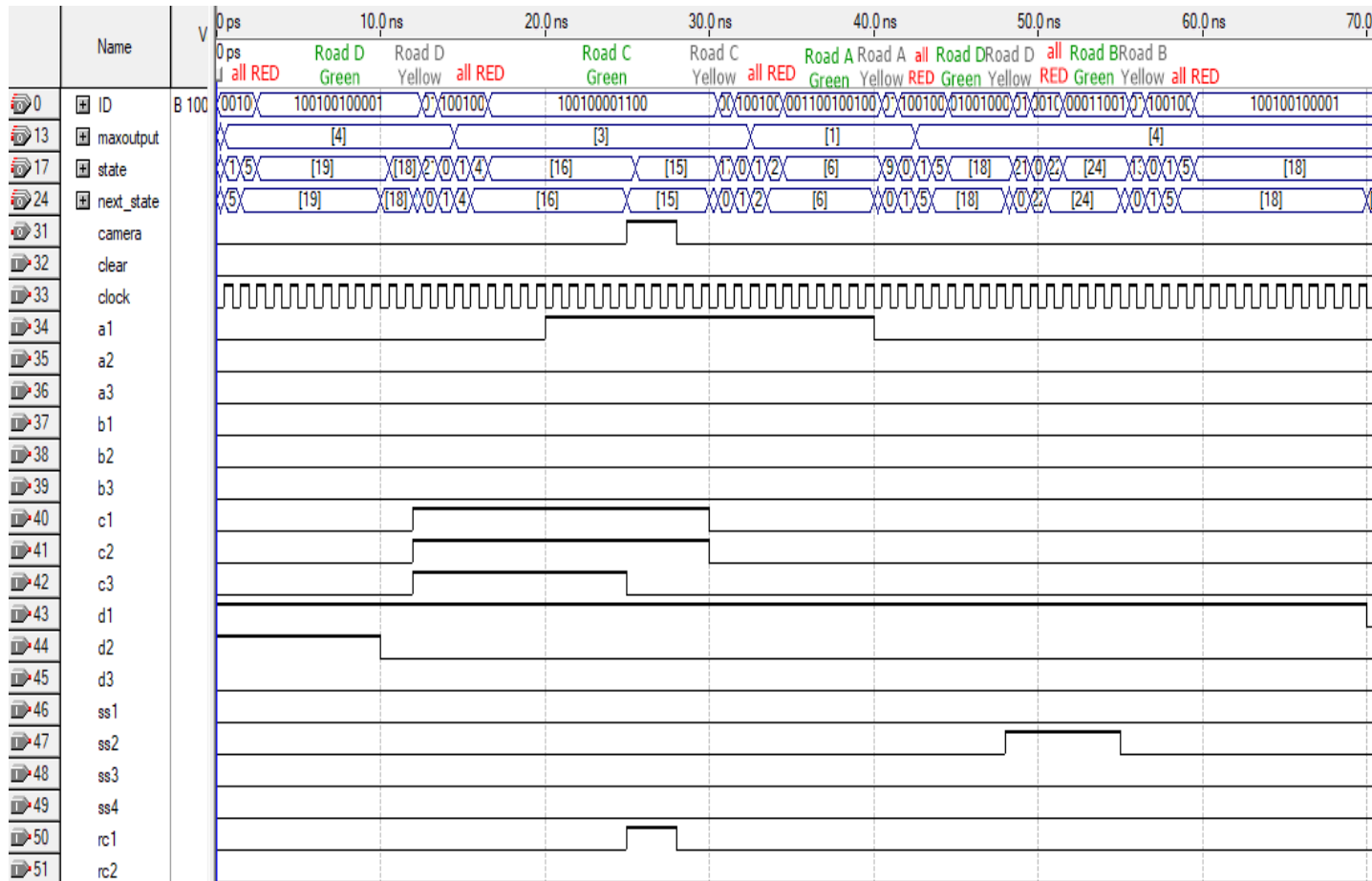


Figure 7: Quartus Simulation for fifth Case Study

Finally, we can say that all the examples shown above satisfy all the cases that can occur in our traffic system.

III. REAL TIME SIMULATION ON PROTEUS

To perform a real time analysis of our system, we simulate our Traffic Light Controller on proteus. We create an identical system using microcontroller to emulate our FPGA based proposed system. We do this because it is not possible to simulate an FPGA board on Proteus. The Arduino board we use is the Arduino ATmega 2560R3 board.

A. Components used from Proteus library in the simulation:

- (i) Arduino Mega 2560 R3 (external library downloaded from www.TheEngineeringprojects.com)
- (ii) Traffic Lights
- (iii) Logic states
- (iv) Logic probes

B. Arduino Board description:

The pin numbers of the board are as follows:

- (i) 16 Analog input pins (pin A0 – A15)
- (ii) 54 Digital input/output pins (pin 0 – 53)
- (iii) 5 V DC source pin (pin Vcc)
- (iv) Common ground pin (GND)
- (v) Reset pin

C. Connections to the Arduino board:

We only use the digital I/O pins for our purpose. Our system requires 21 input pins (pin 21 – 41) and 13 output pins (pin 0 – 12). The distribution of the pins are as follows:

(i) Input pins:

1. Traffic density sensors for road 1 (a1, a2, a3) are connected to digital I/O pins (22, 23, 24). Traffic density sensors for road 2 (b1, b2, b3) are connected to digital I/O pins (25, 26, 27). Traffic density sensors for road 3 (c1, c2, c3) are connected to digital I/O pins (28, 29, 30). Traffic density sensors for road 4 (d1, d2, d3) are connected to digital I/O pins (31, 32, 33).

2. Traffic violation detection sensors (rc1, rc2, rc3, rc4) are connected to digital I/O pins (34, 35, 36, 37).
3. Emergency vehicle detection sensors (ss1, ss2, ss3, ss4) are connected to digital I/O pins (38, 39, 40, 41).
4. Clear button to reset the system is connected to digital I/O pin 21.

(ii) Output pins:

1. Traffic lights for road 1 (R1, Y1, G1) are connected to digital I/O pins (0, 1, 2). Traffic lights for road 2 (R2, Y2, G2) are connected to digital I/O pins (3, 4, 5). Traffic lights for road 3 (R3, Y3, G3) are connected to digital I/O pins (6, 7, 8). Traffic lights for road 4 (R4, Y4, G4) are connected to digital I/O pins (9, 10, 11).
2. Camera output probe which denotes whether the camera is on/off (1/0) is connected to digital I/O pin 12.

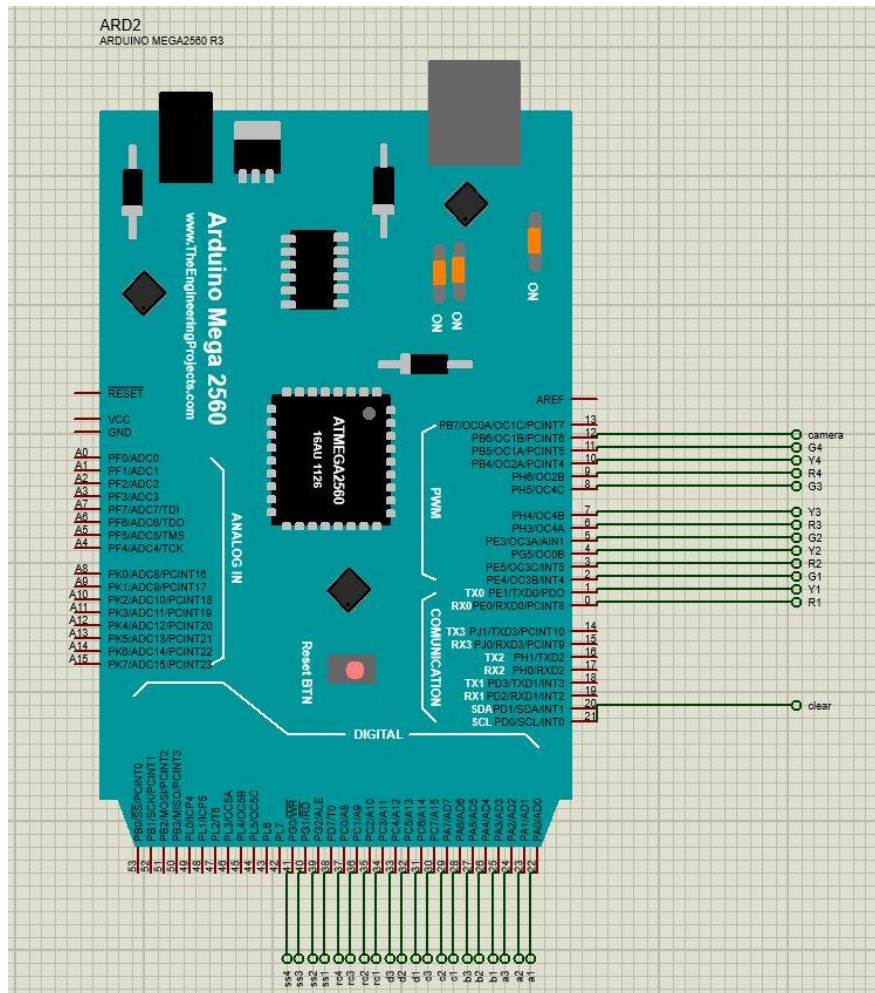


Figure 8: I/O pin connections of the rest of the circuit to the Arduino board

D. Description of the model circuit:

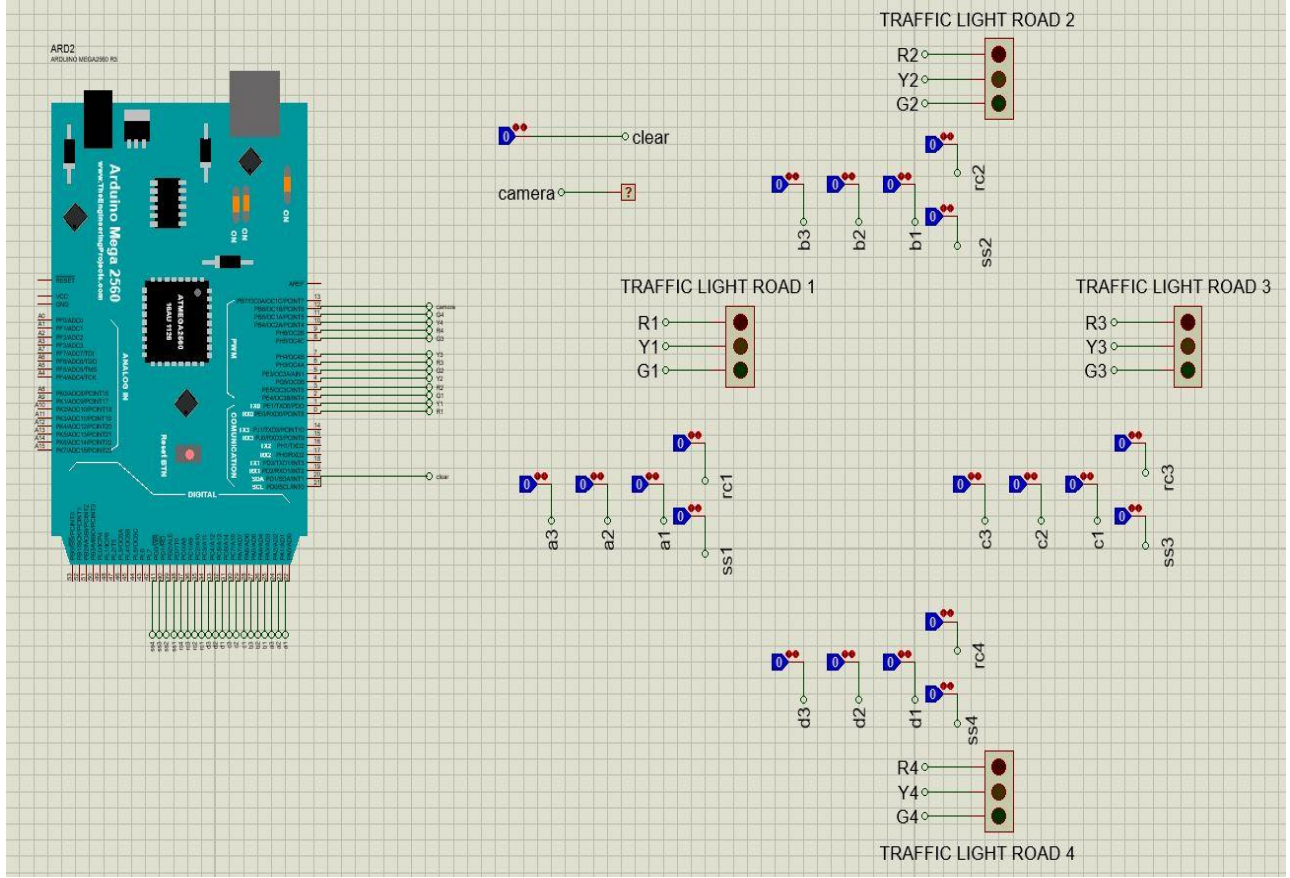


Figure 9: Entire circuit model (Arduino board + traffic junction setup)

We have modelled an intersection with four roads in the schematic from figure 4. There is a traffic light module (R, Y, G) for each road which will determine the state of that road at a certain point in time. Each road also contains a set of 5 sensors. For road 1 the sensors are traffic density sensors (a1, a2, a3), traffic infringement detection sensor rc1 and emergency vehicle detection sensor ss1. Roads 2, 3 and 4 also have similar sensors. To model a sensor, we have used logic states. In theory whenever a sensor detects traffic it is supposed to change state. This change in state needs to be detected. Therefore, we toggle the logic state between 1 and 0 to show the presence and absence of traffic respectively. We have also used a logic state to model the clear pin. This pin is used to reset the system and bring all the traffic lights to Red state irrespective of traffic density on any of the roads. Finally, we use a logic probe to see whether the camera output is 1 or 0 denoting whether the camera is active or inactive respectively.

E. Real time simulation examples:

(i) Example-1: Traffic density detection of the system

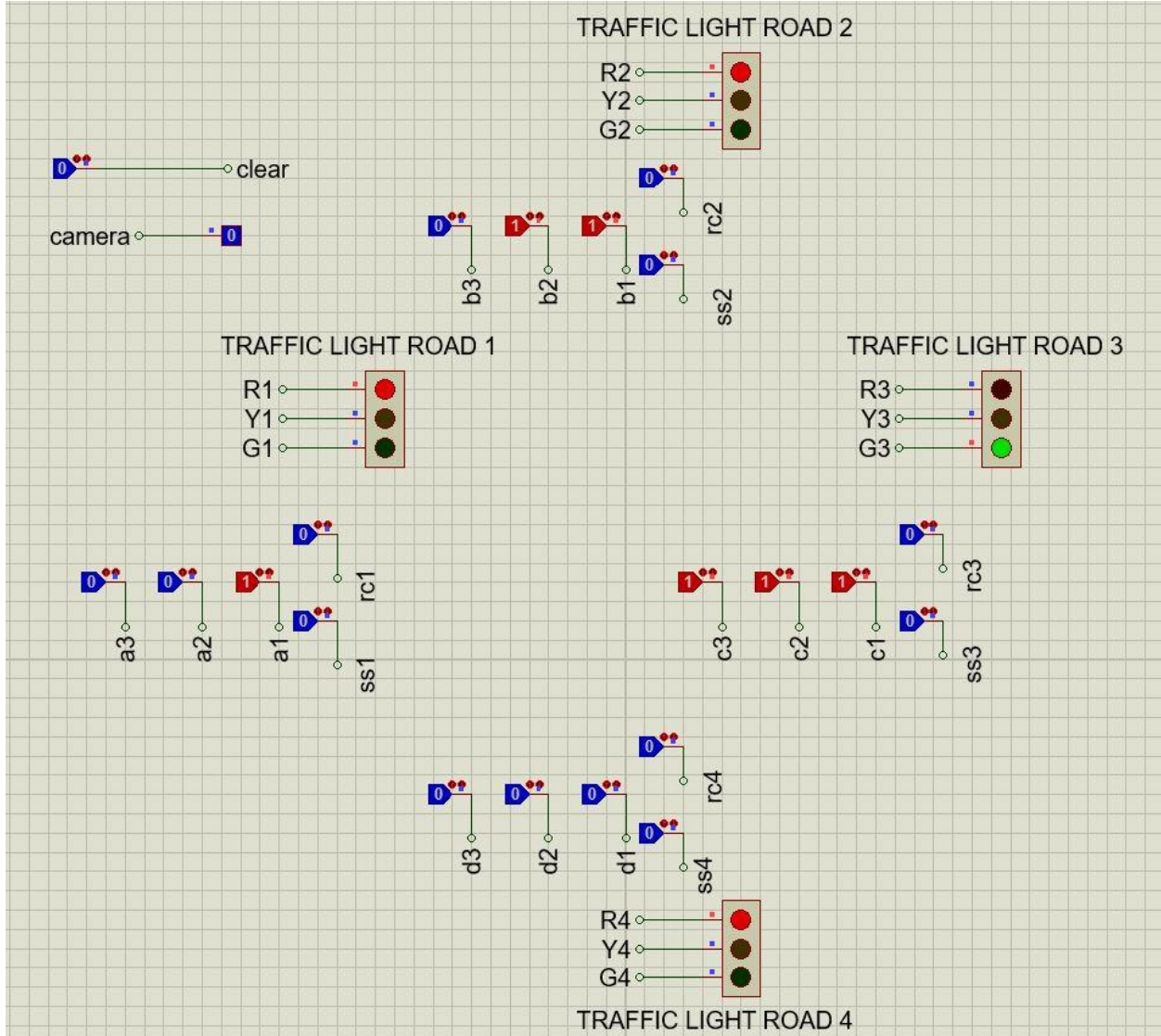


Figure 10: Proteus Simulation for Vehicle Density Detection System

In figure 5, we can see that the system detects traffic in Road 1, Road 2 and Road 3. However, the density of traffic is different for each road. Road 3 has the greatest number of traffic density detection sensors on ($c1 = c2 = c3 = 1$) compared to the other two roads. Hence the traffic light for Road 3 is GREEN and the rest are RED.

(ii) *Example-2: Priority of the different roads*

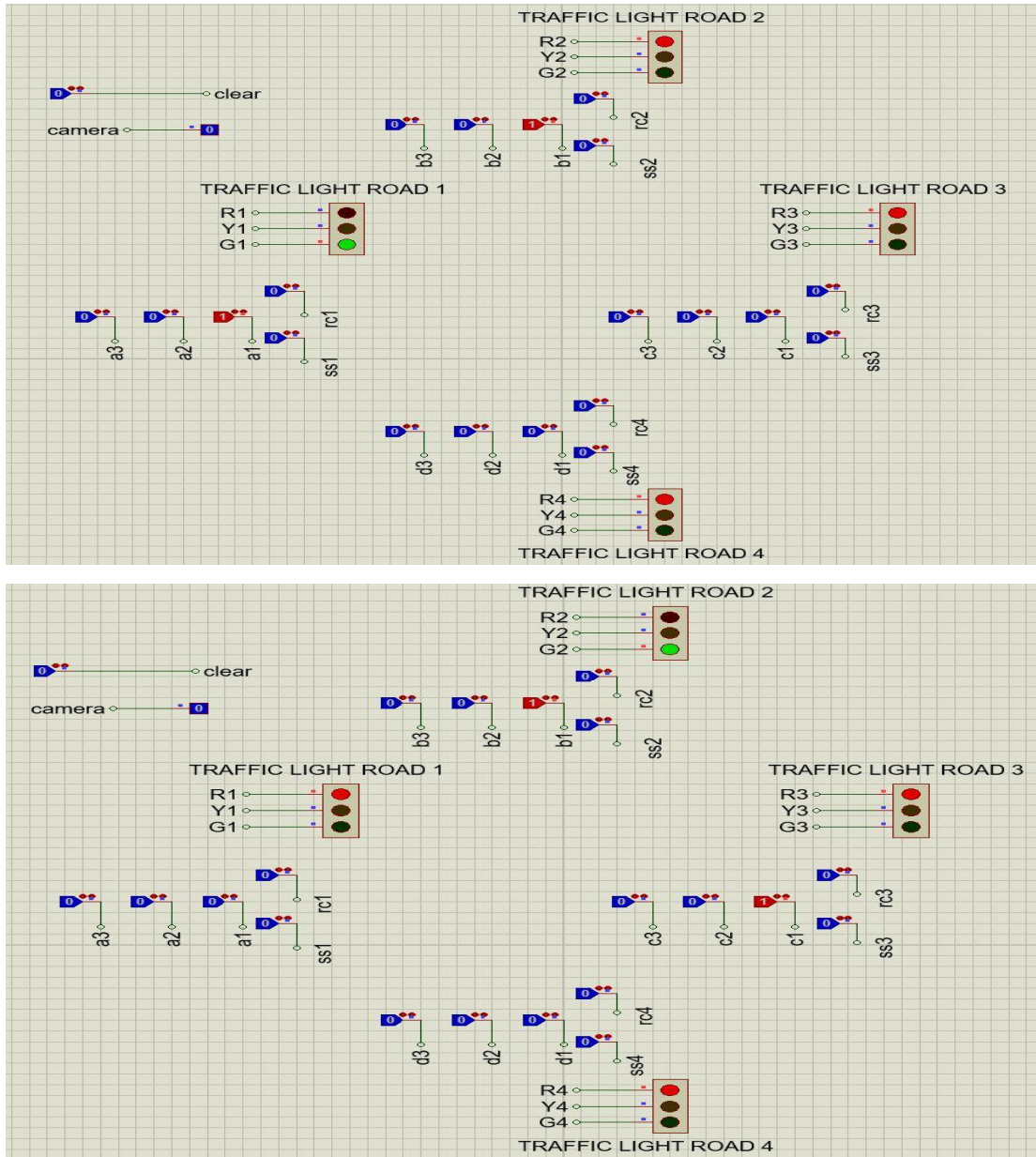


Figure 11: Proteus Simulation for Priority Imposition System

From the upper portion of figure 6, we can see that when both Road 1 and Road 2 simultaneously detect the same amount of traffic density, Road 1 will be prioritized and change to GREEN. Similarly, from the bottom portion, when both Road 2 and Road 3 simultaneously detect the same amount of traffic, Road 2 will be prioritized and turn to GREEN. Therefore, we can see the priority of our system ($1 > 2 > 3 > 4$) in this example.

(iii) *Example-3: Emergency vehicle detection*

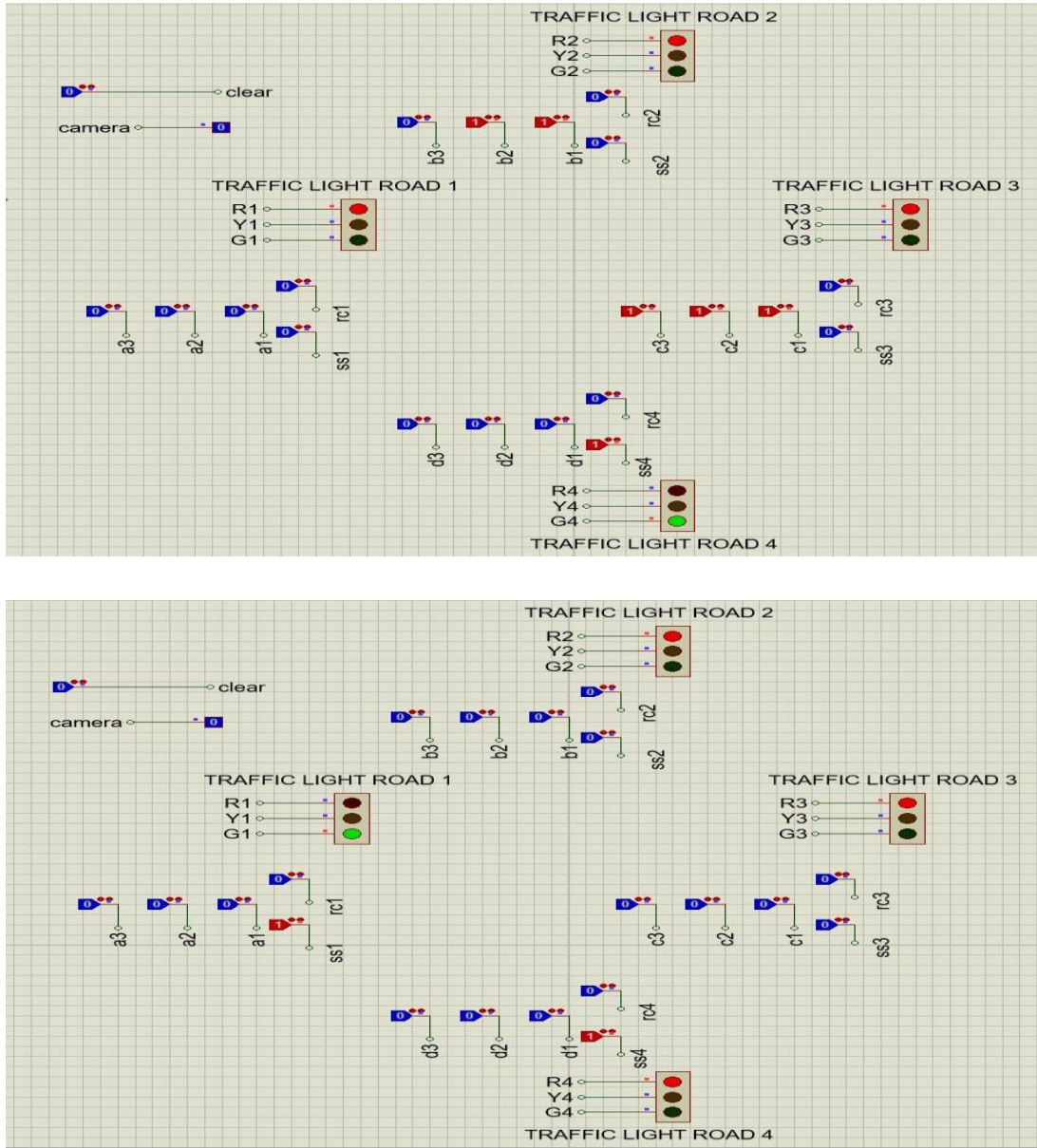


Figure 12: Proteus Simulation for Emergency Vehicle Detection Feature

In the first figure, we can see that b1 and b2 is on, and at the same time c1, c2 and c3 is on. This shows that currently Road 3 has the greatest traffic density, followed by Road 2 which has the second highest traffic density. Therefore, Road 3 should have been GREEN. However, the emergency vehicle detection sensor of Road 4, ss4 is on. Therefore, that Road will be given highest priority, and hence Road 4 is GREEN. From the second figure we can see that emergency vehicle detection sensors also have the same priority between roads ($1 > 2 > 3 > 4$). Therefore, although both ss1 and ss4 is on, Road 1 is GREEN and Road 4 is RED.

(iv) *Example-4: Traffic Infringement detection using camera module*

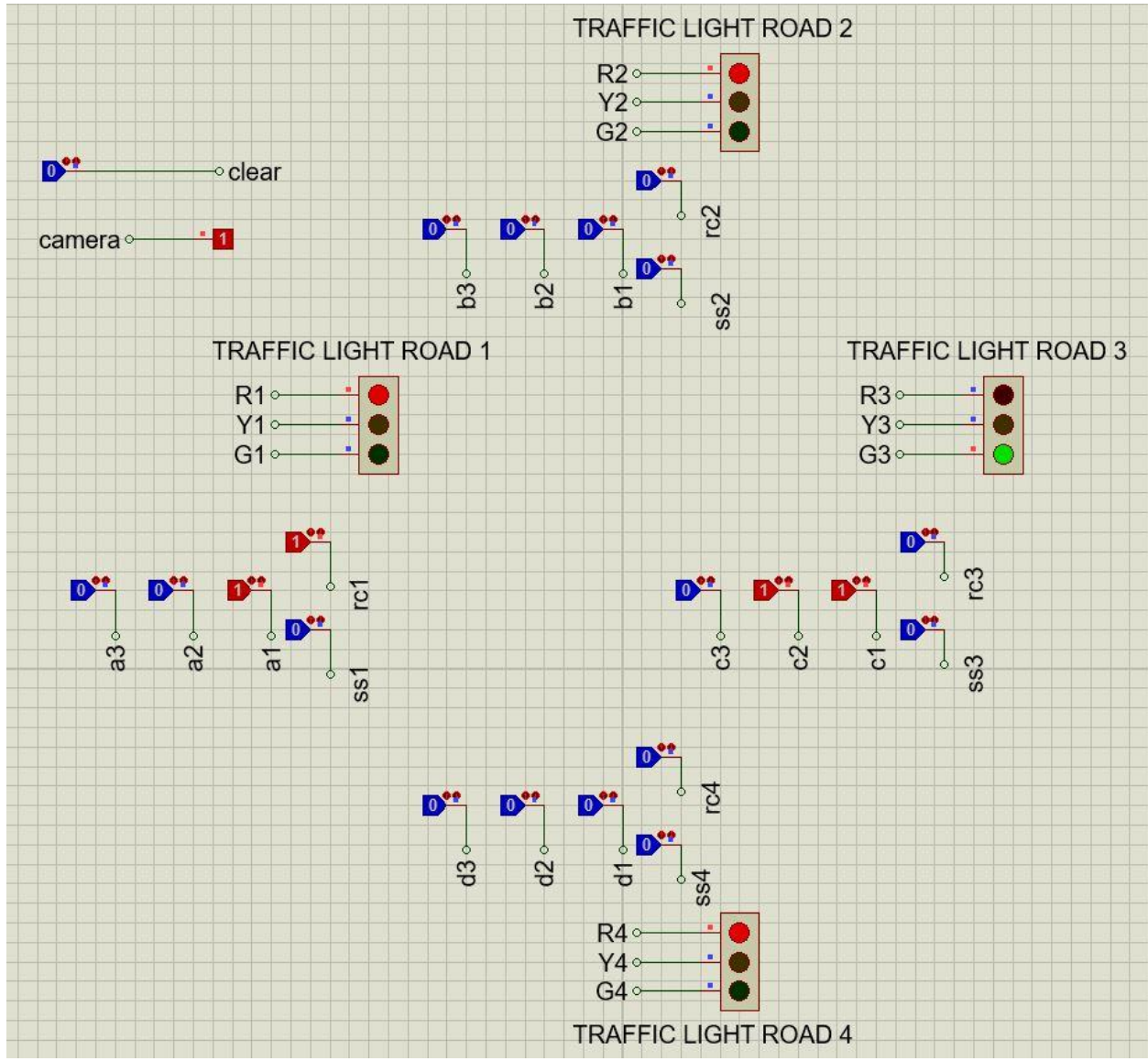


Figure 13: Proteus Simulation for Traffic Infringement Detection feature

From figure 8, we can see that Road 3 has the greatest traffic density. Hence the traffic light for Road 3 is GREEN. At the same time there is some traffic in Road 1 but the density of Road 1 traffic is less than Road 3. So, the signal for Road 1 is RED. We can see that the rc1 sensor is active. This means that while the signal for Road 1 was red, one of the vehicles from that road crossed the junction and triggered the sensor rc1. As rc1 is activated, the camera module is turned on (the camera logic probe shows 1 in the figure) and the picture of the vehicle that is causing the infringement can be taken.

(v) *Example-5: Transition from GREEN to RED (YELLOW) state*

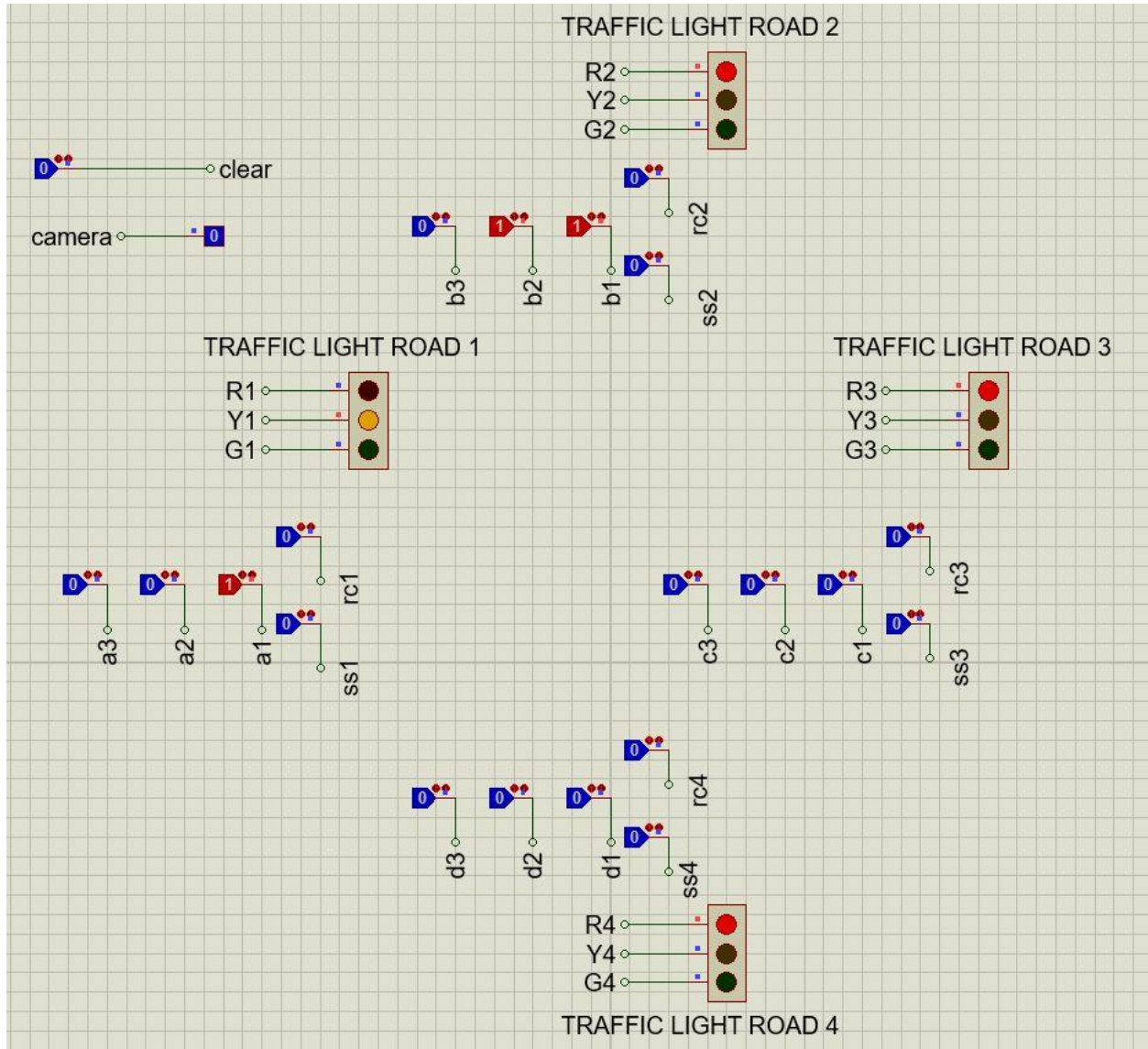


Figure 14: Proteus Simulation for Yellow State Instant

Here we can see the transition of Road 1 traffic light from GREEN to RED. The snapshot is of the instant when the signal is in the YELLOW state. The Road 1 signal changes to RED because the traffic density of Road 2 becomes greater than Road 1. So, the system will now set Road 2 to GREEN state.

(vi) *Example-6: Function of the clear pin*

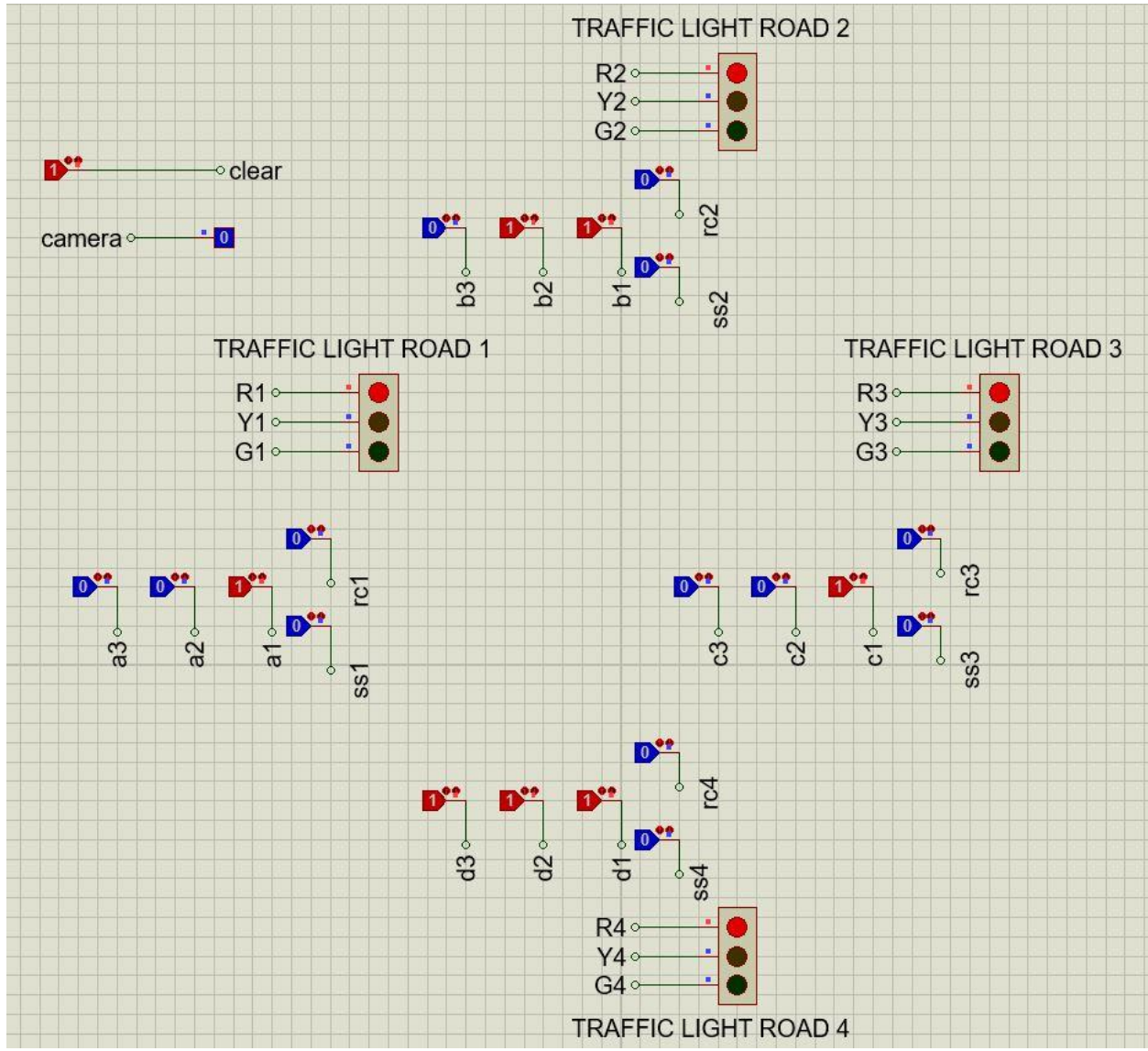


Figure 15: Proteus Simulation for Clear Pin Functionalities Display

The function of the clear pin is to bring the system to the original state (state 1). In this state, all the traffic lights are set to RED, irrespective of the traffic density on any of the roads. In the above figure, we can see that all the roads have some level of traffic on them, with Road 4 having the greatest traffic density. Hence normally, the signal for Road 4 should be GREEN. However, since the clear pin is on (set to 1 in the figure above) all the traffic lights including Road 4 is set to RED. This pin can be used to deal with faults in the system or to force emergency shutdowns when necessary.

IV. CONCLUSION

For an ending remark, it can be said that the intelligent traffic light control system is an innovative way to manage busy intersections with technology. The ability to make real-time decisions based on vehicle density on different roads will contribute by a large margin to reduce heavy traffic congestions. In addition to that identifying emergency vehicles in certain lanes and detecting traffic rules violation is an essential part of any functional traffic management system. In the existing system, since these managements are done by humans, the efficiency is less than what we can achieve from the intelligent system which relies on sensors.

In any economy or society, efficient traffic management system has a large impact. It can make the transportation system much smoother. It can also help citizens implement time management properly which from that individual level eventually affects the national economy greatly.

In conclusion, it can be stated that the intelligent traffic light control system proposed in this study can be much more efficient and energy-saving than the existing system. Hence implementing it in large metropolitan city junctions can prove to be resourceful.

CONTRIBUTION:

ID - 1606063

I created the Traffic Light Controller based on microcontroller using Arduino board and then performed the real time simulation of the TLC on proteus.