IoT Enabled Smart Traffic Management System

Muhammad Moaz Awan
Department of Electrical Engineering
National University of Computer and Emerging Sciences
Lahore, Pakistan
l191367@lhr.nu.edu.pk

Hafiz Ahmad
Department of Electrical Engineering
National University of Computer and Emerging Sciences
Lahore, Pakistan
l191316@lhr.nu.edu.pk

Shazib Qasier
Department of Electrical Engineering
National University of Computer and Emerging Sciences
Lahore, Pakistan
i180921@nu.edu.pk

Hassaan Ahmad
Department of Electrical Engineering
National University of Computer and Emerging Sciences
Lahore, Pakistan
l191287@lhr.nu.edu.pk

Abstract*— *Transportation has existed for thousands of years. Hence, infrastructure for transportation plays a major key role in most of the metropolitan cities, leading to a change in working strategies. These changes are necessary and are made by keeping various aspects that are needed for the city and it results in being a smart city. Due to industrial establishments, IT parks and other related reasons the population density of the urban area increases in vast numbers. This in turn shoots in the number of vehicles in the urban as commutation resulting in the traffic congestions. The present road infrastructures are not able to appease the rapidly increasing number of automotive mobility on the roads. One of the major infrastructures for mobility is traffic signal management. Currently, the conventional method of the signal controlling that fails to a larger extent in managing the time. A modern-era solution is proposed here by employing the Internet of Things (IoT) that was achieved here by integrating several sensors to the microcontroller. The proposed solutions are favorable effects in reduction of delay, fuel consumption, wastage of energy, pollution from vehicles. To validate the proposed approach a prototype model was demonstrated and it was specifically designed for public and emergency vehicles such as fire brigade and ambulance that had to be needed highest priority at the time of mobility. Hence, a completely automated and intelligent IoT based traffic system is developed and proposed here.

Keywords*—*Transportation, Traffic signal, IoT, Mobility, Smart city.

I. INTRODUCTION

A recent statistic shows that Transport accounts for more than 40% of the air pollution produced in Pakistan, according to data from the country's climate change ministry.Researchers also stated that the traffic jam lost 32 lakh business hours per day [1]. The congestion for hours causes different severe types of pollution. Consequently, people are suffering from any kind of physical and mental discomfort [2].

As the population increases across the globe. Traffic management has become a big issue. An average person spends more than 58 hours per year on traffic signals which is a lot of time that goes to waste. The problem does not end just there. Due to long stops at red light a large amount of fuel is also consumed and wasted by automobiles at red lights which causes a lot of air pollution as well due to all the fumes coming out of the exhaust systems. A lot of the times red lights cause traffic congestion as one side of the traffic signal sees high density traffic while the other sees almost no traffic at all in comparison. People often witness scenarios where they are stopped at the red traffic signal for absolutely no reason as all the other approaching roads are completely empty. To get rid of this unwanted congestion, Pakistan needs a system that will overcome the problems of the existing system.

Due to the above-mentioned issues, there is an increasing need for a smart traffic signal management system that monitors traffic density on approaching roads to an intersection and control the traffic signals accordingly.

In these days the static traffic lighting system at traffic junctions has drastically increased the traffic congestion and frustration for drivers while waiting for the statically defined time even at the non-peak hours also wasting the resources like power, time and fuel. Hence this work is proposing an efficient approach to manage the traffic at the junctions dynamically based on the density of vehicles using IoT technology.

So, a smart traffic system can be a solution to these problems. Controlling the vehicle with the help of the traditional traffic light with the combination of the sensors and artificial intelligence is known as the smart traffic system [3]. The proposed smart TMS measures traffic density by analysing sensor data. It sets the timing for traffic signal light by traffic management algorithm. Traffic signals is visualised and controlled by a mobile application connected to cloud. Raspberry pi will sen the status of Traffic signals to the cloud and will also handle the mobile application commands.

Android Things is a lightweight version of Android published by Google that can be flashed onto different hardware prototyping boards, such as the Raspberry Pi 4, Intel Edison, and NXP Pico i.MX7D. Android Things extends the core Android framework with additional APIs that allow apps to integrate with new types of hardware not found on mobile devices. Apps for embedded devices bring developers closer to hardware peripherals and drivers than phones and tablets. By integrating the IoT capabilities of Android Things with MIT App Inventor, we eliminate the need for heavy and complex programming code by simply enabling the visual and interactive block-based drag and drop programming tools to create IoT applications. In the case of input sensors connected to the Raspberry Pi device or any other Android

Things supported device, the data from these sensors can be relayed to the phone app or to an external storage device for the post or real-time data analysis purposes to obtain useful insights to make decisions. The integration of these two technologies (MIT App Inventor and Android Things) with a simple, userfriendly environment enables a larger audience of developers to create applications that can collect a vast amount of data in the fast-growing IoT space.

The prime objective of this paper is to develop a smart traffic management system (TMS) using IoT. To the best of our knowledge, smart traffic system is not implemented in Pakistan yet. This whole approach will cost less than other approaches. Moreover, operating the proposed system is more comfortable than other existing systems.

II. LITRATURE SURVEY

The review gives a short examination of existing procedures in the demonstrating of the traffic framework. In reference [4], proposes a framework that will quantify the traffic upheld the amount of auto thickness of the vehicles inside the street. The control framework offers proficient administration of traffic and dependability above the overall frameworks by effectively utilizing Raspberry Pi pigs as the canter principle. In [5], the authors dealt with a brilliant traffic light control framework that is actualized and structured, it bolsters savvy city transportation for vehicle applications including crisis vehicles, open vehicle, and other traffic light control, for good driving support, and message broadcasting. In [6], a brilliant traffic signals framework is put forward that utilizes an Android application, MQTT convention, Google maps, microcontroller traffic signals, and the Web for associating them that can permit crisis reaction vehicles to securely cross the traffic signals immediately by utilizing Android hogs as the canter principle. An advanced technique for recreation and test framework for a vehicle, at that point an answer calculation dependent on SUMO re-enactment is proposed [7].

The idea is to gathering traffic information from the resources of a particular course to maintain a strategic distance from blockage. This framework gives another approach to control the gridlock design around the traffic circle by building a calculation that controls traffic lights before a blockage occasionally happens. In [8], suggests a system for supervising workforce departure and fiasco goal in crises and improving the urban crisis help the executive design, Web of Things, and information mining innovation, a traffic crisis reaction design for convenient open and crisis vehicles. According to [9] the infrastructure of a smart city can be provided accurately, quickly, and efficiently reset the workforce and vehicle progress, proper road facility to the government vehicle, oversee follow-up salvage tasks, and adequately improve salvage productivity and urban governance levels.

In [10], the proposed framework depends on the developed idea since it gathers data and offers data with others, so they need handy choices. The principle motivation behind the dynamic control framework is to supplant the current arrangement of traffic lights in India. In the current framework, road lights are constrained by time-change markers. In [11], the proposed framework will not follow and effectively control vehicles. Such frameworks diminish the labor as well as give precise outcomes. In [12], the design of the independent and smart traffic framework is introduced. It controls the traffic expels clog. A microcontroller discovers current traffic thickness at the stoplight and doles out opening time. In addition, the crisis vehicle is distinguished by utilizing the recurrence recognizable proof.

In [13], the proposed scheme adaptability of alteration consistently. Time spans for the green, orange, and red signs are fixed then the holding up time is more, so structured this sign controlling for the traffic, for more proficiently with another procedure called Astute Control Framework. By utilizing the sensors along with implanted innovation, the timings of the green lights and red light are shrewdly chosen to help the traffic on the streets easily. IoT platform makes the resulting device more versatile in any domain of application [14].

Firebase is a Google product and gives us many services, like Firebase Database, Firebase Authentication, Firebase Functions, Firebase Storage, etc. Hence it enables us to easily customize and add features to a project.The Notification is the main feature of our application. We can receive alerts for various events, like burglar alert, fire alerts, etc. This is easily customizable as this notification is triggered by Firebase Functions and is sent through Firebase Messaging Services [15].

MIT App Inventor [16] is an open-source app-building platform that allows users to drag-and-drop visual objects to create an application that can run on the Android system as a means of democratizing mobile app development. Application behavior is provided by piecing together blocks in a visual blocksbased programming language. MIT App Inventor is used by students, teachers, developers, hobbyists, and entrepreneurs to develop apps for collaboration, productivity, personal use, recreation, learning, social good, and community activism. The ease of use of the system as a tool for making has contributed to its success and over 10 million people have used MIT App Inventor worldwide to create 43 million projects.

By referring all the existed works and existing methodologies. An overview of traffic sensing technologies in the domain of wireless communication with the suitable hardware is not user-friendly nature for a virtual view of the traffic & the data acquiring, storing in the cloud. By referring to the Mit app inventor based platform capable of providing connectivity with cloud storage and graphical user interface for a virtual view of the traffic and by the help of radio-frequency identification sensing technology it can control the traffic signals remotly.

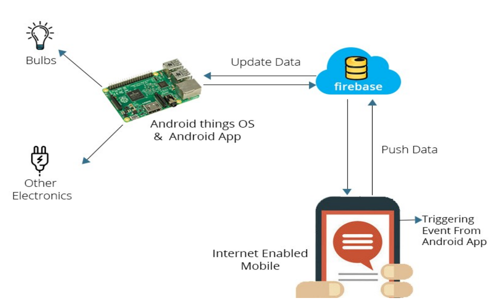

Fig. 1. System model

III. METHODOLOGY

For the system implementation Raspberry pi is being used which is a single board computer used for IOT applications. The raspberry pi OS was downloaded from the raspberry pi website, extracted, and flashed in the raspberry pi 4 through an SD card.After installing the OS, we enabled the VNC in Raspberrypi so that we can control it with the hep of our laptops. After this setup, VNC viewer is installed in the laptop and the IP address of the raspberry pi is entered with the default credentials. Doing so will open the raspberry pi desktop in a VNC window where the GUI is accessible.

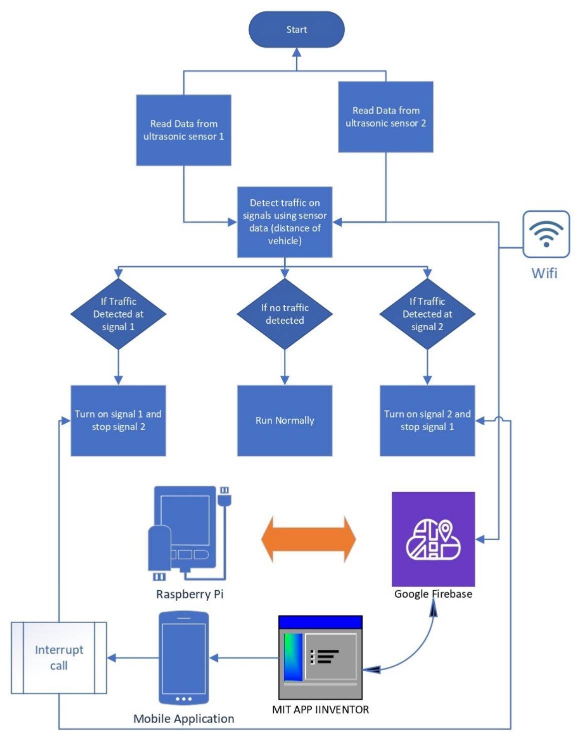

Fig. 2. Flow Chart

At the very first we connected three LED lights with the raspberry pi, representing a traffic light setup in order to check that raspberry pi GPIO pins are in working condition. A resistor was used with every LED for the purpose of system protection. The LED lights were operated in a manner that the red light was turned on for ten second, the orange light was turned on for two seconds and the green light was turned on for ten seconds, and this pattern was repeated in a cycle using an infinite loop.

After this, another set of traffic lights were interfaced with the raspberry pi, like the first set . This setup represented an intersection. The DistanceSensor and threading libraries were imported. We added two ultrasonic sensors, one on each signal, to detect traffic. When there was no vehicle detected on both sides, the traffic signal continued its normal operation where it switches lights after a specific time interval. As soon as a vehicle is detected on one side by the ultrasonic sensor, the systems check whether there is a vehicle is present on the other side or not. If there is no vehicle on the other side, the signal on which the vehicle is detected is turned to green and the other signal turns red. The same is repeated if a vehicle is detected on the other side only. In a case where vehicles are detected by the ultrasonic sensors on both traffic signals, the system operates in its normal state and switches after fixed time periods.

Algorithm: Detection of Traffic using ultrasonic sensor distance threshold and real time controlling and monitoring using monile application and cloud.

---

1. normarl()
   // operate 2 traffic singal in normal condition
   // updating data on cloud
   //cloud linkage with mobile app
2. thread1()=distance1
   // return distance from ultrasonic senor 1
3. thread2()=distance2
   //return distance from ultrasonic senor 2
4. thread3()=signal1_call_from_cloud
   // checkig interrupt is call by mobile app
5. thread4()=signal2_call_from_cloud
   //return distance from ultrasonic senor 2
6. start:
   thread1()
   thread2()
   thread3()
   thread4()

7. **if** *distance1<threshold* i.e traffic detected
8. **then**
   a. Turn green signal 1 and red on signal 2

9. **if** *distance2<threshold* i.e traffic detected
10. **then**
    a. Turn green signal 2 and red on signal 1

11. **else** go to step start

The second set of LED lights were interfaced to the GPIO pins 10, 9 and 11 of the raspberry pi. The first ultrasonic sensor had its echo at pin 24 and trigger at pin 23 while the second sensor had echo at pin 16 and trigger at pin 26. Firstly all the LED pins were set to low using the rest function. In the normal operation, Red LED of one side and Green LED of the other side was turned on and delay was added for eight seconds. After eight seconds, the Yellow LED for both sides are turned on for an interval of three seconds and the previous LED lights are set to low. After three seconds, the Green LED of one side and Red LED of the other side are lit up for eight seconds, after which, again the Yellow LED on both sides are turned on for three seconds.

For the execution of the required functionality, two threads were used. In the threads, interrupt checking was placed for both of the ultrasonic sensors. Whenever a vehicle is detected, an interrupt is triggered, causing the traffic signal to change its operating state. To operate the interrupt, an output signal had to be generated and then detected using two different pins of the raspberry pi as the raspberry pi does not detect internal interrupts and therefore, we had to use external interrupts. For the interrupt of first sensor, GPIO pin 18 was set as output and GPIO pin 15 was set as input pin, while for the seconds sensor, GPIO pin 5 was the output and GPIO pin 6 was the input pin.

Whenever a vehicle is detected at a sensor, its output pin is set to high and the input pin gets a high input which triggers the interrupt and calls the respective my_callback function. When the my_callback function is called, the respective signal is set to green is there is no vehicle on the other side. The ultra1 and ultra2 functions received inputs from the first and second ultrasonic sensor respectively and multiplied their distance values with hundred to get realistic value. If the distance received is less than fifteen centimeters, it means that a vehicle is detected, so the output pin for sensor is set to low which will then trigger the interrupt. If the distance is greater than fifteen centimeters, it means that there is no vehicle detected and it sets the output pin for this sensor to high.

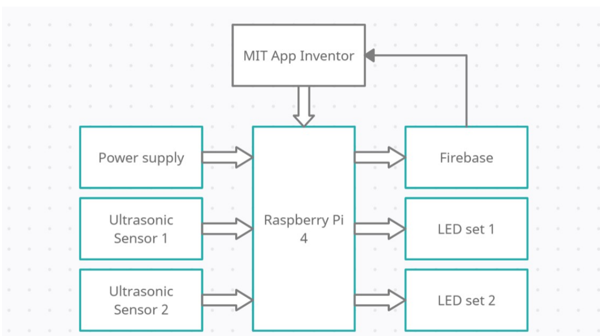

Fig. 3. Block diagram

In the final phase of this project, the raspberry pi was integrated with the cloud and mobile applications. For this purpose pyrebase library was imported. The values of both the ultrasonic sensors as well as the current states of both the traffic signals was uploaded to the cloud and and cloud is linked with mobile app invented by MIT app inventor visible on the mobile application. In this phase, all the states of the traffic signals were given a unique ID in the form of the variable number 'number'. In the first state where signal of the first side is red while that of the other side is green, the value of 'number' for this state is set to 1. In the state where the yellow lights are lit on both sides, for this state the value of 'number' is set to 2. For the state where the signal on the first side is green while that of the other side is red, the value of 'number' is set to 3. After every change in state, the value is updated on the firebase by sending variable 'number' as data to the update function for the firebase.

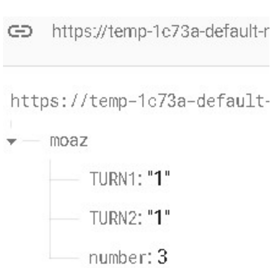

Fig. 4. Firebase data

Moreover, the system was also capable of controlling both traffic signals manually through the mobile application. For this phase we utilized the firebase server and MIT mobile application inventor. First we create a firebase web server and generate an API URL to read or write our channel. After this, we go to the MIT app inventor site and create a mobile application for out traffic signal. After everything is set up, we place the firebase API URL in the URL block on the MIT app inventor site.

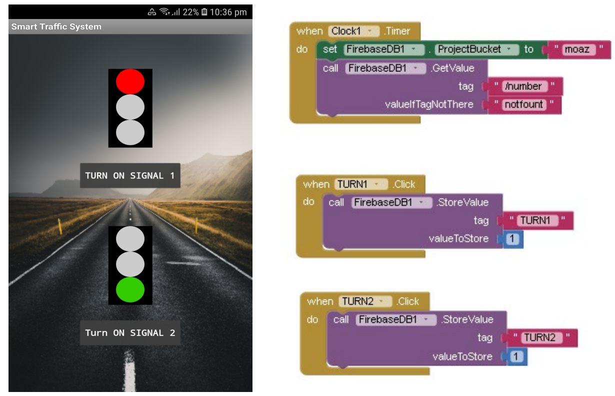

Fig. 5. Mobile application GUI and backend

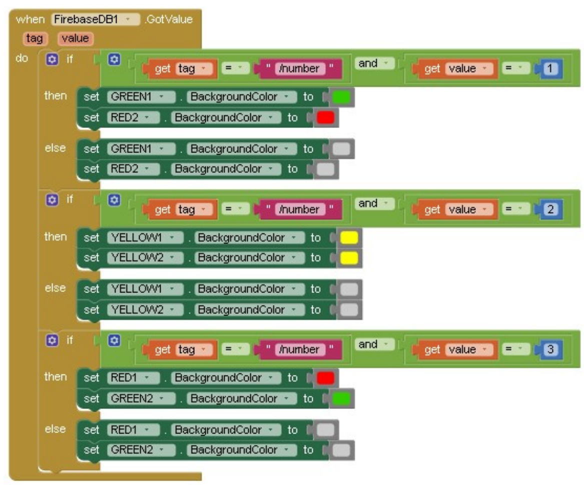

Fig. 6. Application backend

After the application is successfully created, we scan the QR code on the screen from our mobile device and the traffic light status as well as manual controls show up on the screen, enabling us to monitor as well as manually control the system remotely. To get this remote access, two more threads were used. In both of these threads, data was being received from the firebase which was connected to the mobile application. When the user sets a signal to green from the mobile application, it sets the value of the respective TURN variable to 1. In the threads, the firebase is being checked continuously if this value is set to 1 or not. If the value is set to 1, the output pin 18 for the first signal or the output pin 5 for the second signal is set to low, which then triggers the interrupt for the respected side. If the value is not equal to 1, it sets the output pins for both signals to high.

IV. RESULTS

As a result of this project, we have a system that can make our lives easier and greatly reduce waiting times, fuel and time wastage as well as a massive reduction in air pollution due to exhaust gasses of vehicles. This system automatically senses vehicles and operates the traffic signals accordingly while uploading all the data to the cloud and is controllable from a mobile application for manual use. This system helps reduce traffic congestions and is capable of handling more advancements in the future, in terms of feature addition and accuracy improvements.

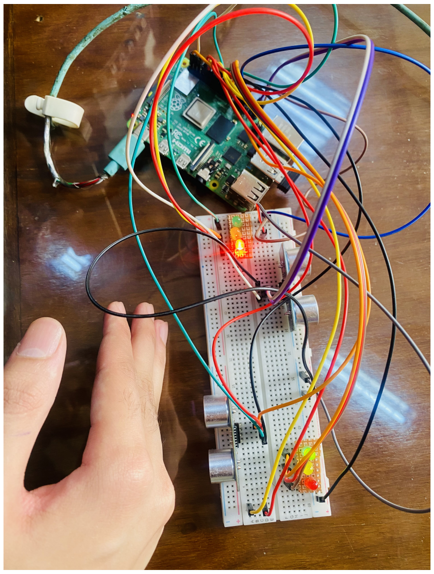

Fig. 7. output of trrafic detected signal

As, we can turn on any signal using an a mobile application and visualize it also on mobile application so, this will help in controlling the traffic signals remotely without being present physically on the traffic signal terminal to operate it manually.

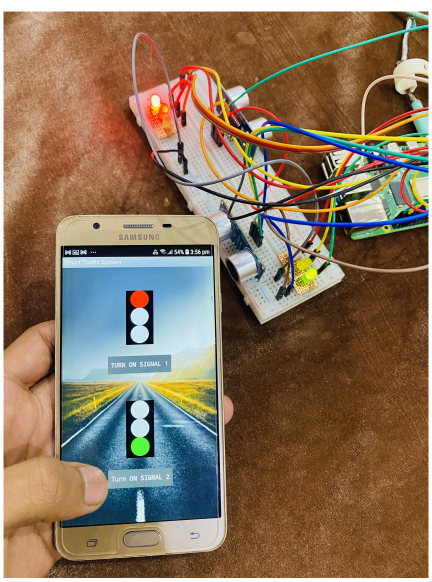

Fig. 8. Mobile application visualizing and controlling the traffic signal

V. CONCLUSION

We can draw the conclusion from this that IOT has the potential to significantly improve our daily lives. We can make our lives easier and safer by resolving numerous issues that we may or may not have been aware existed with the assistance of IoT devices-based frameworks like this undertaking can assist with decreasing a ton of stress and time wastage from our lives as well as significantly lessen the general carbon impression. We can control a lot of resource waste and improve system efficiency by making everyday devices and systems smart. The findings of this project make it abundantly clear that IoT-based systems are, in fact, the technological world's future, and that the majority of systems and devices will soon be replaced by smart IoT-based devices.

ACKNOWLEDGMENT

We would like to express our gratitude to my teacher, Prof. Dr. Mohsin Kamal, who gave us the opportunity to complete this wonderful project on the topic of "A Smart IOT based Traffic Management System for Future Smart Cities." Not only did this project assist us in resolving a real-world traffic management system issue, but it also taught us a great deal of new information, and we are extremely grateful to them..

REFERENCES

[1] a. Samad, "'Traffic jam' silent killer," in The Asian Age, ed., 2016, January 28

[2] A. Khan, "Problems caused by traffic jam " in Daily Observer, ed, 2016, Maech 30K. Elissa, "Title of paper if known," unpublished.

[3] P. Jadhav, P. Kelkar, K. Patil, and S. Thorat, "Smart traffic control system using image processing," Int. Res. J. Eng. Technol, vol. 3, 2016.

[4] A. A M, K. Shivanand Bhat, A. G, B. J and A. P, "Density Based Smart Traffic Control System for Congregating Traffic Information," 2019 International Conference on Intelligent Computing and Control Systems (ICCS), Madurai, India, 2019, pp. 760-763, doi: 10.1109/ICCS45141.2019.9065691. IEEE Xplore: 16 April 2020.

[5] Lee, W.-H.; Chiu, C.-Y. "Design and Implementation of a Smart Traffic Signal Control System for Smart City Applications". Sensors 2020, 20, 508. doi.org/10.3390/s20020508.

[6] K. M. Almuraykhi and M. Akhlaq, "STLS: Smart Traffic Lights System for Emergency Response Vehicles," 2019 International Conference on Computer and Information Sciences (ICCIS), Sakaka, Saudi Arabia, 2019, pp. 1-6, doi: 10.1109/ICCISci.2019.8716429. IEEE Xplore: 16 May 2019.

[7] H. K. S. K. Hettikankanama and S. Vasanthapriyan, "Integrating Smart Transportation System for a Proposed Smart City: A Mapping Study," 2019 International Research Conference on Smart Computing and Systems Engineering (SCSE), Colombo, Sri Lanka, 2019, pp. 196-203, doi: 10.23919/SCSE.2019.8842743. IEEE Xplore: 19 September 2019.

[8] S. Boudaakat, A. Rebbani and O. Bouattane, "Smart Traffic Control System for Decreasing Traffic Congestion," 2019 International Conference on Systems of Collaboration Big Data, Internet of Things & Security (SysCoBIoTS), Casablanca, Morocco, 2019, pp. 1-6, doi: 10.1109/SysCoBIoTS48768.2019.9028014. IEEE Xplore: 19 September 2019

[9] P. Faldu, N. Doshi and R. Patel, "Real Time Adaptive Tra c Control System : A Hybrid Approach," 2019 IEEE 4th International Conference on Computer and Communication Systems (ICCCS), Singapore, 2019, pp. 697-701, doi: 10.1109/C-COMS.2019.8821749. EEE Xplore: 02 September 2019

[10] B. Ghazal, K. ElKhatib, K. Chahine and M. Kherfan, "Smart traffic light control system," 2016 Third International Conference on Electrical, Electronics, Computer Engineering and their Applications

(EECEA), Beirut, 2016, pp. 140-145, doi: 10.1109/EECEA.2016.7470780. IEEE Xplore: 19 May 2016.

[11] Parekh S., Dhami N., Patel S., Undavia J. (2019) Traffic Signal Automation Through IoT by Sensing and Detecting Traffic Intensity Through IR Sensors. In: Satapathy S., Joshi A. (eds) Information and Communication Technology for Intelligent Systems. Smart Innovation, Systems and Technologies, vol 106. Springer, Singapore. doi.org/10.1007/978-981-13-1742-2_6.

[12] P. Sadhukhan and F. Gazi, "An IoT based Intelligent Tra c Congestion Control System for Road Crossings," 2018 International Conference on Communication, Computing and Internet of Things (IC3IoT), Chennai, India, 2018, pp. 371-375, doi: 10.1109/IC3IoT.2018.8668131.

[13] Abdul Mateen, Sabeen Sher, Muhammad Zahid Hanif, and Tooba Akhtar, "AutonomousControllers for Urban Traffic Management," Journal of Traffic and Logistics Engineering, Vol. 7, No. 2, pp. 47-52, December 2019. doi: 10.18178/jtle.7.2.47-52.

[14] R. Shashidhar, S. Abhilash, V. Sahana, N.A. Alok, M. Roopa, "Iot cloud: In health monitoring system", Int. J. Sci. Technol. Res. 9 (2020) 227–230

[15] Firebase Messaging Service [Online] .Available: https://firebase.google. com/docs/cloud-messaging/

[16] S. C. Pokress and J. J. D. Veiga, “MIT App Inventor: Enabling personal mobile computing,” arXiv preprint arXiv:1310.2830, 2013.