Model of Traffic Lights

Master Thesis

Study programme:	N2301 Mechanical Engineering
Study branch:	Manufacturing Systems and Processes
Author:	**Charankumar Mari**
Thesis Supervisors:	Ing. Radek Votrubec, Ph.D.
	Department of Manufacturing Systems and Automation

Liberec 2021

Master Thesis Assignment Form

Model of Traffic Lights

Name and surname:	**Charankumar Mari**
Identification number:	S18000456
Study programme:	N2301 Mechanical Engineering
Study branch:	Manufacturing Systems and Processes
Assigning department:	Department of Manufacturing Systems and Automation
Academic year:	**2019/2020**

Rules for Elaboration:

Traffic lights that are used for road repair are usually controlled independently and synchronized by time from Frankfurt am Main.

It is ussually only one pair.

The aim is to design control system for four traffic lights used on the crossroad. They have to communicate using wifi.

Student will create an education model too.

Scope of Graphic Work:	as required
Scope of Report:	50
Thesis Form:	printed/electronic
Thesis Language:	English

List of Specialised Literature:

[1] BEQUETTE, B. Process control: modeling, design, and simulation. Upper Saddle River, N.J. Prentice Hall PTR, 2003. ISBN 0133536408.
[2] Arduino Learning: Getting Started with Arduino. In: Arduino [online]. 2014 [cit. 2015-01-09]. available from: http://arduino.cc/en/Guide/HomePage

Thesis Supervisors:	Ing. Radek Votrubec, Ph.D.
	Department of Manufacturing Systems and Automation

Date of Thesis Assignment: November 20, 2019

Date of Thesis Submission: May 20, 2021

prof. Dr. Ing. Petr Lenfeld Ing.	Petr Zelený, Ph.D.
Dean	Head of Department

Liberec November 20, 2019

Declaration

I hereby certify, I, myself, have written my master thesis as an original and primary work using the literature listed below and consulting it with my thesis supervisor and my thesis counsellor.

I acknowledge that my bachelor master thesis is fully governed by Act No. 121/2000 Coll., the Copyright Act, in particular Article 60 – School Work.

I acknowledge that the Technical University of Liberec does not infringe my copyrights by using my master thesis for internal purposes of the Tech- nical University of Liberec.

I am aware of my obligation to inform the Technical University of Liberec on having used or granted license to use the results of my master thesis; in such a case the Technical University of Liberec may require reimbursement of the costs incurred for creating the result up to their actual amount.

At the same time, I honestly declare that the text of the printed version of my master thesis is identical with the text of the electronic version up- loaded into the IS/STAG.

I acknowledge that the Technical University of Liberec will make my master thesis public in accordance with paragraph 47b of Act No. 111/1998 Coll., on Higher Education Institutions and on Amendment to Other Acts (the Higher Education Act), as amended.

I am aware of the consequences which may under the Higher Education Act result from a breach of this declaration.

June 2, 2021   Charankumar Mari

ACKNOWLEDGEMENT

With immense pleasure and utmost gratitude, I acknowledge the support of my university for providing me this great opportunity to develop my engineering skills for accomplishing this diploma thesis and help me to further develop the knowledge on model of traffic light concept with the latest trends used in it.

I am indebted firstly to thank Ing. Radek Votrubec, Ph.D., whose expertise and wide knowledge in the field of study and his encouragement, professional guidance and his time all along the thesis to work better under his supervision.

I would like to thank our Head of the Department Ing. Petr Zelený Ph.D., for giving me this opportunity to perform this thesis and for his great support in every way to pursue our academics.

I gratefully thank my entire department staffs at the Technical university of Liberec for their immense support and insights on the various technologies required to complete my thesis. Finally, I am grateful to thank my parents and friends, for their unending help and support in all the situations and encouraging my works.

ABSTRACT

The aim of this diploma thesis is to design a control system for a traffic light. In this case, these are not fixed traffic lights at a light-controlled intersection, but temporary traffic signs at an ordinary intersection, for example during road repairs. It is necessary to install four portable traffic lights, two pairs, in this place. These traffic lights cannot be connected with cables. It is therefore proposed to control them using independent microprocessors communicating with each other via wifi. Various control strategies are proposed for this automaton-based system. A small intersection model was built to demonstrate the control system and for teaching purposes. The sample mast of the model was designed in CAD Catia and Creo and printed on a 3D printer. LEDs were used instead of lights. The proposed control system could be used at a real intersection, where it would be necessary to use amplifiers to connect powerful lights.

Keywords:
Model of wifi, Arduino, sequence of traffic light, Strategy of traffic light

ABSTRAKT

Cílem této diplomové práce je návrh řídicího systému světelné křižovatky. V tomto případě nejde o pevné semafory na světelně řízené křižovatce, ale o dočasné dopravní značení na obyčejné křižovatce například během opravy vozovky. Na toto místo je nutné nainstalovat čtyři přenosné semafory, dva páry. Tyto semafory nemohou být propojeny pomocí kabelů. Je proto navrženo jejich řízení pomocí nezávislých mikroprocesorů vzájemně komunikujících pomocí wifi. Jsou navrženy různé strategie řízení pro tento systém na bázi automatů. Pro demonstaci řídicího systému a pro účely výuky byl sestrojen malý model křižovatky. Ukázkový stožár modelu byl navržen v CAD Catia a Creo a vytištěn na 3D tiskárně. LED diody byly použity místo světel. Navržený řídicí systém by mohl být použit na reálné křižovatce, kdy by bylo nutné použít zesilovače pro připojení výkonných světel.

Klíčová slova
Model wifi, Arduino, sekvence semaforu, Strategie semaforu

Table of Contents

LIST OF FIGURES

1.INTRODUCTION

We simulate the traffic light system as in real life in this project. The first thing is to consider the issue very clearly. We will simulate a road crossing with two pairs of three LEDs acting as traffic lights. We created the education model. The control system consists of these components:

Arduino Traffic Light Controller Components,

1.Arduino Mega 2560.
2.Red LEDs
3.Yellow LEDs
4.Green LEDs
5.Buttons
6.Display

A traffic light controller system based in Arduino is developed in this project. It's a basic light system project that is used for traffic control 3 LEDs with three different colours, are used for transmission lights on this circuit. Arduino is indicated signal to the traffic light due to which we programmed in an Arduino. If Red signal is formed vehicle must be stopped, orange signal is formed its give ready to move of the vehicle due to the signal and the green signal is formed which direction we like to move on the direction. We have formed the designed of traffic light which we programmed in an Arduino for the timing calculation and the traffic light runs in an timing of our design. Apply to all types of different projects new understanding of Arduino LEDs and butts. Why not create a four way intersection, complete with several foot crossings and peat traffic lights if you are going to extend those lights many arduino boards can be easily programmed. One thing in particular is that Arduino boards are programmed by Arduino IDE. You can write and upload code using this program. Some boards are designed to be embedded and do not have hardware to buy for the programming interface. We are prepare a sequence and direction of the traffic light. There are two (or four) independent control systems synchronized via wifi. These are components of Internet of Things.

2. INDUSTRIAL REVOLUTION 4.0

With the arrival of mechanism, steam power and water power, the First industrial revolution arrived. The second industrial revolution followed, which revolved around electricity-based manufacturing and assembly lines. Electronics came into the third industrial revolution: I.T. Systems and automation leading to the fourth industrial revolution in cyber physical systems. This automation produces a production system by which machines in factories are increased by wireless networking and sensors to track, view and take autonomous decisions in an entire production chain. This digital technology can build virtual installation, process and application versions of real-world systems. These can then be evaluated robustly to make decentralized and cost-effective decisions. Through the internet these virtual copies can be generated and connected to allow cyber-physical systems to interact and to cooperate with the rest and the workers to create an integrated real-time data sharing and automation process for manufacture Industry 4.0. This automation involves interconnections between systems, clarity in information, and decentralized decision-making technology. This automation involves interconnections between systems, clarity in information, and decentralized decision-making technology.

Figure 1: Industry 4.0

In the first industrial revolution the manufacturing has switched from hand production method to machine production method by usage of water and steam. Different industries like textile industry, farming industry, mining industry, has affected by this change. Industry 2.0 is also referred to as the industrial revolution. Over this time, technological advancements have made it possible to significantly improve communications and transport. This revolution is referred to as the digital revolution. During this time, computers have evolved to become extremely influential in almost every industry and area of society. Industry 4.0 can mainly be implemented in intelligent factories. Industry 4.0 will, however, also contribute to intelligent ports with automatic cranes that can charge cargo on ships without human participation. The transport industry would also be highly influenced by increasing automation of vehicles, trains and aircraft, thus reducing the reliance on drivers, pilots and drivers.

2.1 SMART INDUSTRY

A smart industry is a highly digitalised and linked production facility focused on intelligent production. The idea of a smart industry is considered an important outcome of the fourth industrial revolution. Industry 4.0, considered to be the so-called factory of the future and is still in infancy. Used by manufacturing firms, an intelligent factory employs innovations such as artificial intelligence (AI), robotics, analytics, big data and the internet of things (IoT). Visibility, connectivity and autonomy are the distinguishing features of the clever plant. Industries have relied on automation for a long time, but intelligent factories take the idea much further and can operate without much human interference. Through using digital technology, intelligent manufacturing systems can learn and adapt in almost real-time or real-time, making plants much more versatile than in the past. The broad use of IoT sensors and devices binds machines and makes both their condition and their manufacturing process noticeable. Analytics and applications based on AI and machine learning are increasingly sophisticated and manage many of routine functions, allowing it freer to deal with exceptions and take decisions at higher level. Smart robots are supposed to work with people in routine factories.

Figure 2: Smart Industry

Intelligent plants rely on intelligent output, which links the plant to other businesses in the digital supply network and facilitates better supply chain management. They are also based on digital processing, which uses a Digital twin for digitally linking a commodity at all stages of its life. Furthermore, production machines can be used to accept orders, report progress, access to job instructions and communicate with quality and traceability systems in the local production system network. Staff in floors are able to see more quickly critical details such as orders, schedules, data quality, stock status and changes in demand.

2.2 BENEFITS OF SMART INDUSTRY

Link, sensor is crucial for links to computers, machines, and systems in order to provide information required to take real-time decisions in a smart plant, are the fundamental characteristics. Like clever home devices perform routine tasks such as dimming light at some time, or cause alarms when something is wrong, the ideal intelligent factory operating on a much bigger scale will self-correct when necessary and warn people if necessary. Furthermore, the extensive data volume offers real-time insight for stakeholders within the factory. The fact

that lower operating costs contribute to higher profits means that all producers want to minimize operational costs.

Industry 4.0 innovations allow this without staff being laid off and any drastic cost saving steps introduced. Help for quality management and real-time reporting eliminates waste. In many old-fashioned factories, waste is a major concern. Cost repairs and downtimes prevent predictive maintenance. Saving effort, increasing efficiency and ensuring quality is assured by automation. They all lead to a reduction in running costs.

Quality of manufacturers or company goods, facilities and processes are essential in today's industry to ensure affordability and economic sustainability. The advent of the fourth industry revolution implies developments in quality management through the use of intelligent electronics that are connected together in data networks, which without human interference will eventually be managed. The organization contributes a great deal to its final targets through a continuous and sustainable development climate. Intelligent technologies increase the efficiency, provide smooth visibility in bottlenecks and minimize human interference. This reduces human errors and redundancy of data. Managers can monitor their main performance indicators through the internet and smart devices with advanced production analysis and automated data collection strategies.

3. ARDUINO – MICROCONTROLLER BOARD

Arduino is an open-source electronic framework focused on easy-to-use hardware and software. Inputs will be read by Arduino boards light on a sensor, a finger on a button and converted into an output that activates a motor activates an LED, publishes something online. By sending a series of instructions to the microcontroller on the board, you can tell your board what to do. You use the Wiring-based Arduino programming language and Processing-based Arduino Software (IDE) to do so. Arduino has been used by thousands of various projects and applications. Arduino is a user software that is still versatile enough for advanced users. It will be running on Mac, Windows and Linux. And get started with programming and robotics. Designers and architects are creating immersive prototypes. And it is used to create many of]

the projects on display at the Maker Faire, for example. Arduino is a vital method for learning new things. Programmers can start tinkering only by following the step-by-step instructions in the package, or by exchanging ideas online with other members of the Arduino community. There are also other microcontrollers and platforms available for physical computing. Arduino also simplifies the method of operating with a microcontroller.

Arduino boards are relatively cheap compared to other microcontroller platforms. The least expensive variant of the Arduino module can be assembled by hand and even the pre-assembled Arduino versions of low cost. The Arduino idea is that we can directly upload the programs to the computer without the hardware programmer needing to burn the files.
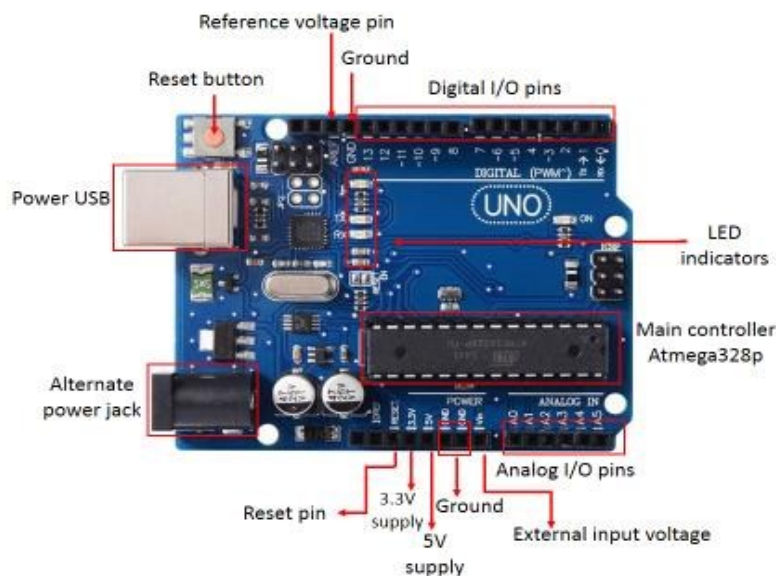

Figure 3: ARDUINO – MICROCONTROLLER BOARD

Open source platform accessible to extension is Arduino software programming. C++ libraries are available and if you want to understand technical specifics, you can safely switch from Arduino to the AVR programming language on which it is based. Windows, Macintosh OSX and Linux are the key source of Arduino applications. On Windows alone the rest of the microcontroller system runs.

3.1 ARDUINO MEGA 2560

The Arduino Mega 2560 is an AT mega 2560 microcontroller-based module. It has 54 digital input/output pins, including 14 PWM outputs, 16 analog inputs, 4 UARTS (serial hardware ports) and 16 MHz crystal oscillators, a USB interface, a power jack, an ICSP header and a reset key. It contains everything you need to support a microcontroller that has a simple connection. The Arduino super can be powered by an external power supply or a USB connection. Power sources will be chosen automatically. External power may be either an AC to DC converter or a battery that can be connected to the adapter by plugging a positive 2.1 mm centre plug into the power jack on the board. Battery lead can be inserted into the Gnd and Vin pin headers. The Board will run on an external supply of 6 to 20 volts. However, if it is supplied with less than 7V and the 5V pin is supplied with less than 5 volts and the board may be unstable. If the voltage regulator uses more than 12V, the system can be overheated and damaged. The range suggested is between 7 and 12 volts.
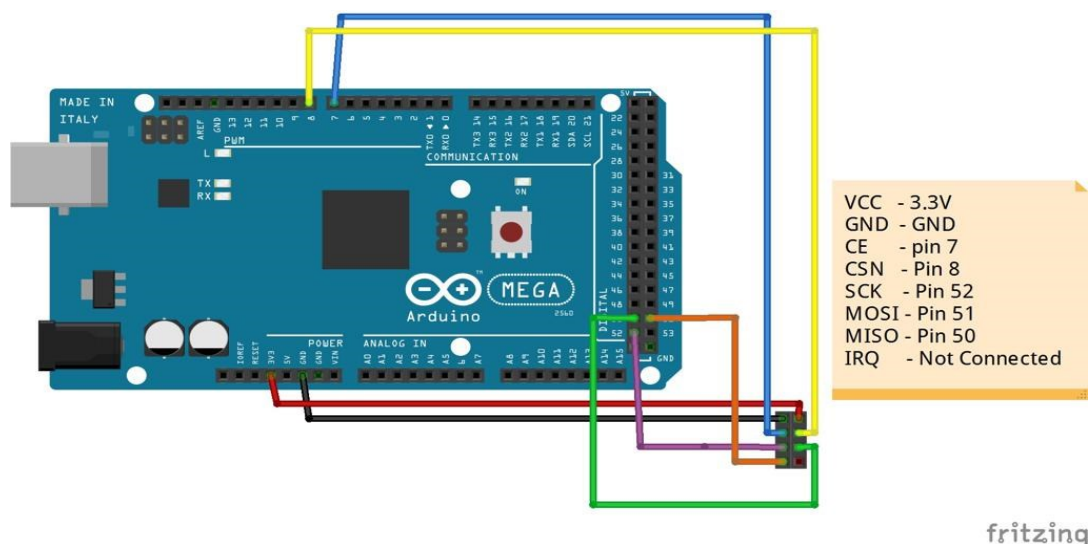

Figure 4:L ARDUINO MEGA 2560

The Arduino mega 2560 has 256 KB of flash memory for storing code, including 8 KB of SRAM (Static Random Access Memory) boot loader 8KB and 4KB of EEPROM boot loader 4KB (Electrically erasable programmable Read-only memory). The Arduino Mega 2560 has several communication facilities with a computer and any other Arduinos or other

microcontrollers. Arduino programming requires a serial monitor that allows simple text data to be transferred to and from the computer.

3.1.1 ARDUINO PIN CONFIGURATION

External interrupts are created using 6-pin interrupt 0(0), 1(3), 2(21), 3(20), 4(19) interrupt 5 Interrupt 5 (18). These pins generate interrupts in several ways, providing low value, rising or falling edge or changing the value of the interrupt pins. This Arduino board includes a pin-13-linked LED called a digital pin 13. Depending on the pin's high and low values, this LED can be worked. This would result in a real-time alteration of the programming skills.

Analog reference voltage (AREF) is the reference voltage for analog input. The board comprises 16 analog pins which are numbered as A0-A15. Knowing that all analog pins on this board can be used as digital I/O pins is very important. Each analog pin can be accessed with a 10-bit resolution that can be calculated from GND to 5 volts. However, the higher value can be modified using the AREF pin. The term SPI is a serial peripheral interface for data transmission between devices.

3.2 WI-FI MODULE

The Wi-Fi Module ESP8266 is a SOC that has its own TCP or IP stack that enables any microcontroller to access your Wi-Fi network. The ESP8266 will host or import all of the WiFi network functions from the application. The ESP8266 can be accessed via the local area network or via the internet. Using an Arduino, the USB to TTL converter configures the module with the receptor and transmitter serial pins (RX and TX). And communication for controlling the components can be programmed. And the Internet is easy to access and cheaper to communicate. The module is both a point of entry and a station to facilitate internet collection and uploading of data.

The transport force can be set as four rates between high and low, and an additional rate for external 3.3-volt power supplies must be recommended to be used so that the maximum power is no longer usable.
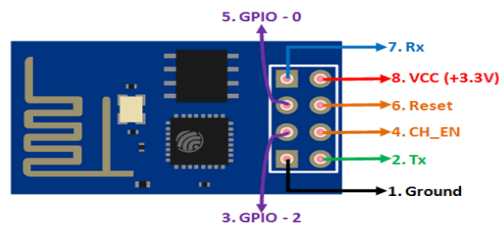

Figure 5: WI-FI MODULE

These wireless module take up less time for hundreds of thousands of amperes when importing and exporting the data and the additional 3, 3V voltage power plug for the easy communication should always be attached to 10 micro farad condensers. The radio waves can be broadcast and received. A Wireless Router receives and decodes the signal. The router moves it to the next board. It is possible to transmit up to 2.4 GHz to the GHz spectrum. More data can relay the signal thanks to the higher frequency. It consists of 8 pins with an Arduino attachment. Analog and digital power supply operating voltage is between 1.3V and 3.3V. If there is a high power supply when the analog signals are transmitted it could be impaired. Below you can find the Wi-Fi pin setup diagram.

3.3. LCD SCREEN

LCD is a flat panel display style that uses liquid crystals to act as a primary mode. LEDs have a wide and diverse variety of applications as they can be used in smartphones, computer monitors and instrument panels. The technologies they replaced, which involve an LED diode and a gas plasma display was a big leap. LCDs were a big leap. LCDs also allowed much more thinning displays than the technology of the cathode tube. LCDs use much less power than LEDs and gas displays because instead of emitting, they operate according to the concept of light blocking. Where an LED emits light, the fluid crystals in an LCD generate a backlight image. The first pin from left to right is the ground pin. It has 16 pins. The second pin is the

VCC that attaches the 5-volt pin to the Arduino frame. Next is the Vo pin that can be added to a potentiometer to monitor the contrast of the display.
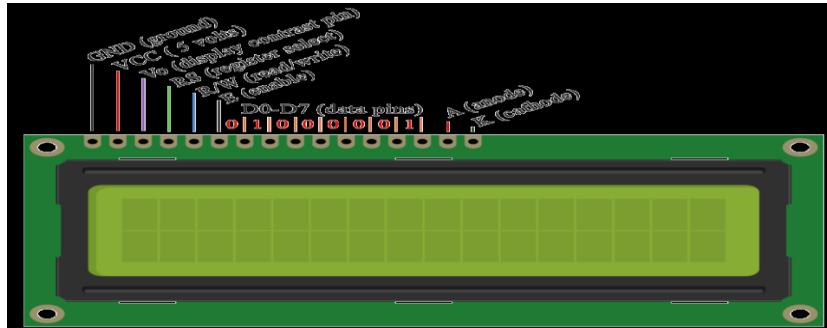

Figure 6: LCD SCREEN

The RS pin register will be used to choose whether commands or data should be sent to the LCD. For instance, if the RS pin is set at low or zero volts, the LCD is sent with the following commands set the cursor to a particular position remove the display, turn off the display. And we send data or features to LCD when the RS pin is set to high or 5 Volt. The R or W pin that chooses to read or write to the LCD in the mode. The type mode is clear here and used to write or send LCD commands and data. The pin E allowed for register writing and next eight D0 to D7 data pins. Then we send 8-bit data via these pins when we write to the registers or if we want to see the latter case A on our display, for example, 0100 0001 is shown in the registers under the ASCII Row. And for an LED back light the last two pins A and K are anode and cathode.

3.4. BUTTONS

There are only two links for a pushbutton, although it has four legs. When the button has been pressed, all the legs are linked. If not the top two legs are connected and the bottom two legs are connected, but between the top and the bottom is not connected. The VCC or GND pin of a single button is associated. An Arduino pin has the other pin. We can detect the button being pressed or NOT by reading the state of the Arduino pin. It depends on how the button is attached to Arduino and what the Arduino pin is set to be between the button state and pressing status. The one pin is connected to VCC, while the other is connected with a pull-down resistor

Arduino's pin. The pin state of Arduino is high when you push the button. If not the pin status of Arduino is LOW. The external resistor is needed.
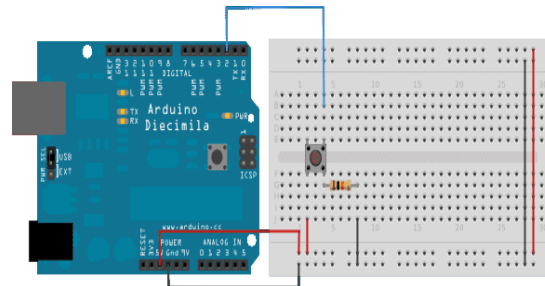

Figure 7: BUTTONS

4. AUTOMATONS

Automatons are abstract machine models that measure the input by traveling over a series of configuration statuses. The next configuration is determined on the basis of the finite part of the present configuration at each computing state. Automation itself is an electronic method for the processing of various procedures that is closely related to automatic. In the case of basic machinery called automation, plainly mentioned automation theory deals with calculation logic.

Through this automated method we can understand how the system measures the functions and effectively solves the problems. Function to be defined as computable or as derisible. This is the system's abstract model. The input is computed by shifting a number of states or settings. The next configuration is determined on the basis of the finite part of the present configuration at each computing state. As a result, after a specified configuration of the computation, the data is acknowledged. And for cars there are several representations, which means circle represents for countries such as S1, S2, S3, S4 and ellipse for situations such as hearing press and communications servers and the box represents for acts such as the ON device and OFF system timers. The job is divided into several pieces by these diagrams, and the words for the Many

states regulate workflow. This is how the conditions are broken. It becomes easier to program and solve solutions.

There are three sections of these diagrams.

1) States
2) Conditions
3) Action

4.1 STATES

It is an automaton's finite state. It reflects the stage at which the software flows. This is essentially demonstrated by circles and the work is carried out during execution from State to State.

4.2 CONDITIONS

The conditions in every state are verified and the workflow is determined. Continue. In every automaker state, there are several conditions tested. If one of the criteria is met, the program continues to the next state.

4.3 ACTIONS

This is the job done as the diagram flows. If a condition is present Satisfied, work runs into the next step in a state. But the work performed in the method is called operation before going to the next level. These activities cover all tasks, beginning with the switching of an LED diode to the complex task of transmitting or receiving signals.

Figure 8: Components of automaton
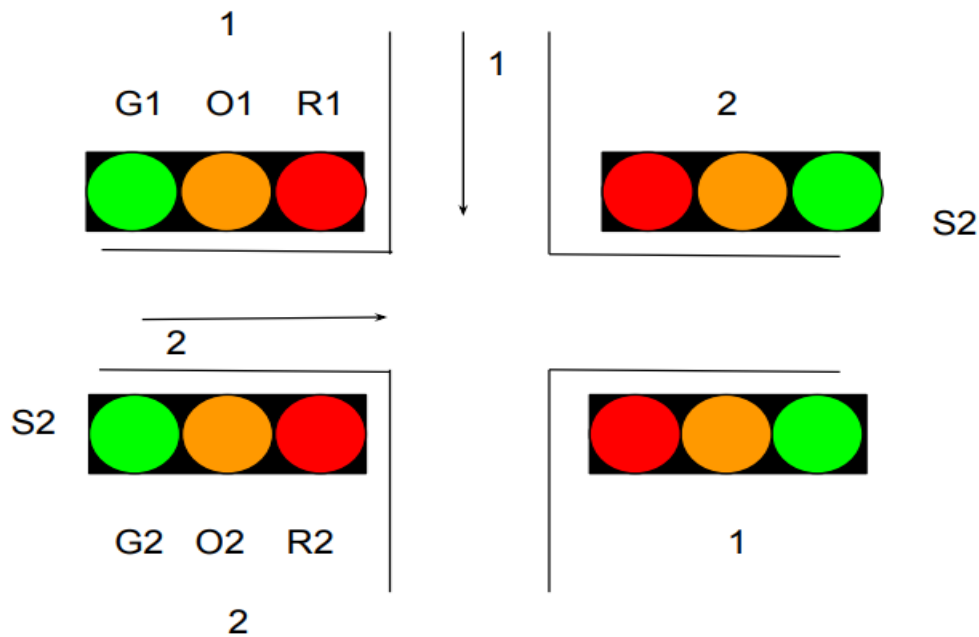
5. LIGHTING OF SIGNAL

Figure 9: LIGHTING OF SIGNAL

There are four ways in this area and four signals are there we are going to indicated the traffic instances.

In most parts of Europe, traffic signals in the stop lines are on the same side as the approaching traffic and are mostly placed overhead and on the side of the street. The traffic lights installed on the far side of the intersection in Spain are intended for traffic exiting the intersection in that direction. This is mostly done due to pedestrian crossings, so that traffic needs to pause if they

get a red light. These intersections also have a stop line in the intersection's exit area. In The Netherlands, a large number of road signals that are networked can be seen from the side by means of a small bulbous window, signalling to the drivers whether or not the signal in a crossing direction is actually red.

European approach to a marked transition is the dual and, more seldom, triple feature of a

pictogram pedestrian with a blackened out lens. The same technique is used by cyclists and the lens is blackened for a bike frame. Lenses with both symbols on them are not rare to see. For

the middle sun, most European countries use orange rather than amber.

The green colouring on highway exit signs and directional signs will always be used when it's simply necessary to inform the driver about an upcoming convenience. Like all the other sign colours, green signs are meant to not be distracting, and to inform the driver rather than disturb or alert the person.

Red signal

When the light passes through atmosphere, it gets scattered by small air particles. The scattering of light is inversely proportional to the fourth power of its wavelength. As the red light has the highest wavelength among all the colours, so it is scattered the least.

Orange signal

In general, when a traffic light turns orange, you have to start slowing down to stop. But if you cannot do it in time, you can drive through orange light if it is hazardous to stop otherwise. Red light cameras often sit on the top of traffic signals.

Green light

A green portion traffic sign lit to indicate the safe operation: Before crossing the street wait for the green light. Green light allows you to do something as well. The developers had a green light to continue the project.

5.1 EUROPE SIGNAL

The European approach to the signalised crossing is double or, occasionally, triple. The solution for cyclists is the same, with the lens for a bicycle frame blackened. Lenses with both symbols on them are not unusual. For the medium sun, most European states use orange rather

than amber. A mixture of red and orange lights is illuminated just before the green process in

Germany, the Czech Republic and some other Central European nations. Change of traffic signals delay depending on the number of vehicles passed through a portion of the road allocated, is the model's working concept. There are 4 sensors on four sides of four-way lane, which count the number of vehicles going through the sensor area. In the current circumstances, a very economical and efficient traffic management system is essential in our country, and every day there are more road accidents in India. In order to address this traffic congestion and unnecessary delay, we build a compatible device. In realistic terms the use of that technology will effectively reduce the chaos of traffic by allocating the time slots with the required delays, supporting the advantage of the vehicle load and multi-junctional lanes.

1. Connect the LEDs in the order as Red, Green, and Orange in the breadboard.

2. Place the negative terminal of the LEDs in common and connect the 220ohm resistor in series.

3. Connect the connector wires accordingly.

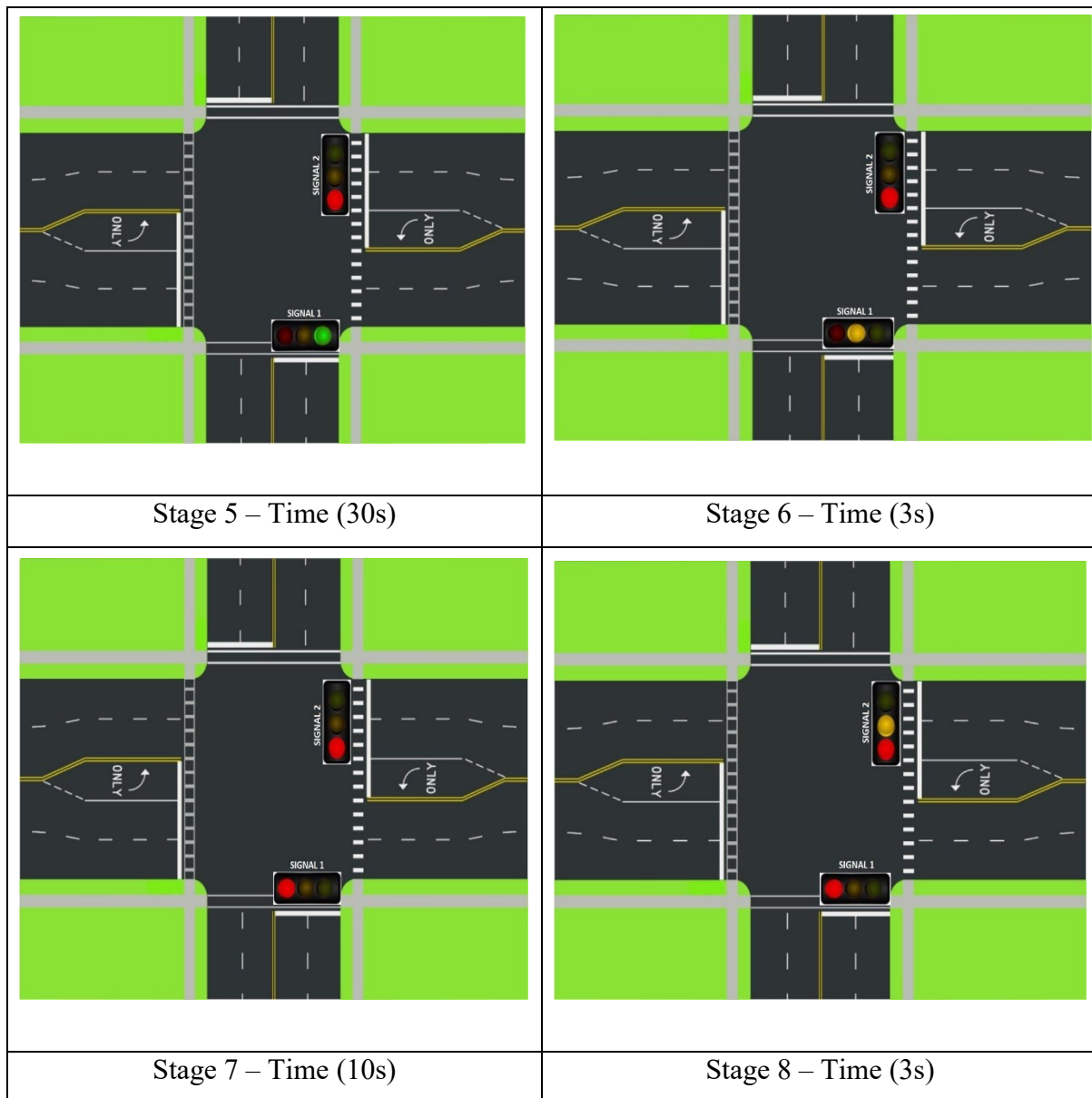4. Connect the other end of the wire to the Arduino Uno in the consecutive pins.
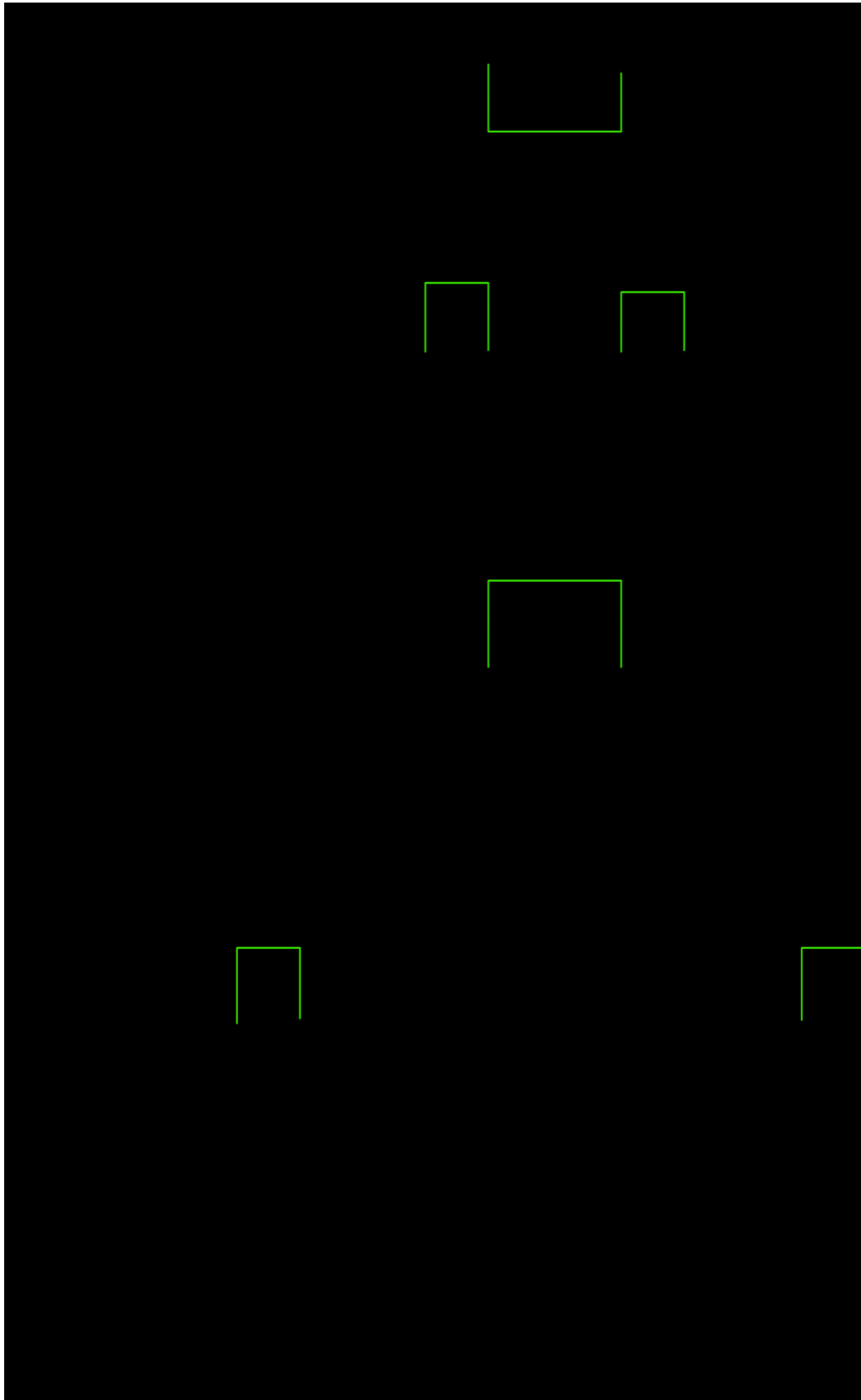
Figure 10: EUROPE SIGNAL

6. AUTOMATION SEQUENCES

Automation consists of developing and applying systems for the manufacture and distribution of goods and services with minimum interference by humans. The introduction of technology, techniques and processes for automation improves the performance, reliability and speed of many previous tasks carried out by people.

Stage 1 – Time (30s)	Stage 2 – Time (3s)
Stage 3 – Time (10s)	Stage 4 – Time (3s)

Stage 5 – Time (30s)	Stage 6 – Time (3s)
Stage 7 – Time (10s)	Stage 8 – Time (3s)
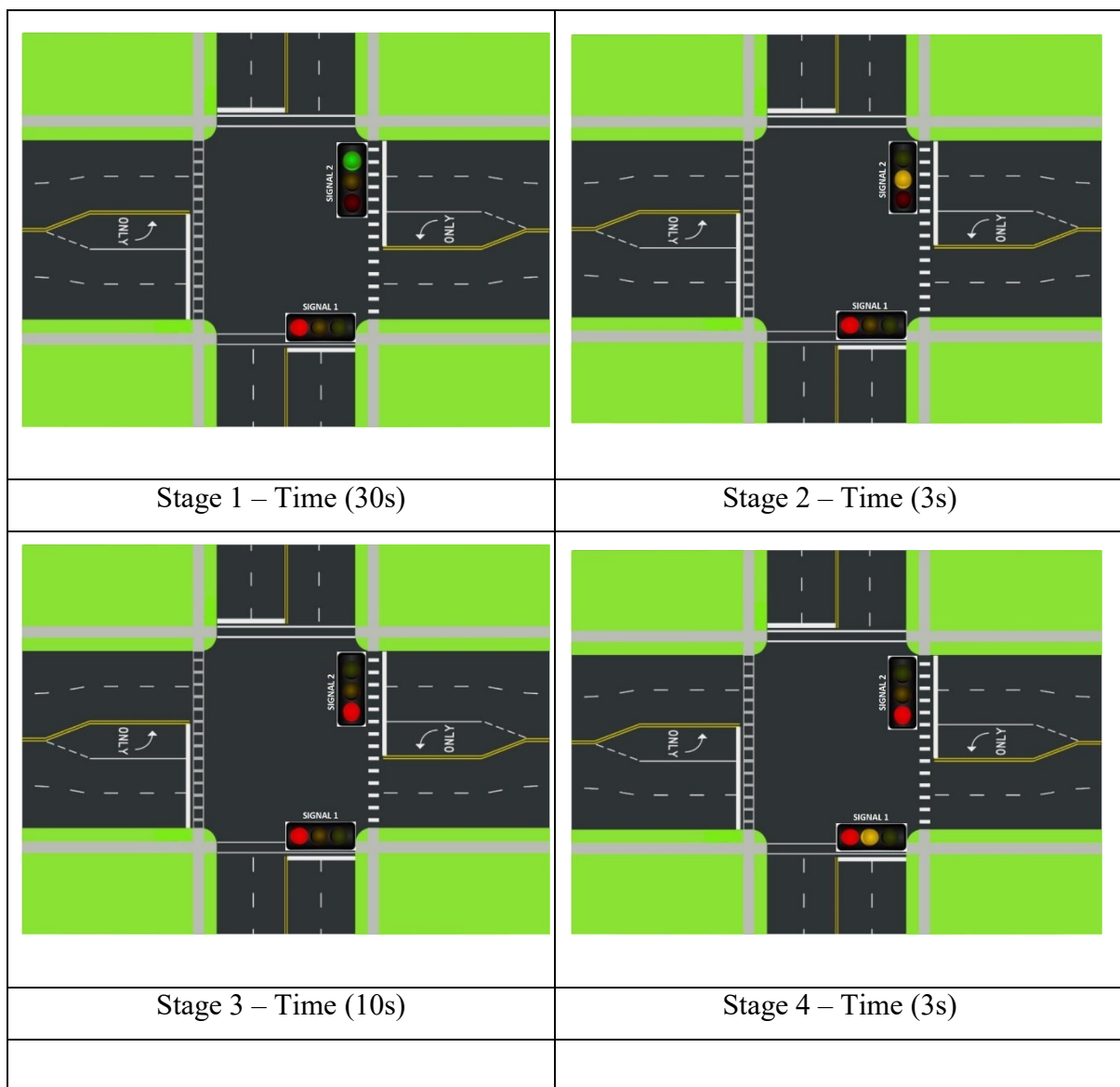
This is a lighting diagram, its indicated four side of traffic signals.

1. First signal indicated the red and green light for 30s.

2. Second signal indicated the red and orange for 3s.

3. Third signal indicated the both red two side would be 10s.

4. Fourth signal indicated orange and both side red for 3s.

5. Fifth signal indicated green and red for 30s.

6. Sixth signal indicated orange and red for3s.

7. Seventh signal indicated both red two side would be 10s.

8. Eighth signal indicated red on both side and one side would be orange for 3s.

R1	Red light of first signal
O1	Orange light of first signal
G1	Green light of first signal
R2	Red light of second signal
O2	Orange light of second signal
G2	Green light of second signal
T	Time of the signal

Its indicated the lighting signal R, O, G are indicated the lighting colour and signals.
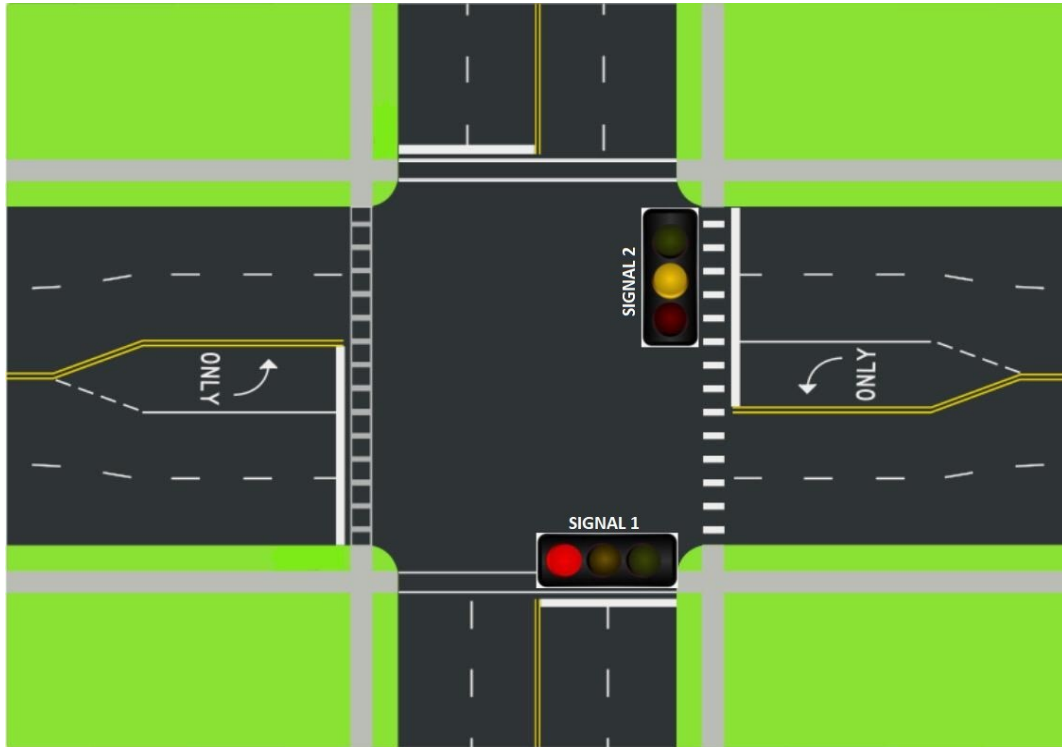
6.1 SEQUENCE 1
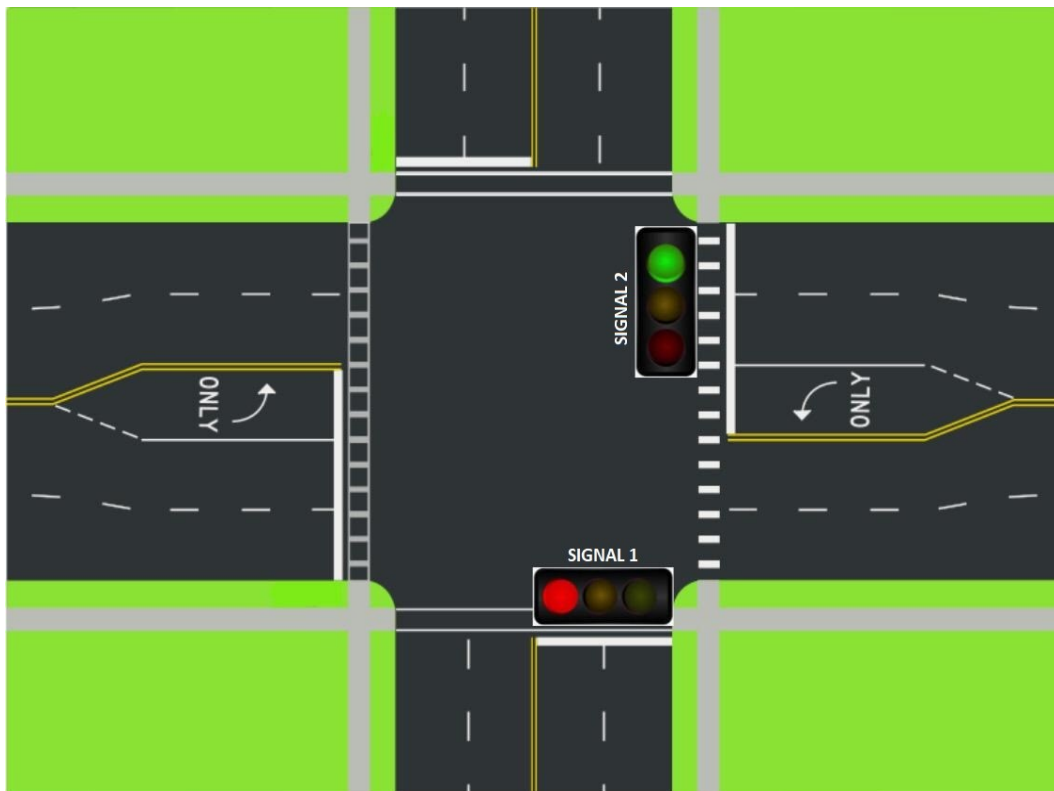
Figure 11: SEQUENCE 1

The above (Figure 2) represents the four road intersection, with two traffic light signal lamp. The red light signal means to stop, orange light warns that the red light signal is about to appear or red light in conjunction with orange light means green light is about to appear and green light signal means free to go.

Where the Signal 1 lamp shows the red light (stop sign) for the vehicles travelling towards the direction of the signal 1 and Signal 2 shows the green light (free to go) for the vehicles travelling in the respective signalling road.
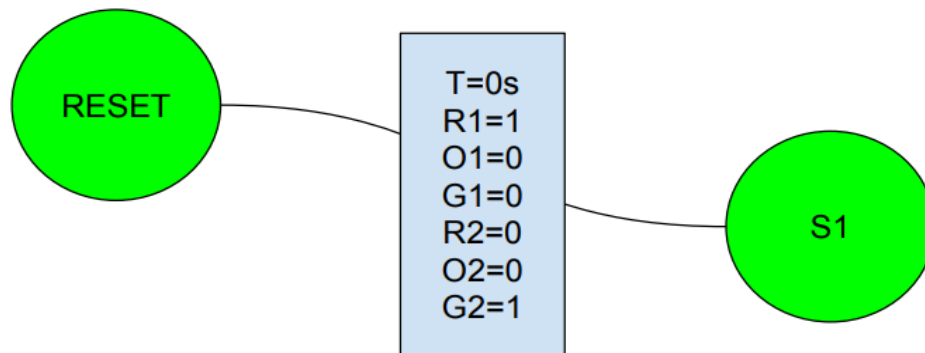
Figure 12: SEQUENCE 1

A sequence diagram displays interactions of objects organized in time. Sequence diagrams shows workflow or time operation by Messages transmitted from item to object.
In this sequence red light is indicated on left side of the traffic light and green light is indicated on right side of the traffic light. Timing of the sequence is T=0 and we can give the Red1=1 and Green2=1, all other of the traffic light are 0.

6.2 SEQUENCE 2

Figure 13: SEQUENCE 2

The above (Figure 4) represents the four road intersection, with two traffic light signal lamp. Where the Signal 1 lamp shows the red light (stop sign) for the vehicles travelling towards the direction of the signal 1. Signal 2 shows the orange light (warns that the red light signal is about to appear) for the vehicles travelling in the respective signalling road.
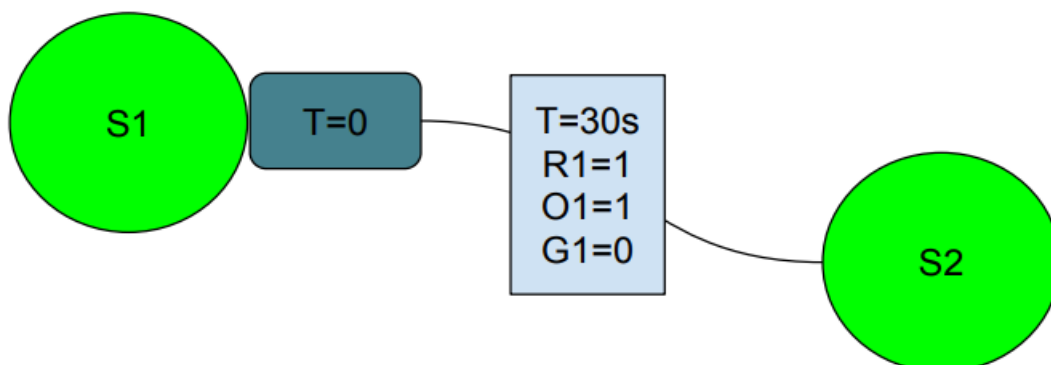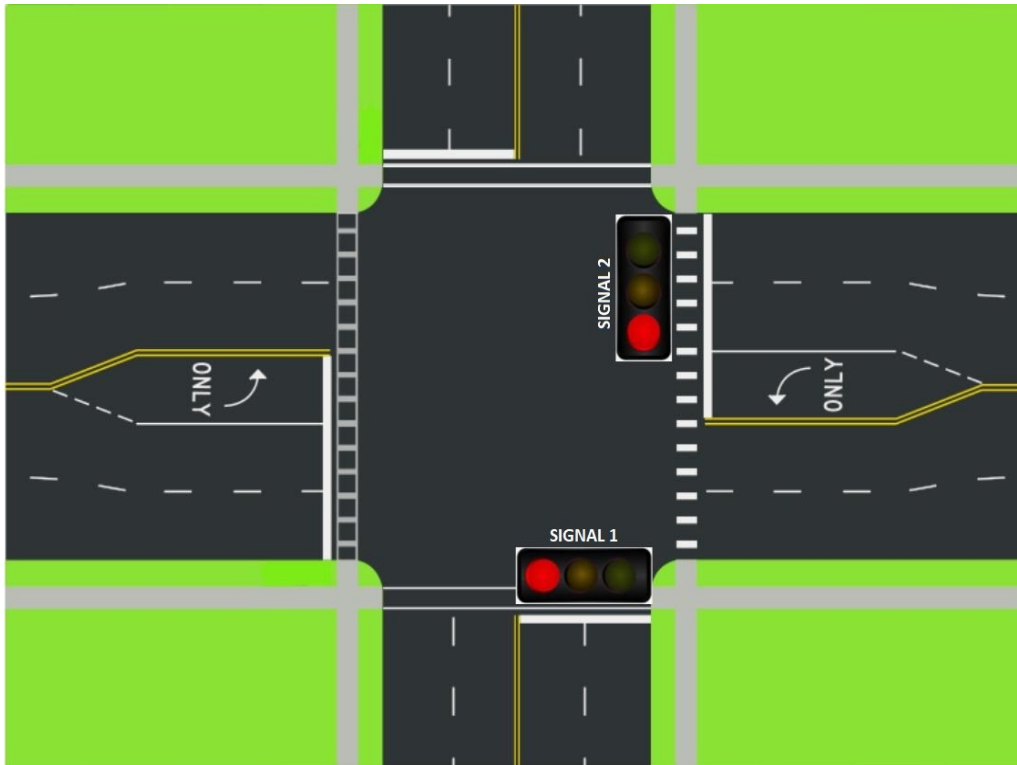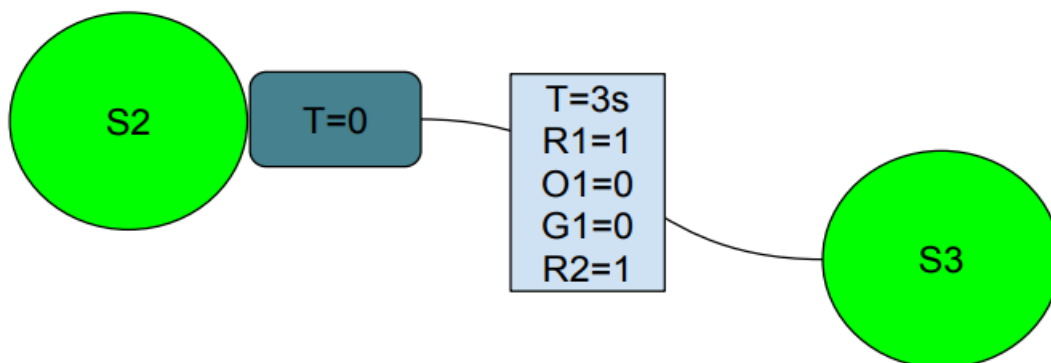

Figure 14: SEQUENCE 2

In this sequence red light in left side and orange light in right side of the traffic light. Timing of the sequence is T=30s Red1=1, Orange1=1 and Green1=0.
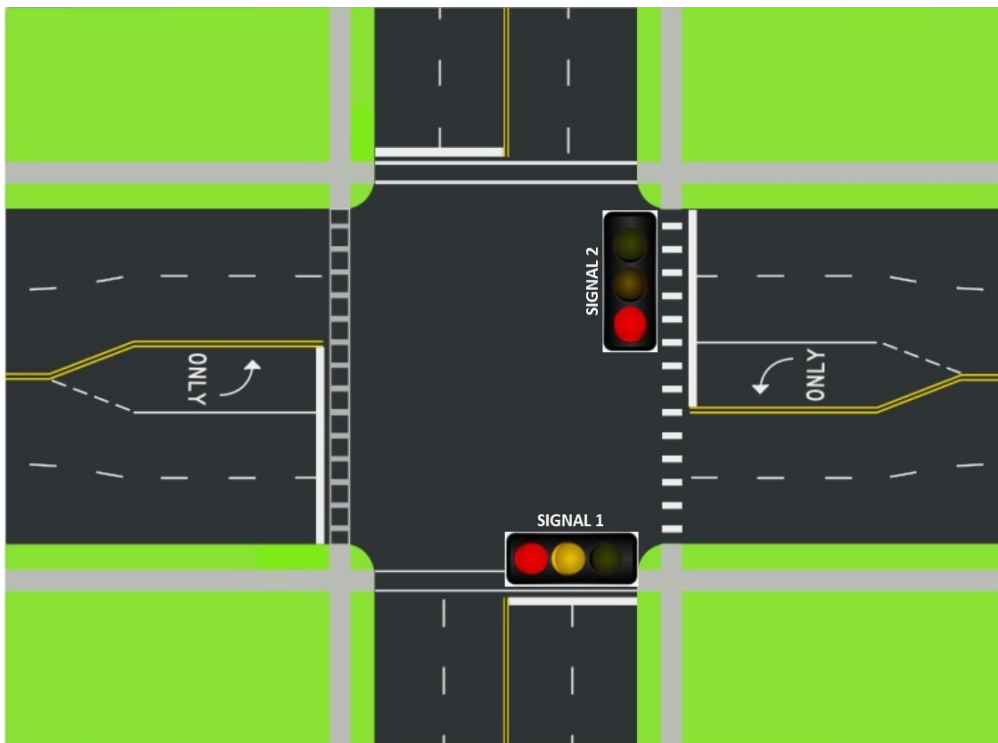
6.3 SEQUENCE 3

Figure 15: SEQUENCE 3

The above (Figure 5) represents the four road intersection, with two traffic light signal lamp. Where the Signal 1 lamp shows the red light (stop sign) for the vehicles travelling towards the direction of the signal 1. Signal 2 shows the red light (stop sign) for the vehicles travelling in the respective signalling road.
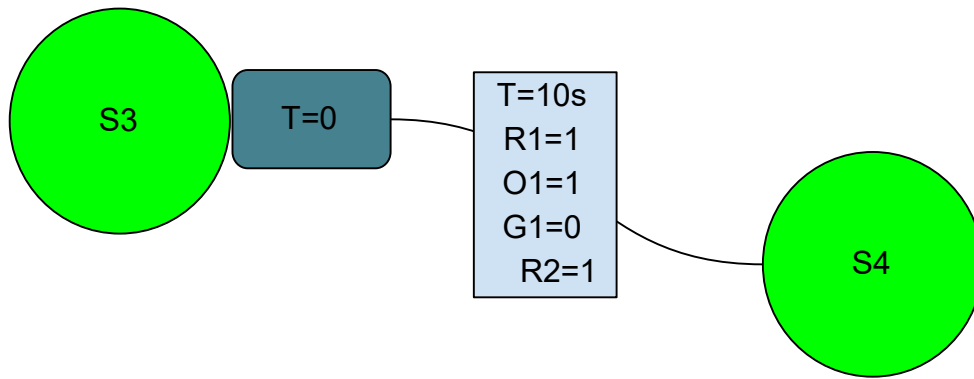
Figure 16: SEQUENCE 3

A graphical diagram is constructed in such a way that a time-line starts at the top and progressively descends to mark the interaction series. Each object has a column and arrows reflect the messages exchanged between them.

In this sequence both side are having red light. Timing of the traffic light is T=3s we can have added Red1 and Red 2 is on.
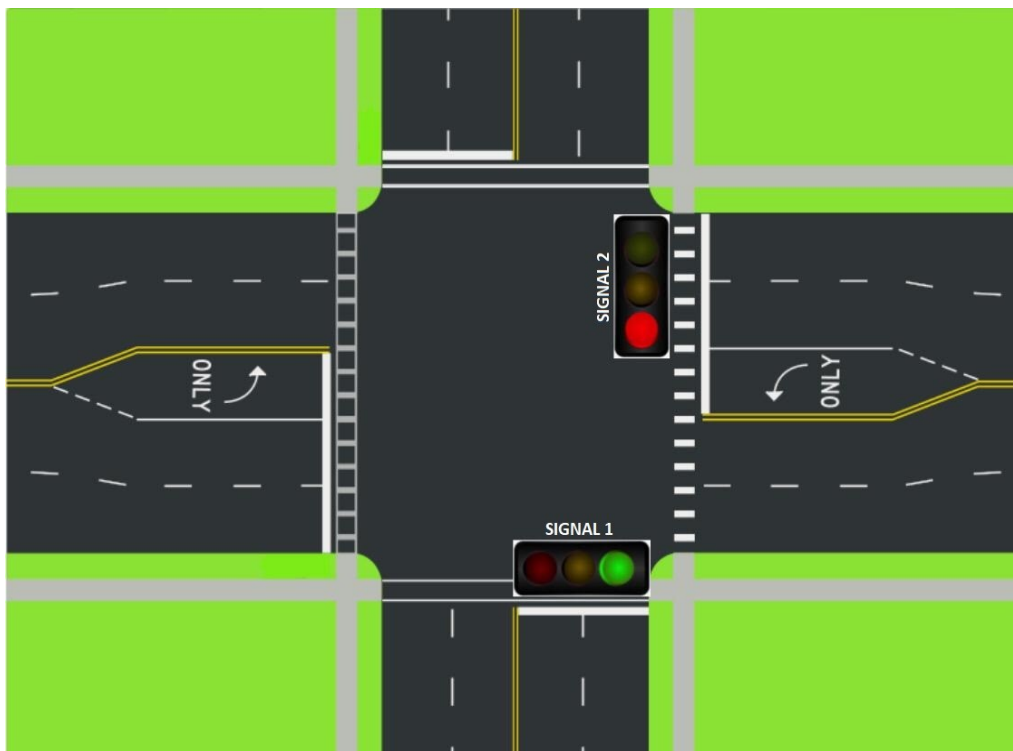
6.4 SEQUENCE 4

Figure 17: SEQUENCE 4

The above (Figure 6) represents the four road intersection, with two traffic light signal lamp. Where the Signal 1 lamp shows the red light in conjunction with the orange light means green light is about to appear for the vehicles travelling towards the direction of the signal 1. Signal 2 shows the red light (stop sign) for the vehicles travelling in the respective signalling road.

Figure 18: SEQUENCE 4

In this sequence left orange and red, another side of the part red is on right. Timing of the parts are T=10s we can indicate the Red1=1, Orange1=1 and Red2=1.

6.5 SEQUENCE 5

Figure 19: SEQUENCE 5

The above (Figure 7) represents the four road intersection, with two traffic light signal lamp. Where the Signal 1 lamp shows the green light (free to go sign) for the vehicles travelling towards the direction of the signal 1. Signal 2 shows the red light (stop sign) for the vehicles travelling in the respective signalling road.
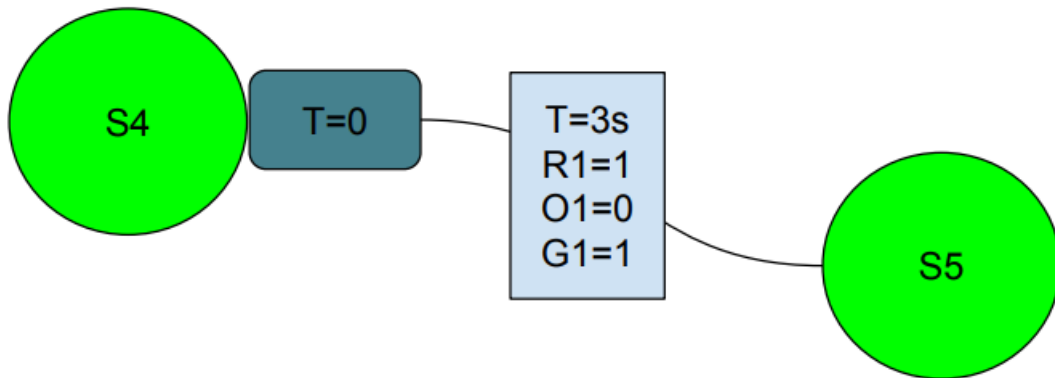
Figure 20: SEQUENCE 5

In this sequence green light in left side and red light in right side of the traffic light. Timing of this path is T=3s. Red1=1 and G1=1.
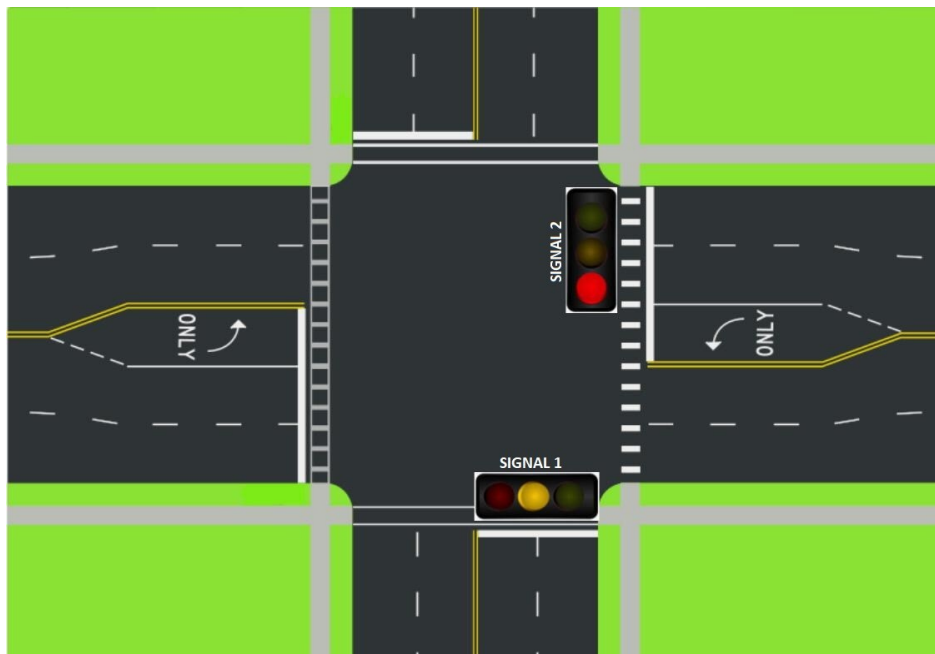
6.6 SEQUENCE 6

Figure 21: SEQUENCE 6

The above (Figure 8) represents the four road intersection, with two traffic light signal lamp. Where the Signal 1 lamp shows the orange light (warns that the red light signal is about to appear) for the vehicles travelling towards the direction of the signal 1. Signal 2 shows the red light (stop sign) for the vehicles travelling in the respective signalling road.
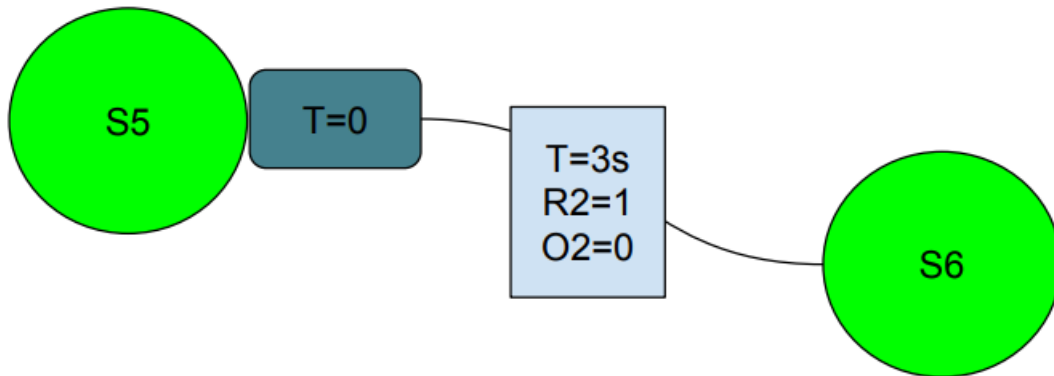

Figure 22: SEQUENCE 6

The multiplicity refers to how many objects may be involved in the given connection or the permissible number of instances.

Display sequences of states that during its lifetime a subject or an interaction is passing through in response to received stimuli and its responses and behaviour.

Displaying graphs of states and transitions describing the response to outside stimuli from a given class object

In this sequence orange light in left side and red light in right side of the traffic light. T=3s

Is the timing of the path R2=1 and O2=1.
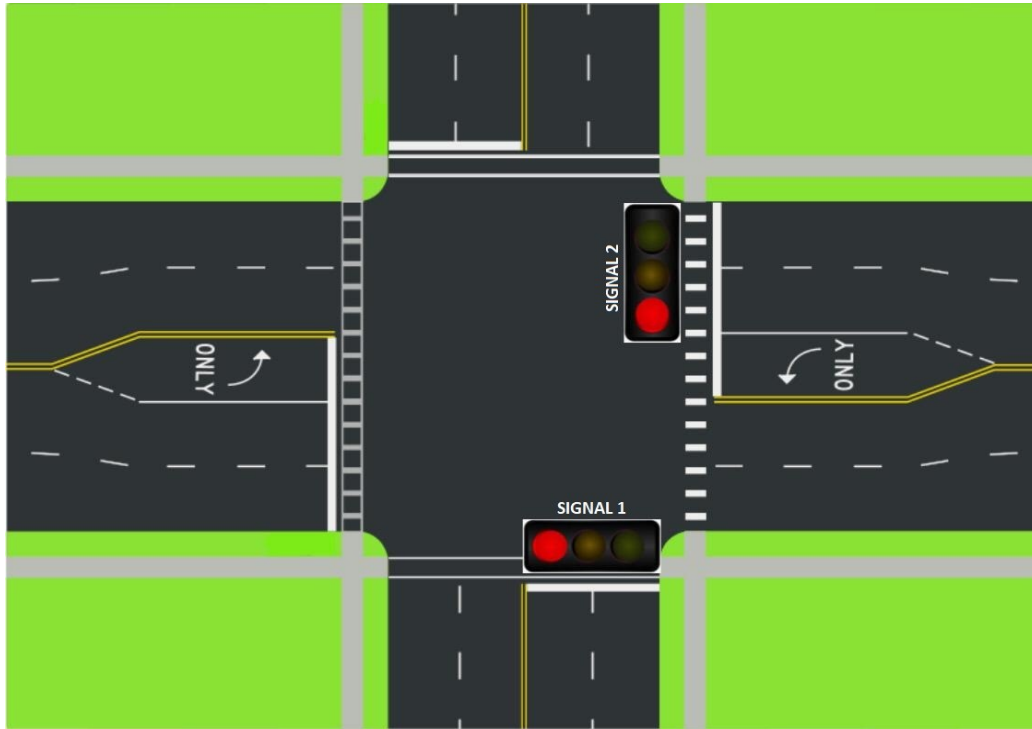
6.7 SEQUENCE 7

Figure 23: SEQUENCE 7

The above (Figure 9) represents the four road intersection, with two traffic light signal lamp. Where the Signal 1 lamp shows the red light (stop sign) for the vehicles travelling towards the direction of the signal 1. Signal 2 shows the red light (stop sign) for the vehicles travelling
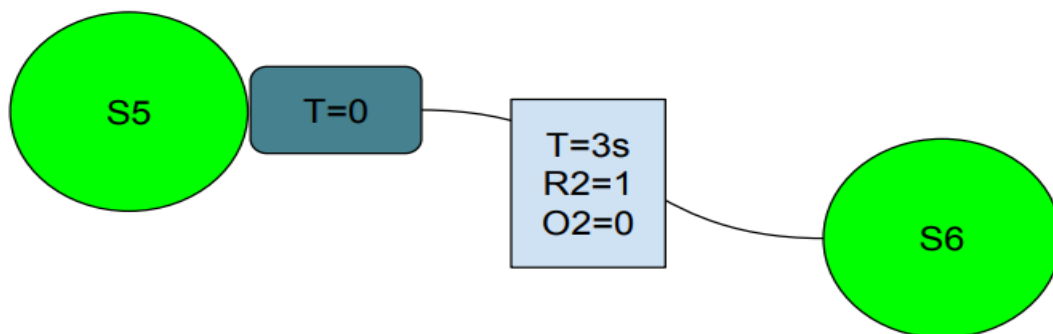

Figure 24: SEQUENCE 7

in the respective signalling road.

In this sequence orange light in left side and red light in right side of the traffic light. T=3s

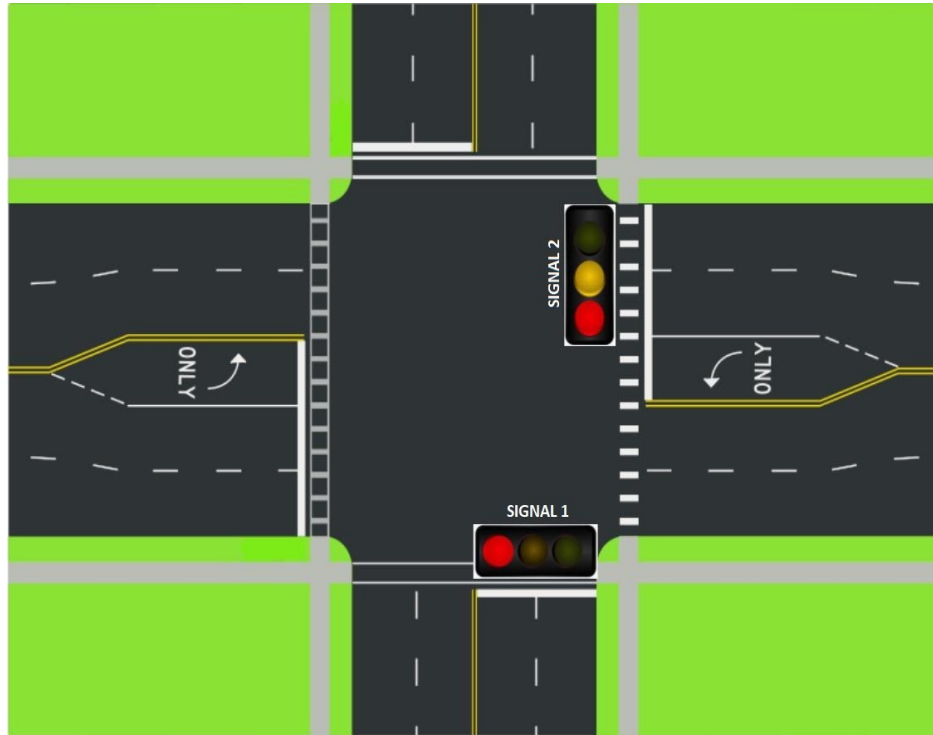Is the timing of the path R2=1 and O2=0.

6.8 SEQUENCE 8

Figure 25: SEQUENCE 8

The above (Figure 10) represents the four road intersection, with two traffic light signal lamp. Where the Signal 1 lamp shows the red light (stop sign) for the vehicles travelling towards the direction of the signal 1. Signal 2 shows the red light in conjunction with the orange light means green light is about to appear /for the vehicles travelling in the respective signalling road.
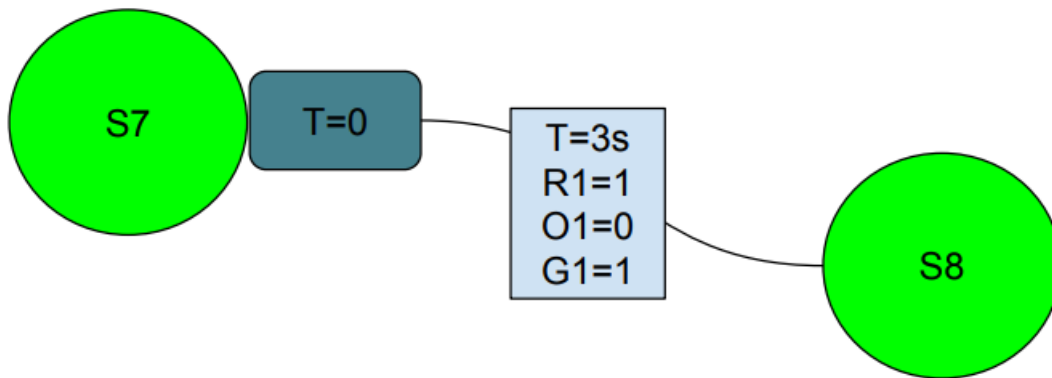
Figure 26: SEQUENCE 8

In this sequence red light in left red and orange on in right side of the traffic light. T=3s is the timing of the traffic light R1=1, O1=0 and G1=1.

7.DESIGN OF TRAFFIC LIGHT

7.1 BASICS OF CATIA

Enables smooth progression of design methodologies from 2D- to 3D Process full and constant drawing. Since it supports many phases of product creation from conception, design and engineering to production, it is often referred to as a 3D product cycle software suite. It promotes cooperation engineering through an integrated Cloud Infrastructure like most of its competition and is supported in all areas, including surface and design, electrical, fluid and electronic systems design, engineering and device engineering. It is mainly used in the aerospace and automotive industries. Many car manufacturers use CATIA to varying degrees, such as BMW, Porsche, Daimler Chrysler, Audi, Bentley Motors Limited, Volvo, Fiat, Benteler AG, PSA Peugeott Motors and Mahindra and Mahindra Limited, as well as PSA Peugeot Citroën, Renault and Toyota.

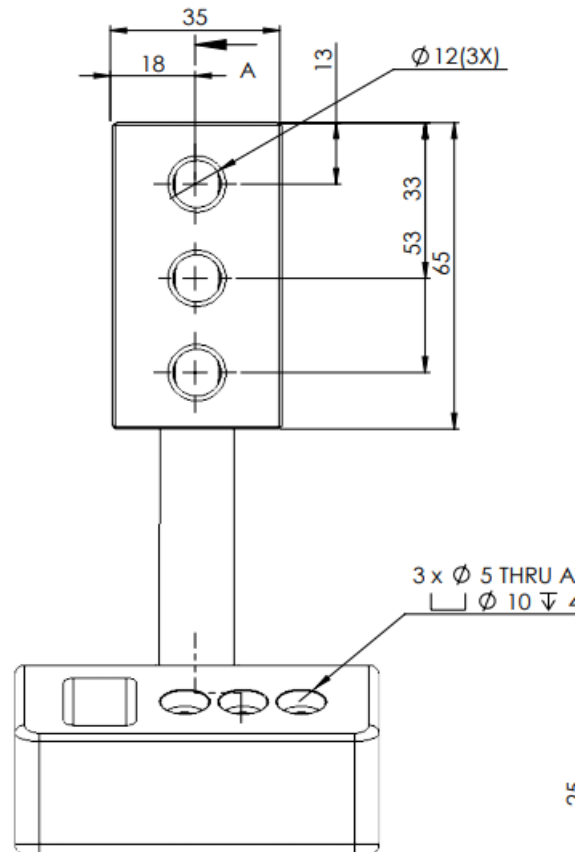*Figure 27: DESIGN OF TRAFFIC LIGHT*

7.2 MODELS   OF   CATIA

Specify simple code that can include geometrical, textual and modifiable dimensions to create 3D printable designs. Opens cad makes fantastic parameter designs – templates that can be easily modified in numbers to alter the scale, form or some other element of a design. CAD is conveniently a totally free tool on every big platform. After downloading open cad and opening a SCAD file, it's easy to convert.

We want to draw a triangle; in a that triangle we want to converter three circle diagram.

At the upper of the triangle one circle to connected the wire from the display and switch.

This is the original material which we prepared in catia model. After we send this model to 3D Printer.
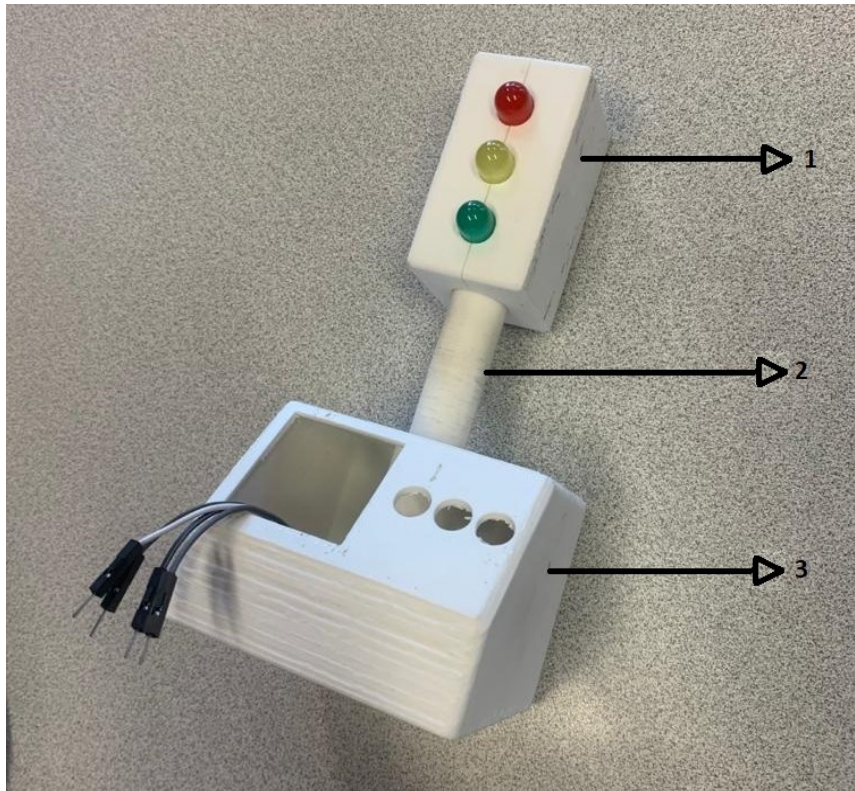
Figure 28: Prototype with Lights

As discussed in previous topics the complete traffic light model is designed in CATIA and the model is prototyped in 3D printer, also known as additive manufacturing which is a method of creating a three dimensional object layer by layer using a computer created design.

In this thesis the complete traffic light 3D model was printed with (PLA) material, which is a thermoplastic polyester. The reason behind selection of this material was because it can be produced economically from renewable resources.

The different types of parts include,

1. Signal head (2 pieces) with LED's

2. Pole mount (1 piece)

3. Stand with display and buttons (1 piece)

First step is to fix the LED lights; it is fixed in the signal head. The signal head consists of two parts which is to be assembled after fixing the LED lights. As shown in the Figure 1 the LED light is fixed in their respective position and the other side of the signal light is assembled. The hole below the signal head provides the assembling of the pole mount and this is the space through which the wirings from LED's travel to the display and buttons placed in the stand below. And the hole above the stand provides the space to assemble the lower end of the pole mount. This is the complete assembling procedure of the traffic light.
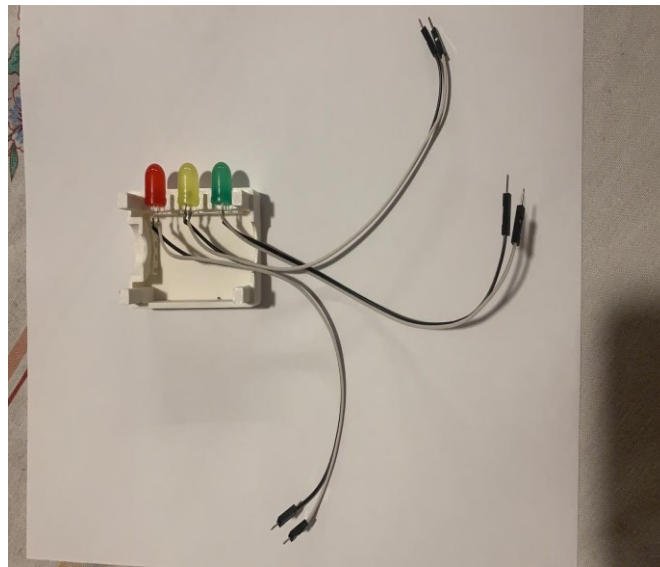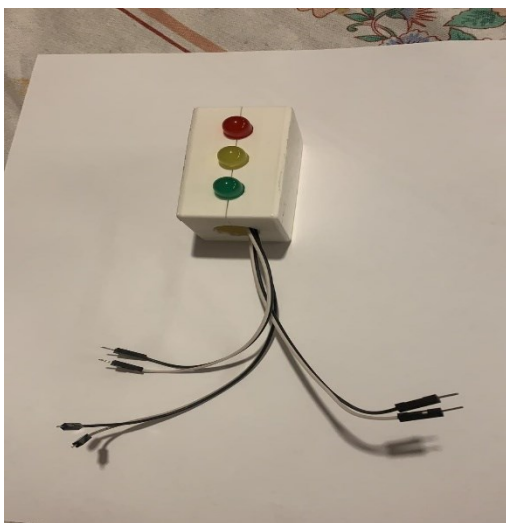

Figure 29: LIGHTS
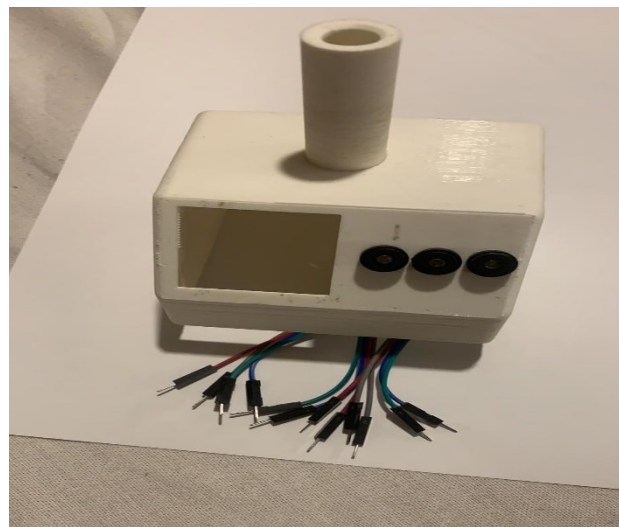

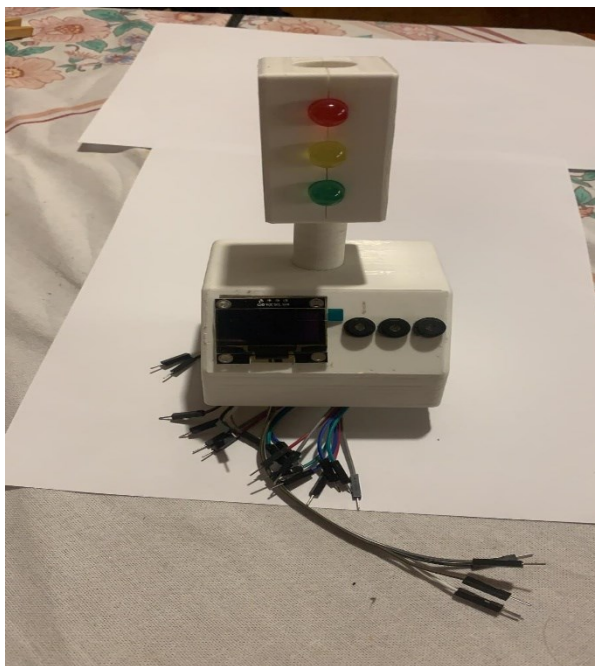*Figure 31: ASSEMBLY PART 1*

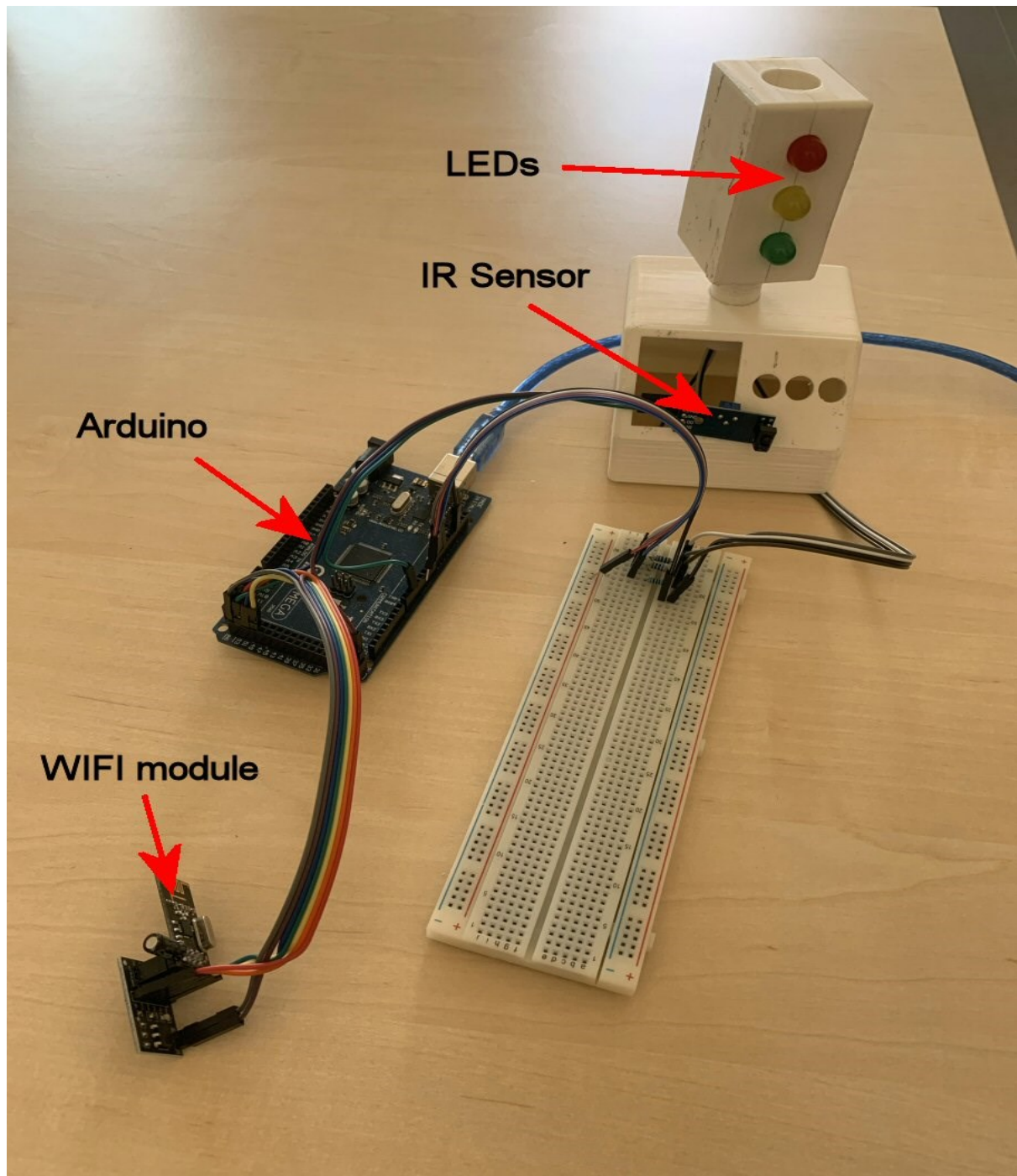*Figure 30: ASSEMBLY PART 2*

Figure 32COMPLETE ASSEMBLY

Figure 33 The model of the traffic light

8. TRAFFIC LIGHTS STRATEGIES

There are several strategies I used to control the lights. First of all, there are two situations:

a) Central control system with one Arduino only. All six colours R1, O1, G1, R2, O2 and G2 are connected with wires to this Arduino. These six colours are presented with six LED diods. only, but twice three pair – totally twelve LEDs are in real crossroad. It is in fig. 33

b) Four alone control systems with four Arduinos, because it is not possible to use wires on the real crossroad. There are four traffic lights masts. First mast is a Master. He does the timing and sends sync signal, current position, to other masts. There are two kinds of Slaves. Slave1 is in the same direction like Master and two Slaves2 are in second direction, Fig 34
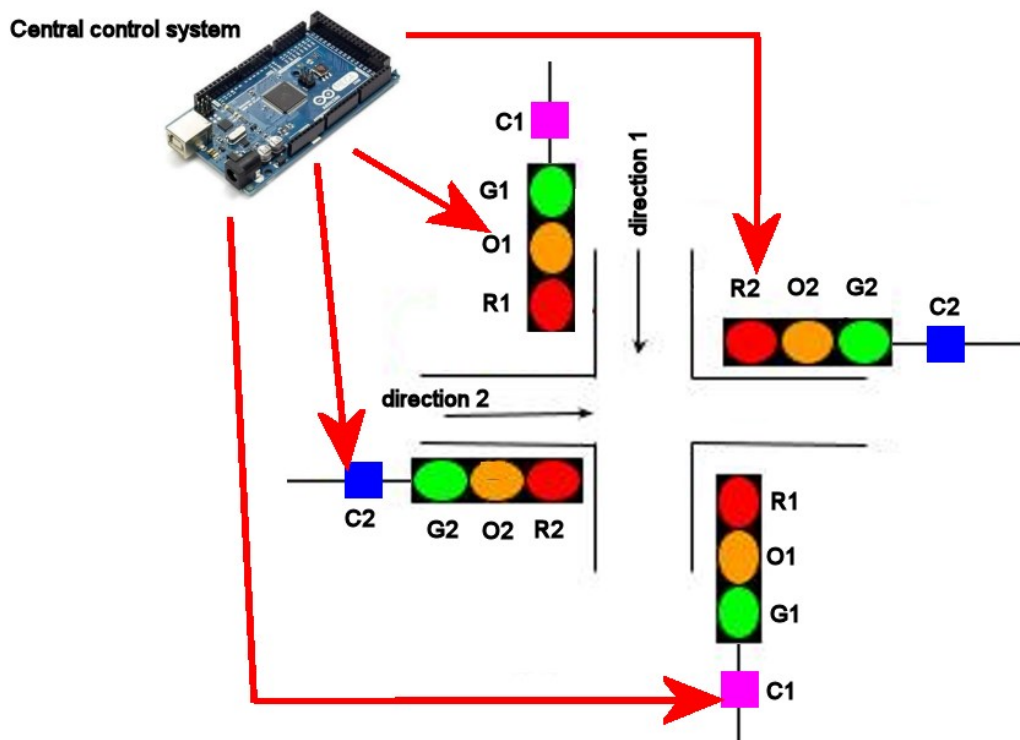

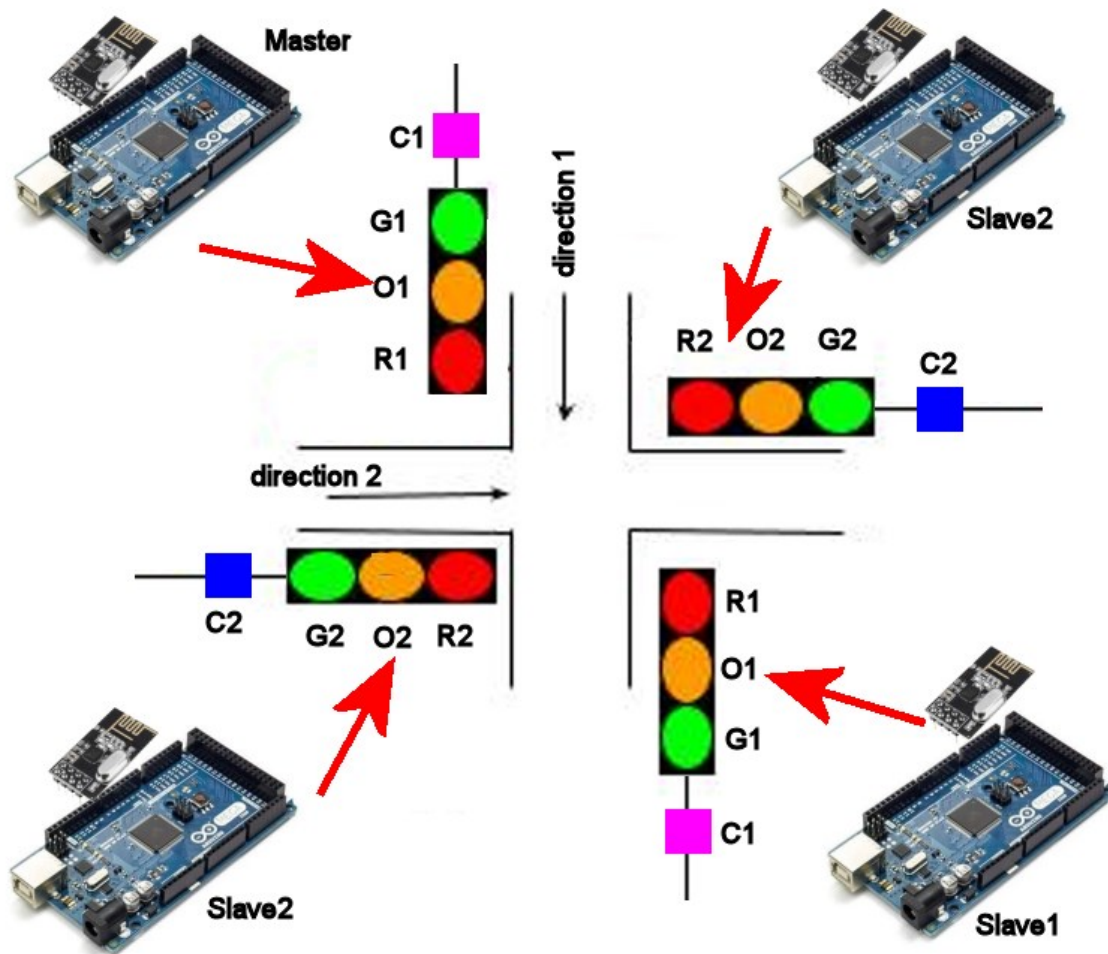*Figure 34 Central control of the model*

Figure 35 Four control systems synchronized via wifi

I designed these four strategies:

Strategy 1 - Traffic lights are alternating with predefined time sequence only. (30s for greens)

Strategy 2 – Sensors of cars added. Arriving car changes these 30s to 10s and gets green sooner.

Strategy 3 – One direction is the main and has green. Second direction has sensor of arriving

car. When car arrives it gets green and after a while (30s) the green returns to main direction.

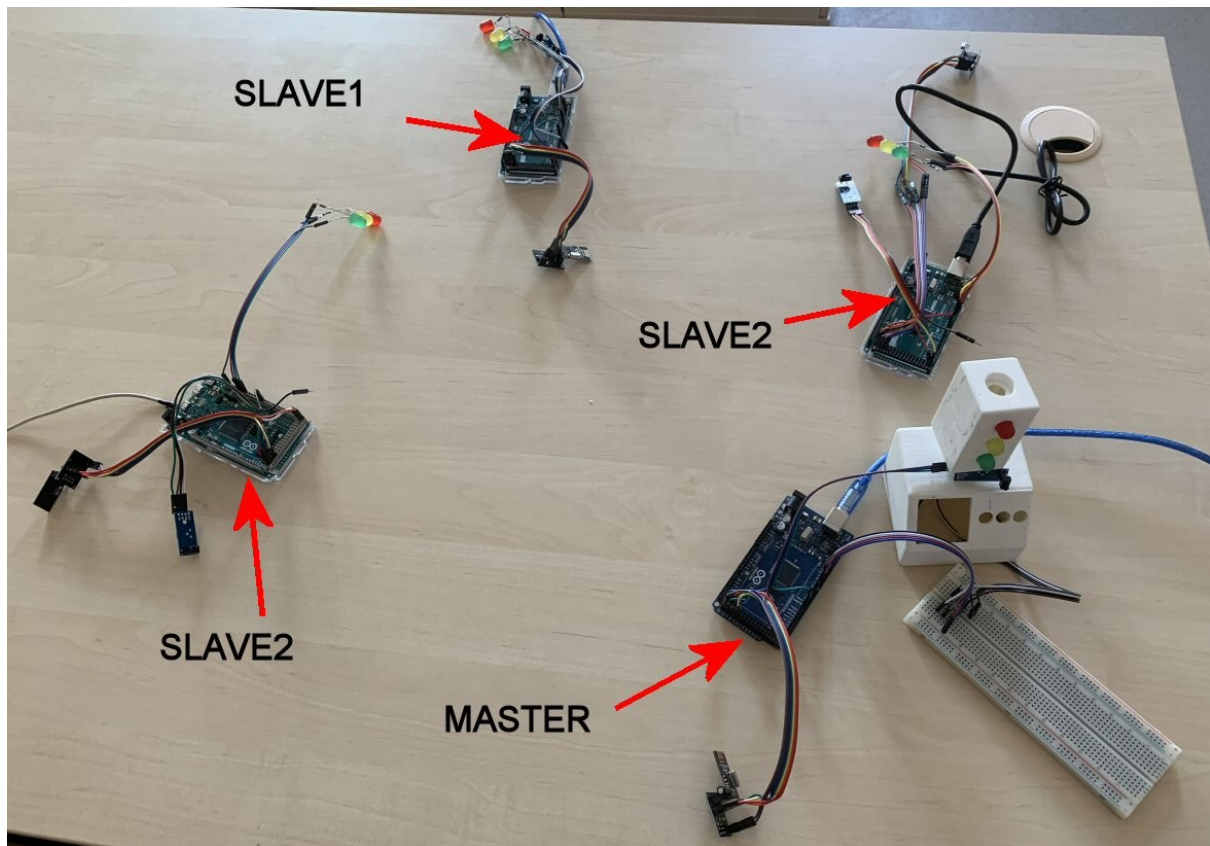Strategy 4 – Both direction have the same priority. Green is taken by sensors of arriving cars in both directions.

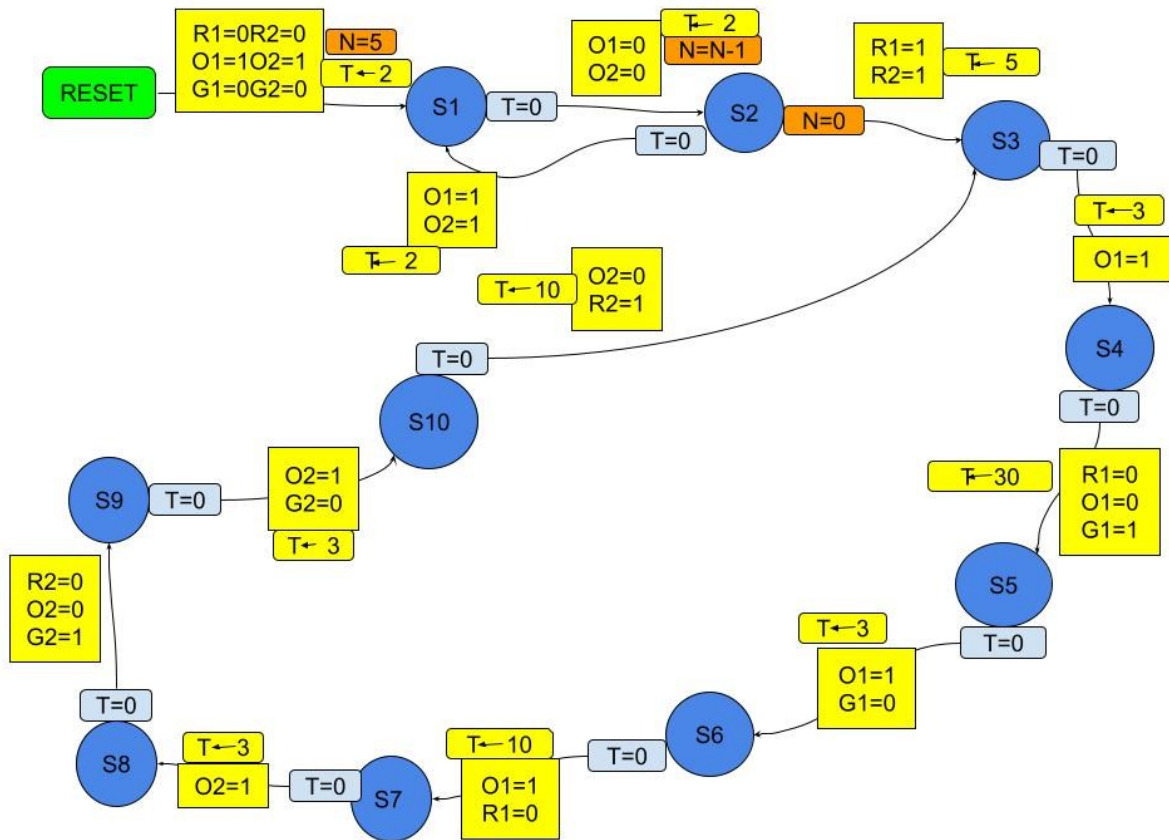*Figure 36The model of the crossroad*

8.1 Strategy 1 – timing only

Figure 37Automaton with STRATEGY 1

The automaton of this strategy is in fig.35. The automaton starts in RESET.There is initial action after RESET. It includes settings of all six lights. It is necessary to set the timer to 2s in state S1. The lights configuration is in the fig. 33, both oranges on. It continues with blinking of both oranges. If timer ends – condition T=0 – it jumps to state S2 and both oranges gets off. The timer is adjusted again to next 2s. After these 2s the automaton jumps back to S1. This circle can continue to infinity. Because of this, I added the orange variable N. N is adjusted to number 5 at the beginning and it is decreased in each cycle. If it reach N=0, the automaton leaves this circle and jumps to S3. Both reds are on and the timer is adjusted to 5s. The automaton continues throw next states around big circle. The timer is always adjusted to wanted time and the lights are changed according to real traffic lights on the crossroad. Then it returns back to the state S3 and goes again.

This strategy works on base of timing only. Both directions have their green for 30s.

8.2 Strategy 2 – timing with car sensors

There are two couples of sensors of cars. Without cars the greens changes after 30s. If a car at red appears, it decreases these 30s to 10s only. It is the same graph like in previous case, but here aux variable M is created. Before S6 M is adjusted to 30. If sensor C1 doesn't watch the car the variable M stays 30. But if car appears C1 watches this car and M is changed to 10. So M=30 if C1 is not 1 and M=10 if C1 is 0 during states S6-S7-S8. The same situation is with sensor C2 during states S10 –S3- S4. The automaton is in Fig. 36.
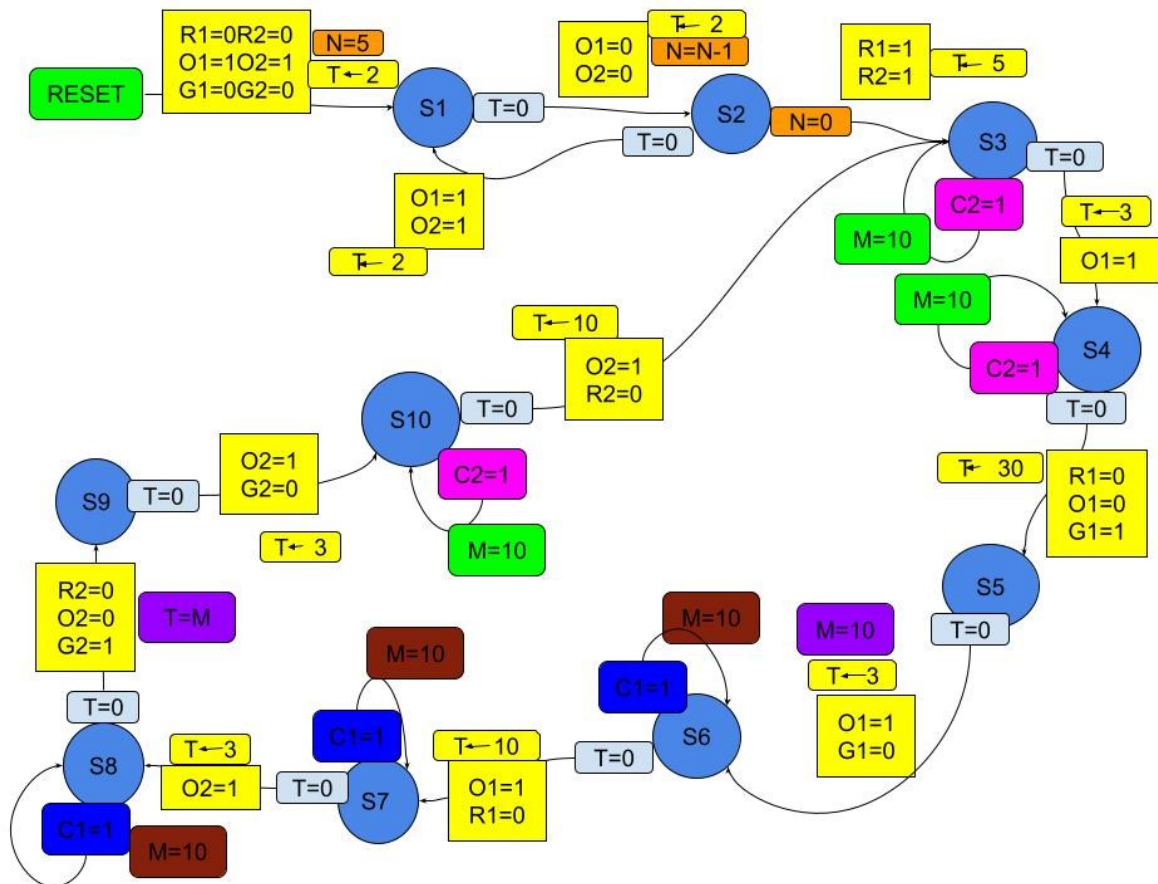

Figure 38 Automaton for Strategy 2

8.3 Strategy 3 – Main direction 1 and sensor C2

This strategy changes strategy 1. The time 30s for green in main direction 1 is changed to infinity. The green is still in direction 1. If a car appears in direction 2, the sensor C2 goes to 1 and it takes green to direction 2 for 30s. The automaton is in Fig. 37.
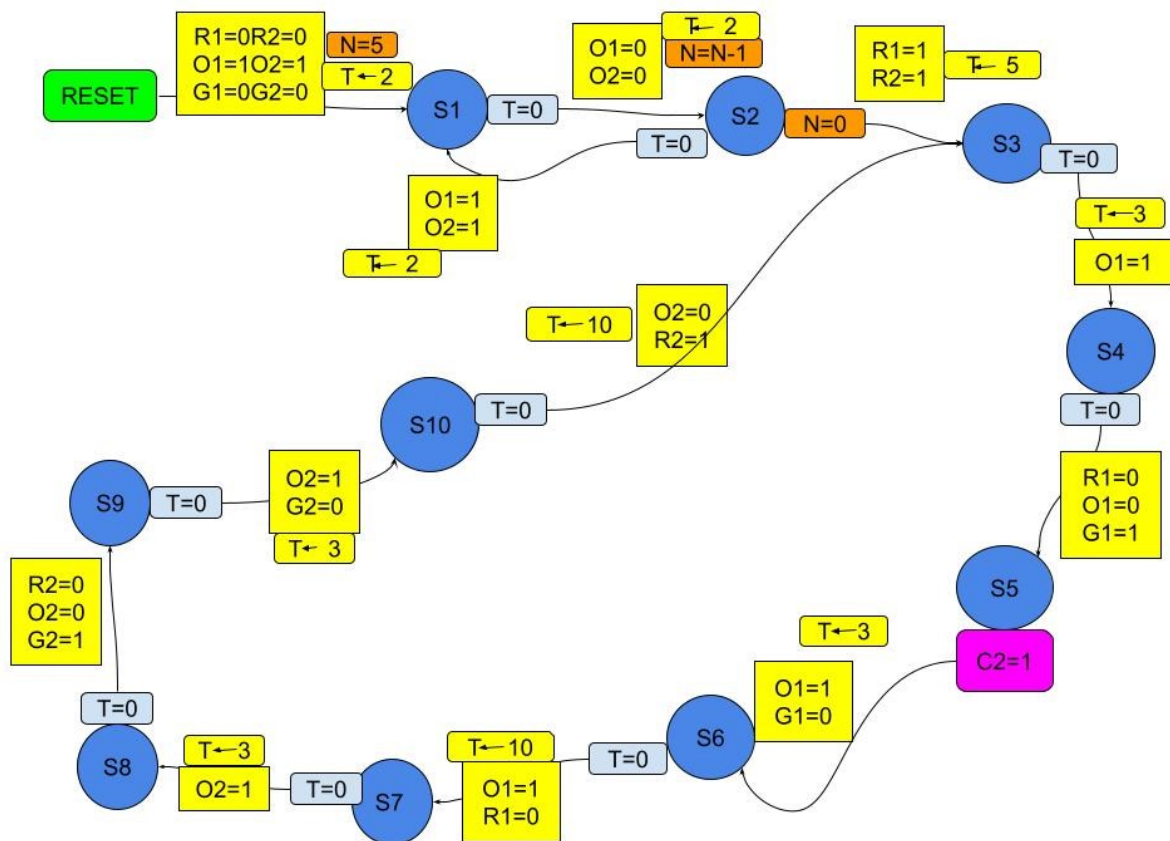
Figure 39Automaton for Strategy 3

8.4 Strategy 4 – Both directions with the same priority

In this diagram S5 is C2=1 which indicated the orange signal of the traffic light.

During this automation if time ends with condition T=0, and its jump to S6 orange gets off the

timer is adjusted again. The 30s for opposite direction is changed with red and the green is still
to direction 1, if car appears in direction 2, it enables its green for 30s. The automaton is in
fig.38.
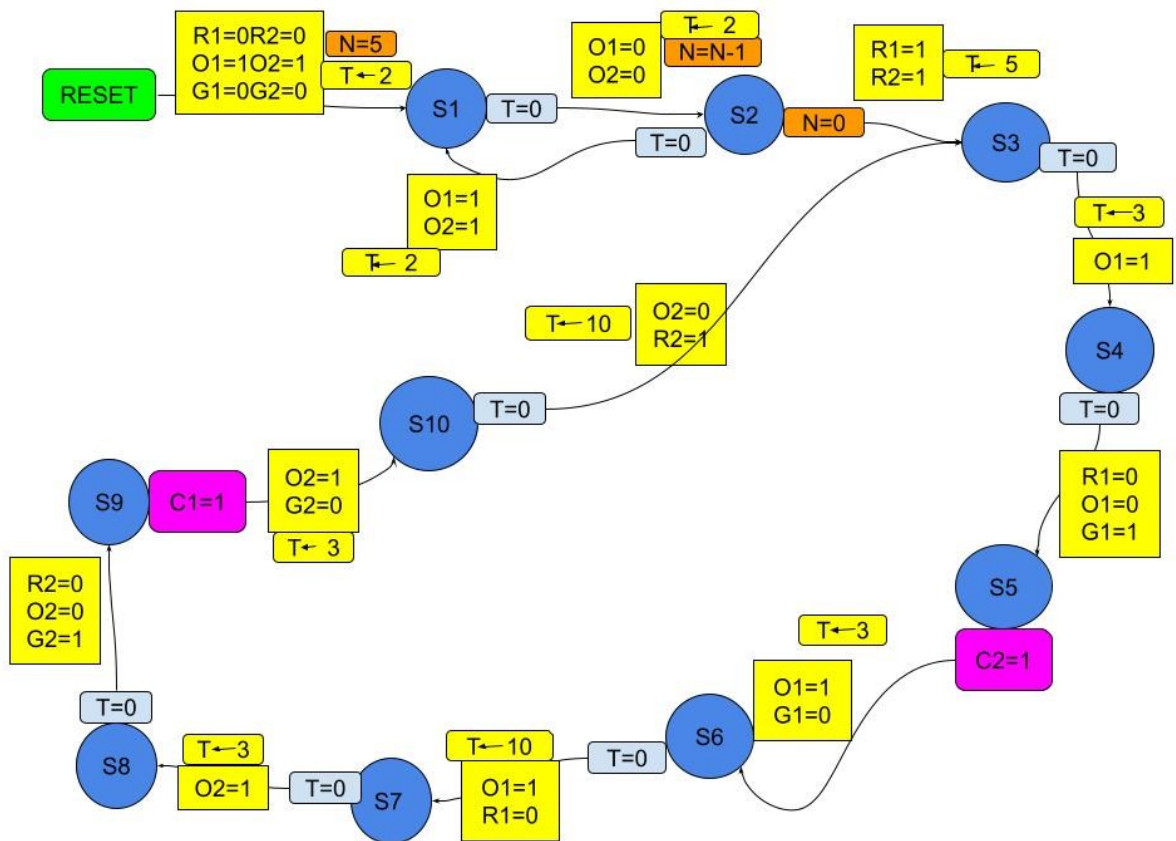
Figure 40 Automaton for Strategy 4

8.5 Wifi communication between independent masts

Four alone control systems are necessary to use instead of wires. There are four Arduinos, because it is not possible to use wires on the real crossroad. There are four traffic lights masts. First mast is a Master and it works the timing and sends its current position in grapg to other masts. There are two kinds of Slaves. Slave1 is in the same direction like Master and Slaves2 twice are used in second direction, like in Fig 34
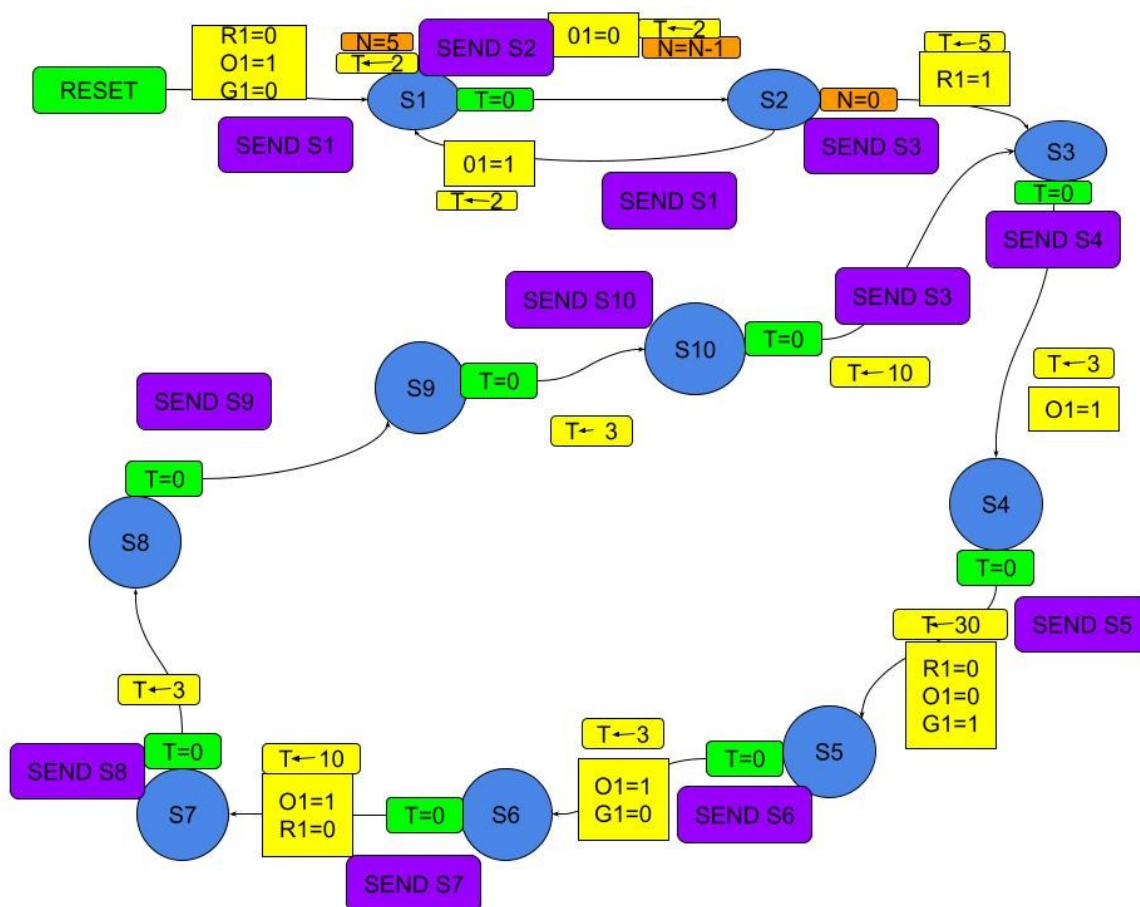
Figure 41Automaton of Master mast

The automation of Master corresponds to central automaton of strategy 1. But all actions with non local variables are removed. R1, O1 and G1 stay here. The wifi messages with current state are sent.
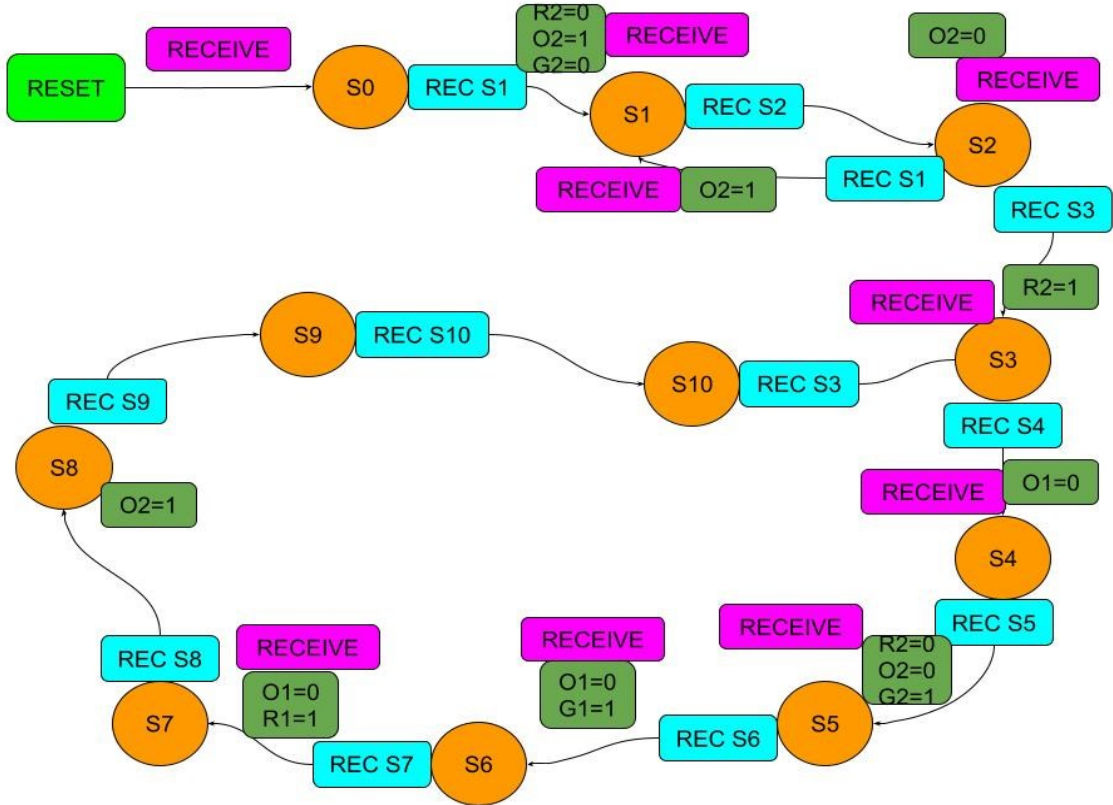
Figure 42 Automaton of Slave 1

The automaton of Slave 1 is in figure 40. The position in the graph is controlled with received messages via wifi. This control system is in the same direction like Master, so all actions have index 1.

And finally the automaton of Slave 2 is in fig. 41. This is controlled with messages too and the actions have indexes 2.
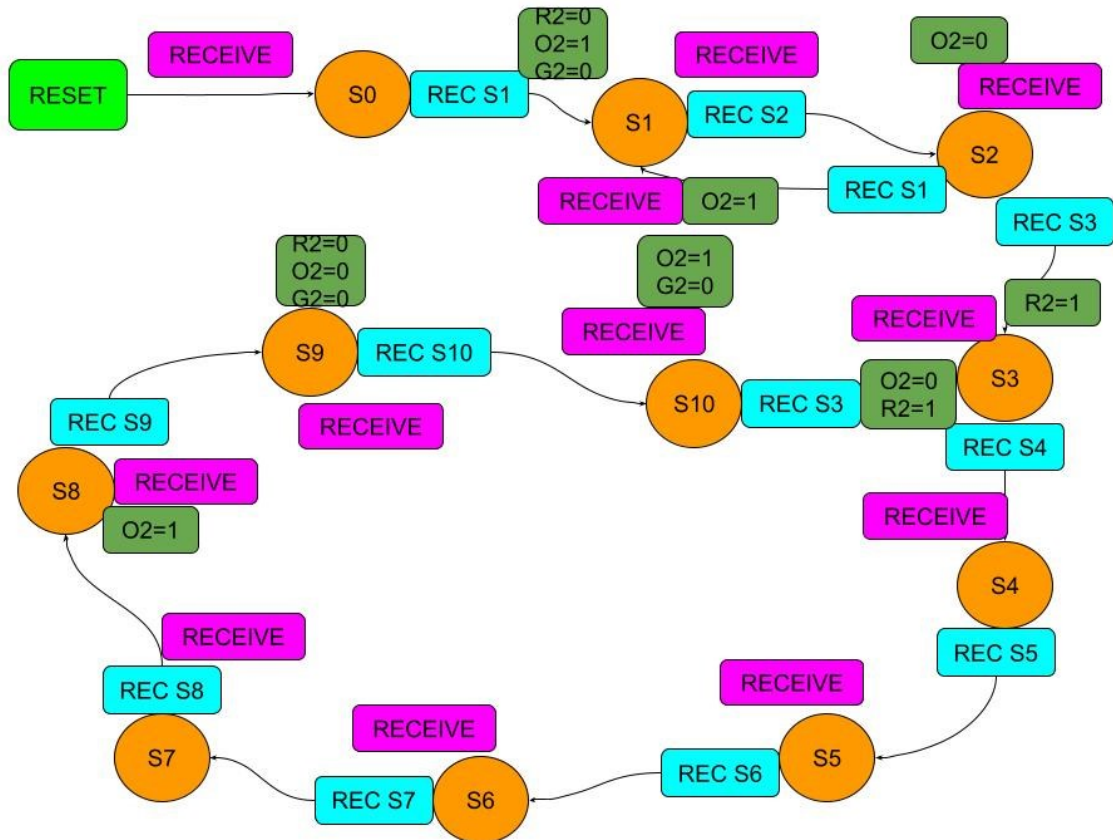
Figure 43 Automaton of Slave 2

The initialization and settings of wifi module is here. The message consists of the number of state only.

```
#include <Wire.h>                //LCD
#include <LiquidCrystal_I2C.h>      //LCD
#include <SPI.h>                //WIFI NRF
#include "RF24.h"                //WIFI NRF
LiquidCrystal_I2C lcd(0x27, 20, 4);  //LCD (adress,size)
RF24 myRadio(49, 53);             //WIFI NRF (MEGA)
byte addresses[][6] = {"0"};       //WIFI NRF
struct package {                //WIFI NRF (data in message)
  int S=0;
};
```

```
typedef struct package Package;  //WIFI NRF

Package dataRecieve;           //WIFI NRF

Package dataTransmit;          //WIFI NRF


myRadio.begin();               //WIFI NRF
 myRadio.setChannel(0x60);
 myRadio.setPALevel(RF24_PA_MAX);
```

The sending of current state is Action and the code is here:

```
myRadio.stopListening();          //WIFI NRF TRANSMIT
dataTransmit.S=s;
myRadio.openWritingPipe(addresses[0]);
myRadio.write(&dataTransmit, sizeof(dataTransmit));
```

The receiving of these messages is Action RECIEVE and Condition and their code is here:

```
myRadio.openReadingPipe(1, addresses[0]); //automaton-action listen (RECEIVE)
myRadio.startListening();
printLcd(0,"Server-listening");

if (S==7) {while (myRadio.available()) {
         myRadio.read(&dataRecieve, sizeof(dataRecieve));}
         if (dataRecieve.S==8) {   S=9 }
```

9. PROGRAM OF TRAFFIC LIGHT - example

```
const int r1Pin = 8;    //light pins 1.direction

const int o1Pin = 9;

const int g1Pin = 10;

const int r2Pin = 7;    //light pins 2.direction

const int o2Pin = 6;

const int g2Pin = 5;


int S;               //State variable

int N;

unsigned long stoptime;   //variable for the timer

void setup() {

 pinMode(r1Pin, OUTPUT);

 pinMode(o1Pin, OUTPUT);

 pinMode(g1Pin, OUTPUT);

 stoptime=millis()+2000;  //action settings of timer T=2

 N=5;                 //action small violet rect

 digitalWrite(r1Pin, 0);  //action the big rect

 digitalWrite(o1Pin, 1);

 digitalWrite(g1Pin, 0);

 S=1;                //arrow from RESET to S1

 }

void loop() {

 if (S==1){if (millis()>=stoptime) {digitalWrite(o1Pin, 0);

                  stoptime = millis () + 2000;

                  N=N-1;

                  S=2;

                  }
```

```
        }
    if (S==2){if (millis()>=stoptime) {digitalWrite(o1Pin, 1);

                        stoptime = millis () + 2000;

                    S=1;

                    }

        if (N==0) {digitalWrite(r1Pin, 1);

            S=3;

            }

        }

}

int red = 10;

int orange = 9;

int green = 8;

int button = 12; // switch is on pin 12

void setup(){

    pinMode(red, OUTPUT);

    pinMode(orange, OUTPUT);

    pinMode(green, OUTPUT);

    pinMode(button, INPUT);

    digitalWrite(green, HIGH);

}

void loop() {

    if (digitalRead(button) == HIGH){

        delay(15); // software debounce

        if (digitalRead(button) == HIGH) {
```

```
    // if the switch is HIGH, ie. pushed down - change the lights!

    changeLights();

    delay(15000); // wait for 15 seconds

  }

 }

}

void changeLights(){

  // green off, orange on for 3 seconds

  digitalWrite(green, LOW);

  digitalWrite(orange, HIGH);

  delay(3000);

  // turn off orange, then turn red on for 5 seconds

  digitalWrite(orange, LOW);

  digitalWrite(red, HIGH);

  delay(5000);

  // red and orange on for 2 seconds (red is already on though)

  digitalWrite(orange, HIGH);

  delay(2000);

  // turn off red and orange, then turn on green

  digitalWrite(orange, LOW);

  digitalWrite(red, LOW);

  digitalWrite(green, HIGH);
```

delay(3000);

}

9.1 BEGINNING OF THE SEQUENCE - explanation
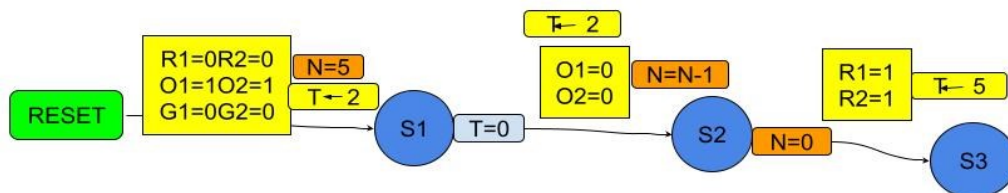### 9.1 SEQUENCE 1 - 3

Figure 44SEQUENCE 1 - 3

Stop time = millis()+2000 ;   // action setting of timer T=2

N=5   // action small violet rect

digitalwrite (r1pin, 0)   // action the big rect

digitalwrite (01pin, 1)

digitalwrite (g1pin, 0)

S=1   // arrow from RESET to s1

```
void loop() {

 if (S==1){if (millis()>=stoptime) {digitalWrite(o1Pin, 0);

                        stoptime = millis () + 2000;

                        N=N-1;

                        S=2;


                        }


 if (S==2){if (millis()>=stoptime) {digitalWrite(o1Pin, 1);

                        stoptime = millis () + 2000;

                        S=1;

                        }

        if (N==0) {digitalWrite(r1Pin, 1);

                S=3;
```

The sequence of signal first started with orange light at the stage o1=0 and N=N-1.

At the stage two the signal is o1=1 and r1=1, the are variation of the timing.
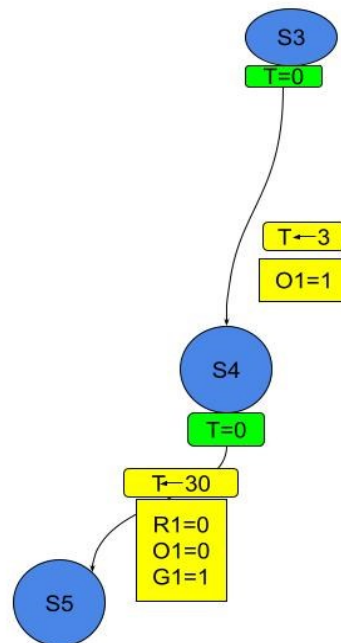
9.2 SEQUENCE 3 - 5

Figure 45SEQUENCE 3 - 5

The sequence of signal three light at the stage o1=1 the are variation of the timing.

The sequence of signal four light at the stage o1=0 the time variation is 30s.

```
if (S==3){if (millis()>=stoptime) {digitalWrite(o1Pin, 1);

                    stoptime = millis () + 2000;

                 S=3;

               }
        if (N==0) {digitalWrite();

             S=4;


if (S==4){if (millis()>=stoptime) {digitalWrite(o1Pin, 0);
```

stoptime = millis () + 2000;

S=30;

}

if (N==0) {digitalWrite(g1pin, 1);

S=5;

9.3 SEQUENCE 5 - 7

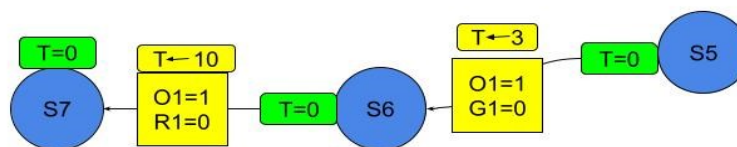

Figure 46SEQUENCE 5 - 7

The sequence of signal fifth light at the stage o1=1 the are variation of the timing 3s.

The sequence of signal sixth light at the stage o1=0 the time variation is 10s.


```
if (S==5){if (millis()>=stoptime) {digitalWrite(o1Pin, 1);

                stoptime = millis () + 2000;

                S=3;

                }
        if (N==0) {digitalWrite(g1pin, 0);
```

S=6;

if (S==6){if (millis()>=stoptime) {digitalWrite(o1Pin, 0);

stoptime = millis () + 2000;

S=10;

}

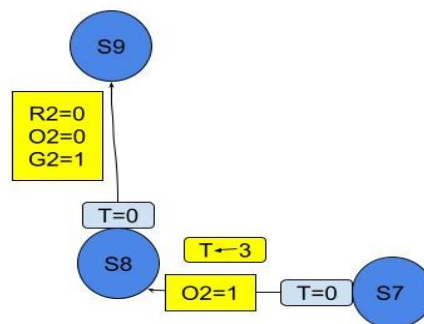if (N==0) {digitalWrite(r1pin, 0);

S=7;

9.4 SEQUENCE 7 - 9

Figure 47SEQUENCE 7 - 9

The sequence of signal seventh light at the stage o1=0 the are variation of the timing 3s.

The sequence of signal four light at the stage o1=0 the time variation is 1s.

if (S==7){if (millis()>=stoptime) {digitalWrite(o1Pin, 0);

stoptime = millis () + 2000;

```
                    S=3;

                }

        if (N==0) {digitalWrite();

                S=8;




 if (S==8){if (millis()>=stoptime) {digitalWrite(o1Pin, 0);

                    stoptime = millis () + 2000;

                S=1;

                }

        if (N==0) {digitalWrite();

                S=9;
```
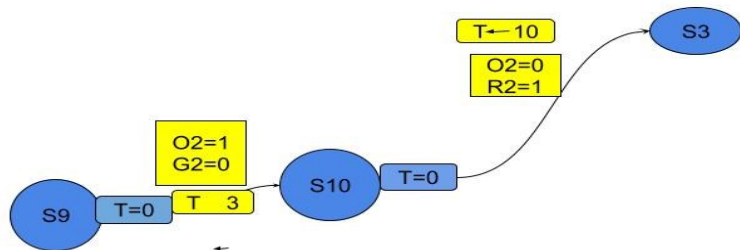
9.5 SEQUENCE 9 - 3

Figure 48SEQUENCE 9 - 3

The sequence of signal ninth light at the stage o2=1 the are variation of the timing 10s.

The sequence of signal tenth light at the stage o1=0 the time variation is 3s.

```
if (S==9){if (millis()>=stoptime) {digitalWrite(o2Pin, 1);

                        stoptime = millis () + 2000;

                        S=3;

                        }

        if (N==0) {digitalWrite();

                S=10;


if (S==10){if (millis()>=stoptime) {digitalWrite(o1Pin, 0);

                        stoptime = millis () + 2000;

                        S=10;

                        }

        if (N==0) {digitalWrite(r2pin, 1);

                S=3;
```

```
if (S==9){if (millis()>=stoptime) {digitalWrite(o2Pin, 1);

                        stoptime = millis () + 2000;
```

10.CONCLUSION

Finally, I have successfully completed my project. The model of the crossroad with traffic lights controlled by independent Arduinos was created. This control system could be changed to using at real crossroad. The led diodes would be replaced with amplifiers and bigger lights. The main aim was to use this system to traffic control during repair of the road.

Several strategies were designed. First strategy was timing of lights only, when each direction has green for adjusted time. This strategy was improved using sensors of the car. In this case arriving car can decrease the time to green light – strategy 2. Next two strategies replaces adjusted time to green forever of till the car in red direction appears. In this moment it takes green for itself and in strategy 3 it returns to previous direction after adjusted time or in strategy 4 the green stays forever or to next car at this road.

There were two control systems designed and created. One system is one Arduino which controls all lights, all four masts. It is not usable in real situation. Then the second control system was developed with four independent arduinos communication via wifi messages.

The education model was created when one sample of the mast was developed in CAD Catia system and printed on 3D printer.

REFERENCES

1. "New York's award-winning traffic control system". *ITS International. January– February 2013*. Retrieved 3 May 2014.

2. Vlasov, A., 2017. Features of Calculation of Traffic Light Control Modes in the Conditions of Intensive Road Traffic. Transportation Research Procedia, 20, pp.676-682.

3. Wen, W., 2008. automatic traffic light control expert system for solving the road congestion problem. Expert Systems with Applications, 34(4), pp.2370-2381.

4. Sahu, S., Paul, D. and Senthilmurugan, S., 2018. Density Based Traffic Signal Control Using DArduino And IR Sensors.

5. M. A.A. Parkhi, Mr. A.A. Peshattiwar, Mr. K.G. Pande "Intelligent Traffic System Using Vehicle Density". Yeshwantrao Chavan College of Eng., Nagpur. International Journal of Electrical and Electronic Engineers, 2016.

6. *Robinson, Larry. "Traffic Signal Progression". Retrieved 22 May 2014.*

7. Madisa, M.K. and Joseph, M.K., 2018, August. Android and Cloud Based Traffic Control System. In 2018 International Conference on Advances in Big Data, Computing and Data Communication Systems (icABCD) (pp. 1-4). IEEE.

8. *"Traffic Control and Traffic Signals". Wolverhampton City Council. Archived from the original on 20 September 2008. Retrieved 1 July 2008.*

9. "The Traffic Signs Regulations and General Directions 2016". *legislation.gov.uk. The National Archives. 2016*. Retrieved 14 July2017.

10. *"Traffic Lights Sequence". drivingtesttips.biz.*