ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal
Regular Issue, Vol. 9 N. 3 (2020), 89-104
eISSN: 2255-2863
DOI: https://doi.org/10.14201/ADCAIJ20209389104

Intelligent Traffic Light for Emergency Vehicles Clearance

Raneem Nono, Rawan Alsudais, Raghad Alshmrani, Sumayyah Alamoudi and Asia Aljahdali*

College of Computer Science and Engineering, University of Jeddah, Saudi Arabia
rnono0001.stu@uj.edu.sa,ralsudais.stu@uj.edu.sa,ralshmrani0007.stu@uj.edu.sa, salamoudi0068.stu@uj.edu.sa, aoaljahdali@uj.edu.sa*

KEYWORD

Traffic light; Arduino; ZigBee; Embedded system; Intelligent tarffic.

ABSTRACT

Human life is a serious matter, so we should not neglect anything that might threaten it. It must be protected in all possible ways. Consequently, all health services such as hospitals, medicines, ambulances and so on need to evolve continuously to overcome life-threatening problems. Since many people could lose their life because of an ambulance delay. We proposed a system that provides a way to overcome the ambulance delay problem. With the current traffic light system, the ambulance can get stuck in the traffic or may cause an accident while it crosses the red light. To avoid that, the proposed system enables the ambulance to control the traffic light. When comparing the estimated time required for an emergency vehicle to move from location A to location B while passing two traffic lights, it would take 10 minutes with the current traffic light system based on the GPS assuming no traffic delay. While with the proposed system, it would take 8 min to pass the same distance. So the difference is 1 min for each traffic light on the way. Thus, the system will facilitate the emergency vehicle movement to save people's life. The hardware used to implement this project includes Arduino UNO and mega with network shield (ZigBee).

1. Introduction

Saudi Arabia is one of the biggest countries in the middle east region. The population as estimated in mid-2016 is 31.78 million and it keeps increasing (Azalghamdi01, 2017) . Most Saudis use their private vehicles for transportation because the lack of public transportation. Recently, Saudi government allows women to drive. As a result, the number of vehicles on roads have been increased. Consequently, traffic is becoming a serious problem.

In big cities due to the traffic congestion, emergency vehicles such as ambulance, fire engines are affected by traffic jams and consequently many people could lose their lives because of an ambulance delay. Although the emergency vehicles in Saudi Arabia have the right to pass red lights and exceed the speed limit on roads to reach the patient, but this adds another problem and it might cause farther accidents.

The proposed system would save people life and the environment from the consequences of emergency vehicle delay. Furthermore, it saves the emergency vehicle passengers from any accident that would result from crossing the red light. Also, it avoids the time wasted by waiting the emergency vehicle for the red light to turn off and avoiding forcing cars in front of it to cross the red light to enable passing the emergency v vehicle. In addition to the mentioned above when an emergency vehicle has to pass a distance from A to B, it would save the duration time of the red light for each traffic light in the that way from A to B.

This project proposed a solution for this problem by proposing an emergency mode to the current traffic light system, which gives ambulance the priority to pass the traffic light to arrive to patients and hospitals smoothly. The rest of the paper is organized as follows. Section 2 provides a background sight about the domain that the proposed system is covered. Section 3 discusses related work to our project. Section 4 presents our proposed solution. The system design and implementation are presented in section 5 and 6. Section provides the paper's conclusion and future work.

2. Related Work

This section presents several techniques proposed in different papers in the same domain of the proposed system and discusses their features. Also, it summarizes the similarities and the differences between these systems.

Intelligent ambulance system proposed in (G. Beri et al., 2016) combines two important systems that helps for saving lives, vital sign monitoring system and traffic control system using Advanced Virtual RISC (AVR) microcontroller. Vital sign monitoring is a system that records patient's important vital signs and sends it at real time to the hospital server via personal computer (PC) using serial communications. Health parameters that the system records are electrocardiogram (ECG), heart rate, and body temperature. The importance of electrocardiogram record presences in case of heart attack patients. Electrocardiogram (ECG) uses electrodes that is placed on the patient's body to measure the contraction and relaxation of the heart. Heart rate record is measured when the patient's finger is placed in an instrument that uses infrared Light Emitting Diode (LED) that transmits signal that is reflected by the patient's blood plasma in the finger. Body temperature is recorded using temperature sensor. On the other hand, traffic control system allows ambulance driver to control the traffic light. This kind of control is handled through a keypad by the ambulance driver by choosing the path that it will pass. Once ambulance driver presses the keypad, RF (Radio Frequency) transmitter, which is in the ambulance, will send a signal (binary signal) to the RF receiver that it is in the traffic light controller. This signal contains information about an ambulance location, path, and the traffic congestion. When the signal is received by RF receiver, it will be decoded, if the signal sent by an ambulance the traffic light sequence will be interrupted and emergency mode will be executed. In the emergency mode, traffic light will give the priority to the ambulance to pass by turning on the green light to the ambulance lane and all other lanes traffic lights will be red. After the ambulance passing, the traffic light will resume its original flow. Researchers from Vishwakarma institute of technology decide to use Radio Frequency (RF) module instead of using InfraRed (IR) because IR signals can travel only in short distance while RF signals can travel through wider distance. Also, IR signal cannot travel in case of path obstacles, while RF signal can travel even when there are obstacles between the transmitter and the receiver. Moreover, IR signals can be affected by other IR transmitting sources. However, RF signals are more reliable and stronger than IR signals. This system can be improved by using actual Global Positioning System (GPS) navigation system that may help in improving congestion detection. The intelligent ambulance research concentrates on clearing the path for an ambulance by controlling traffic lights using RF signals and monitoring patient's important vital signs then sending it to the hospital server so the patient can get the suitable treatment within time (G. Beri et al., 2016) .

A smart ambulance system in (D. S. Reddy and V. Khare, 2017) is used to provide clearance to any emergency ambulance vehicle by turning all the red lights to green on the path of the emergency vehicle. The system is implemented using GPS, Global System for Mobile Communication (GSM) and ZigBee technology, every few minutes the system sends patient parameters to the hospital to monitors patient condition. The microcontroller unit is connected to the ambulance control room to send the details control center and traffic signal. The system uses ARM Cortex-M32 as interfaced with traffic signal and ambulance section, ARM Cortex-M3 offers system enhancements and a higher level of support block integration. This smart system is low cost system, due to ZigBee technology, which is less expensive than other WPANs, GPS is freely available to all and GSM. The primary objective is to recognize emergency vehicle and track its location to provide wireless signal to the emergency vehicle. Ordinary technologies use image processing to recognize the emergency vehicle, but this technology is affected by the weather conditions, which prevents recognize the emergency vehicle. Thus, this smart system is using ZigBee transponder and receiver which works will in all weather conditions. In the smart system high frequency reader is needed to provide long range. ZigBee transponder should be embedded inside the dashboard of the vehicle. The system consists of three sections: ambulance, control center and traffic signal. The smart ambulance sends the location and the patient parameters to the control center. Control center sends the path to the nearest hospital to the smart vehicle, the ambulance will choose the path, and every traffic signal within this direction will be green (D. S. Reddy and V. Khare, 2017) .

Paper (R. Sundar et al., 2015) presents an intelligent traffic control system that allows emergency vehicles to pass easily. The system is consisting of three parts, and the technologies used in the system are ZigBee, radio frequency identification (RFID) and global system of mobile communications (GSM). Part one is dealing with counting the number of vehicles that passes on a particular path during a specific time and the green light timing for the path. RFID tag is placed in each vehicle so when it passes the RFID reader will count the number for passed vehicles. The methodology for knowing the duration of the green light associating with the number of vehicles will be as the following; if the count is more than 10, the duration of the green light is set to 30 seconds. If the count is between 5 and 9, the green light duration is set to 20 seconds, and if it is less than 5, the duration is set to 10 seconds. Part two is handling the ambulance clearance, each ambulance has a ZigBee transmitter module and the ZigBee receiver is implemented in each traffic junction. The ZigBee receiver waits for a signal from a ZigBee transmitter which is implemented in each ambulance. When the ZigBee receives this signal, the traffic light will turn to green. The traffic light turns back to red as soon as the ambulance passed. Part three is presenting a method for stolen vehicle detection, the RFID reader detects a stolen vehicle, the module compares the unique RFID tag read by the RFID reader to the stolen RFIDs stored in the system. If a match is found, the traffic light is turned to red for 30 seconds. In addition, a SMS is sent to the police station (R. Sundar et al., 2015) .

The paper (H. Singh et al., 2012) provides a technique that controls the traffic light instead of the existing static traffic light. So, the traffic light time will change according to the number of cars and priority of the vehicle. Also, it makes the operation of detecting the violators of traffic rules more accurate and traces the stolen vehicle. In the proposed system, each intersection has four traffic light and data base. Each rode in the intersection divided into two lanes. A RFID reader is placed at each lane. So, there will be 8 RFID readers at each intersection. A RFID tag will be placed in each vehicle and a vehicle identification number (VIN) is stored. VIN consist of three parts: the first part indicates the vehicle's priority. Second part indicates the vehicle's type. Last part indicates the vehicle's number. The RFID reader will store vehicle's VIN and time stamp of each vehicle that pass by it. The time stamp is used to find the violators.

The system handles a vehicle according to its priority. The highest priority is for ambulance, fire brigade vehicles, and V.I.P vehicles. The second priority is for buses including college and school buses. Cars, motorcycles and scooters have the third priority. While the heavy vehicle has the fourth priority. At night the priority of heavy vehicle is higher than third priority. The traffic light controller uses round robin sequences. While when a high priority vehicle is detected by a RFID reader, the traffic light controller will interrupt the round robin schedule and display the green signal for this vehicle. The pseudo code for this system is: all lights will store in queue. If a RFID reader defected a high priority vehicle, an emergency signal will be sent to center traffic light controller. Then it finds the road that contains the RFID reader and turns the corresponding traffic light to green for that vehicle. If no high priority vehicle was detected, steps will operate on each picked traffic light. First, a traffic light will be picked from the queue. Second, At the picked traffic light the number of vehicles will be counted, and the type of vehicle will be checked. Third, if an emergency vehicle detected; the steps that mentioned above will be applied. Else, At the picked traffic light the priority of each vehicle will determine. Fourth, the duration of green light will be calculated based on the number of vehicles. Fifth, if there is a traffic light that does not pick for a time that exceed the limited time, it will give the turn (to prevent the starvation). The algorithm considers the traffic density, starvation, vehicle priority and queue length while it makes the decision about green signal displaying and its duration (H. Singh et al., 2012).

When analyzing the above techniques, we and see that the paper (G. Beri et al., 2016) and paper (H. Singh et al., 2012) use RFID to send a signal. However, the RFID is limited range and can be affected by the weather. While paper (D. S. Reddy and V. Khare, 2017) and paper (R. Sundar et al., 2015) use ZigBee to send a signal, which is better since the ZigBee with a stronger power source can increase its range. Additionally, the detect stolen vehicles in paper (R. Sundar et al., 2015) and detect traffic light violations in paper (H. Singh et al., 2012) will cost some money to put a tag in each car and it can be easily removed by anyone since there is no secure place to put it there. Also, paper (G. Beri et al., 2016) and paper (D. S. Reddy and V. Khare, 2017) presents monitoring patient's vital sings feature which we believe it is not that useful depending on our interview with the head of emergency department in king Abdulaziz hospital in Jeddah (Dr. Nader Gazzaz). We have discussed with him this feature to monitor patient's body temperature, heart rate and the electrocardiogram (ECG). However, the doctor said that it is not enough for emergency doctors to monitor only these three vital signs because important vital signs would be differed from one case to the other . Finally, in these papers there is no solution for having more than one emergency vehicle in the same junction at the same time. Table 1 gives a comparison between these techniques.

Table 1. Comparison between the discussed techniques.

	Intelligent ambulance system (2016)	A smart ambulance system (2017)	Intelligent traffic control system for congestion control (2015)	Intelligent traffic lights based on RFid (2012)	The proposed system
Control traffic light using AVR Micro-controller	Yes	No	No	No	No
Control traffic light using Arduino Microcontroller	No	No	No	No	Yes
Send signal to the traffic light using RFID	Yes	No	No	Yes	No
Send signal to the traffic light using ZigBee	No	Yes	Yes	No	Yes
Detect stolen vehicles	No	No	Yes	Yes	No
Monitor patient's vital sings	Yes	Yes	No	No	No
Detect Traffic light violations	No	No	No	Yes	No
Handle two ambu-lance in the same time and same junction	No	No	No	No	Yes

Based on the literature review all the all investigated systems did not handle an important case. The case is when more than one ambulance sends a signal at the same time in the same intersection. Our proposed system suggests a solution of such case; the proposed system enables the ambulance's driver to enter the severity level of the patient beside the target traffic light. So, the ambulance with a highest severity level will have the highest priority. The proposed system is using ZigBee and Arduino. We decide to use the ZigBee due to the fact that it can increase its range by increase the power source, while the RFID and IR are limited range. Also, the ZigBee does not affect by the weather which is unlike the RFID and IR. Regarding to the Arduino, it is a cross platform, it can be run on Windows, Macintosh OSX, and Linux operating system. It is inexpensive, easy to use, open source, and extensible hardware and software.

The proposed system allows users to select the severity level of the patient. Then, the use select the target traffic light (Choices:1=North, 2=East, 3=South, 4=West) that turns the emergency mode on (which gets green light on) to be able to send a signal to the traffic light controller, then back to normal mode after some fixed time. The hardware tools required to implement this project are two Arduino UNO microcontroller, two Xbee shields, four LED traffic light modules, male wires, breadboard, Keypad, and LCD screen. Basically, the system depends on Arduino Uno microcontrollers and LED traffic lights modules which are developed using Arduino IDE software. The activity diagram of the system is given in Fig. 1. The sequence of the activity diagram is as follows:

1. First the ambulance driver should choose the patient severity level and traffic light.
2. Send the signal from the Zigbee transmitter to the Zigbee receiver.
3. Receive the signal from the Zigbee transmitter which contain the severity level and the traffic light number.
4. The microcontroller in the traffic light unit will check the number of signals.
5. If more than one signal ,select the traffic light with the highest severity level.
6. If one signal ,check the statute of the selected traffic light.
   If red ,change the light to green.
   If green, increase the time of it .
7. Back to normal mode.

3. Technical Background

The world is entering a new period of computing technology which shows a huge expansion in a technology called Internet of Things (IoT). IoT is a giant network with connected devices; from air conditioners that can be controlled with your smartphone to smart cars providing the shortest route, or your smartwatch which is tracking your daily activates. These devices gather and share data to help in taking decisions (Hivemq, 2020) . The physical part of IoT devices includes lots of different things: for example, the engine, air conditioner, and navigation system in a smart car, refrigerator in a smart home and watches and fitness trackers. We can make these devices smart by using various sensors and microprocessors that enable advanced functionality. For example: the electronic control units in a smart car, motion activated cameras in a home security system, and wearable hypoglycemia sensors that automatically alert diabetic patients when their blood sugar levels are dangerously low. Finally, IoT devices are connected to the Internet and other systems for different purposes (J. Osborne, 2017) . Environment of IoT consist of hardware board called microcontroller. The most popular microcontrollers used in IoT are Arduino and Raspberry Pi. Arduino is an open-source electronics platform based on easy-to-use hardware and software (Arduino, 2020) . Arduino board can read inputs using sensors, buttons and producing output in various ways such as moving actuators and turning on LED lights. The board is programmed using Arduino software Integrated Development Environment (IDE) to perform the processing on the inputs and produce outputs. Arduino is widely used because of its portability and low cost. Moreover, Arduino's programming environment is an open source environment that is simple and could extend C++ libraries (Arduino, 2020) .

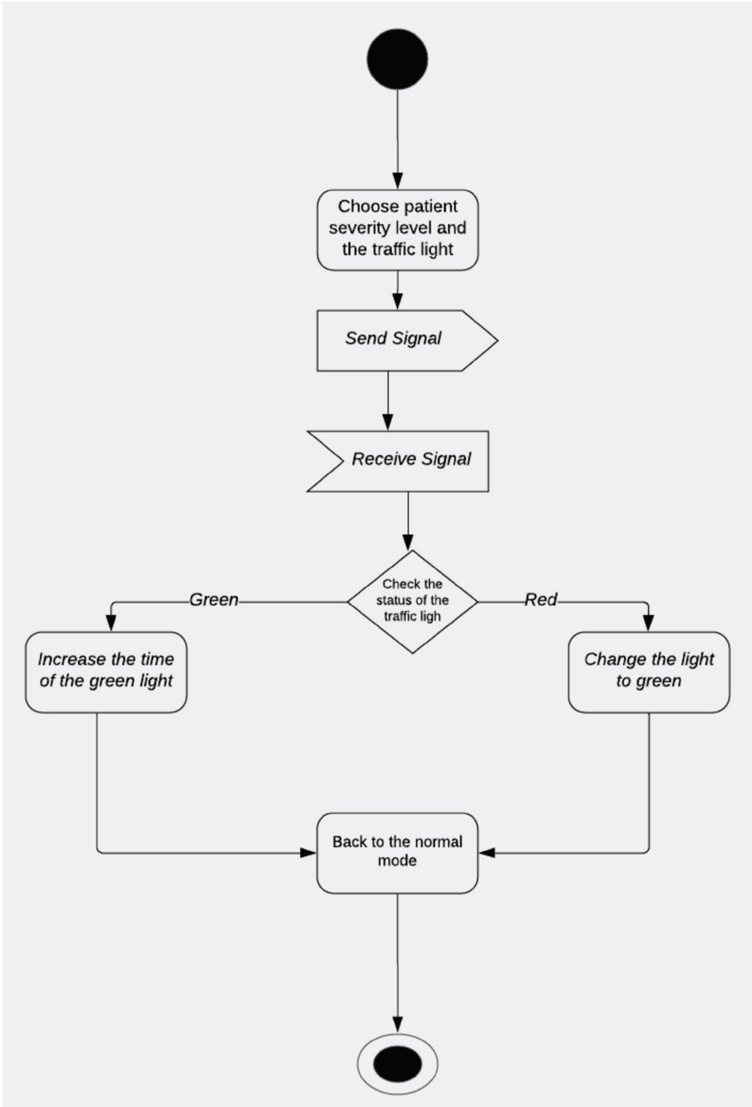


Fig. 1: Activity diagram of the proposed system.

Arduino launched different versions of Arduino boards that suits for different applications. The most popular Arduino boards are Arduino Uno, Arduino Nano, Arduino Due, and Arduino Mega. Arduino Uno is one of the best Arduino boards because of its diverse capabilities that can be used by beginners or even for advanced projects. Arduino Nano is used in projects where small size of the board is important. Arduino due is preferred for large scale projects. Arduino mega is used where large memory space is required. However, most Arduino boards have common components. The common component of the Arduino boards consists of power, pins, rest button, power LED indicator, TX RX LED's, main IC and voltage regulator (Arduino, 2020) . The Raspberry Pi is a low cost, credit-card sized computer that plugs into a computer monitor or TV and uses a standard keyboard and mouse (Raspberrypi, 2020) . Using Raspberry Pi enables to explore computing and learns how to program in languages such as Scratch and Python. It can be used for almost everything starting from browsing the internet to make spreadsheets. Raspberry Pi can communicate with the outside world. It is also used in lots of projects from music machines to weather stations (Raspberrypi, 2020) . Since the Raspberry Pi is a credit-card sized computer, it can be described like other computers using terms such as operating system, processor, power, memory. Raspberry Pi uses Raspbian operating system, which is based on Linux, but it also provides a few options of another operating systems. The processor is BCM2835, that is based on ARM (Advanced RISC Machines). Regarding the power, it can be powered using solar cell or battery. There are four power modes in Raspberry Pi: run mode, standby mode, shutdown mode and dormant mode. In Raspberry Pi the Storage process occurs in Secure Digital (SD) Card. Raspberry Pi does not have a hard drive, but it can connect with external hard drive using USB ports (M. Maksimovi_c et al., 2014) .

4. System Design

In this section we illustrate the system using circuit diagram and present the proposed system in block diagram. A circuit diagram is a visual display of an electrical circuit using either basic images of parts or industry standard symbols. These two different types of circuit diagrams are called pictorial (using basic images) or schematic style (using industry standard symbols). We use the pictorial circuit diagram to represent the proposed system. The system can be represented using two circuit diagrams, one for the traffic light unit and the other for the ambulance unit.

- Traffic light unit

This circuit diagram shown in Fig. 2 represents the traffic light unit, which shows the connection between the Arduino Uno microcontroller, ZigBee receiver, and traffic lights modules. This unit is the receiving signal unit which will receive the ZigBee signal from the ambulance unit and will process the signal in the microcontroller to control the traffic lights.

- Ambulance unit

This circuit diagram shown in Fig. 3 represents the ambulance unit, which shows the connection between the Arduino Uno microcontroller, ZigBee transmitter, LCD display, and the keypad. This unit is the transmitting signal unit, which will transmit the ZigBee signal to the traffic light unit. This unit takes inputs (patient severity level and traffic light to control) from the user using the keypad and sends these inputs to the traffic light unit using ZigBee.

The block diagram is used to represent the components of the system and shows the flow of the system's work. It represents each component by block and uses arrows to show the relationship between the components and the system's work. Fig. 4 shows the block diagram of ambulance unit and the block diagram of traffic light unit respectively.

We have designed the interface to be user friendly and consistent. The interface operations are done in the same way and the interface have consistent colors, font size, and terms. The interface shows a feedback to provide information to the user about what action has been taken and what has been accomplished. Fig. 5 shows the system architecture. Fig. 6 shows the overall system workflow.

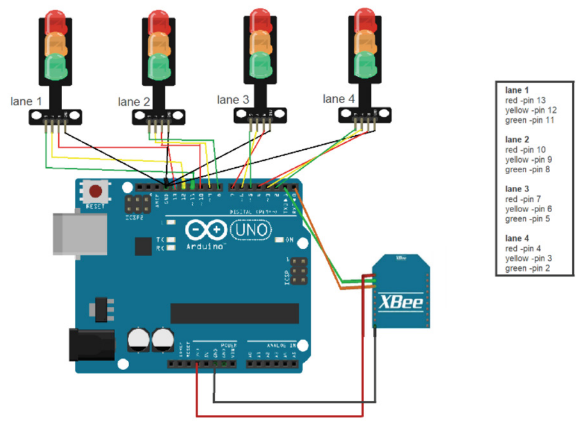


Fig. 2: Traffic light unit circuit diagram.

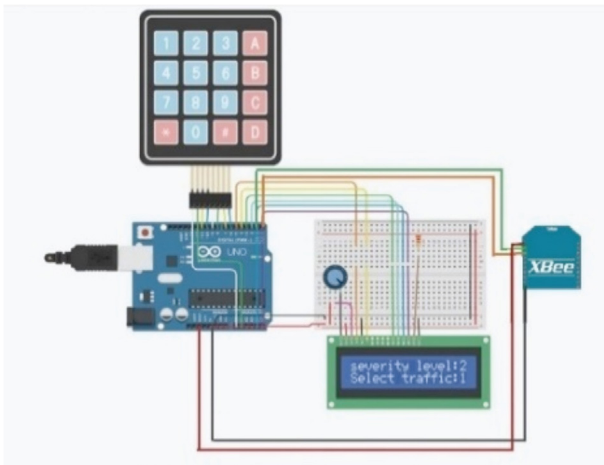
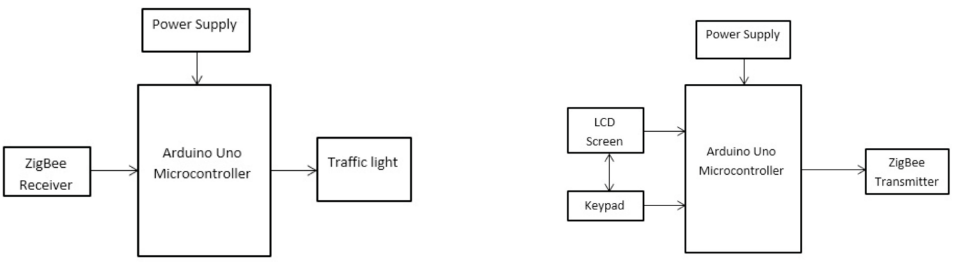


Fig. 3: Ambulance unit circuit diagram.



a. Block diagram of traffic light unit. *b. Block diagram of ambulance unit*

Fig. 4: Block diagram of the units involved in the system.

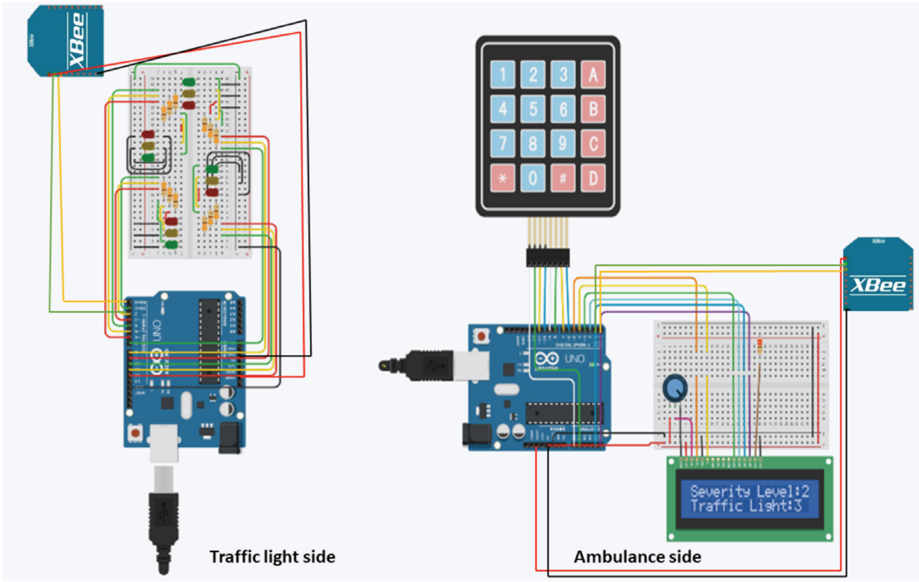


Fig. 5: System architecture

Here is an explanation of the system workflow:

Ambulance driver side

1. Start Xbee sender
2. The driver will choose the patient severity level and the microcontroller will check
    i. If the number is larger than 5, it will ask the driver to enter it again
    ii. If the number is less than 6, it will save the value
3. The driver will choose the traffic light number and the microcontroller will check
    i. If the number is larger than 4, it will ask the driver to enter it again
    ii. If the number is less than 5, it will save the value
4. Send the two saved values to the zigbee coordinator (Xbee).

Traffic light controller side

1. Start Xbee receiver
2. Check if Xbee received a signal
3. If Xbee didn't receive a signal, the traffic light will be at the normal mode
4. If Xbee did receive a signal, check the number of signals
5. If there is more than one signal, chose the signal with the highest severity level
6. If it is one signal, turn on green light for the selected traffic light
7. After that see if there is another signal or not. if there isn't a signal stay in the normal mode and if there is a signal repeat from step5.

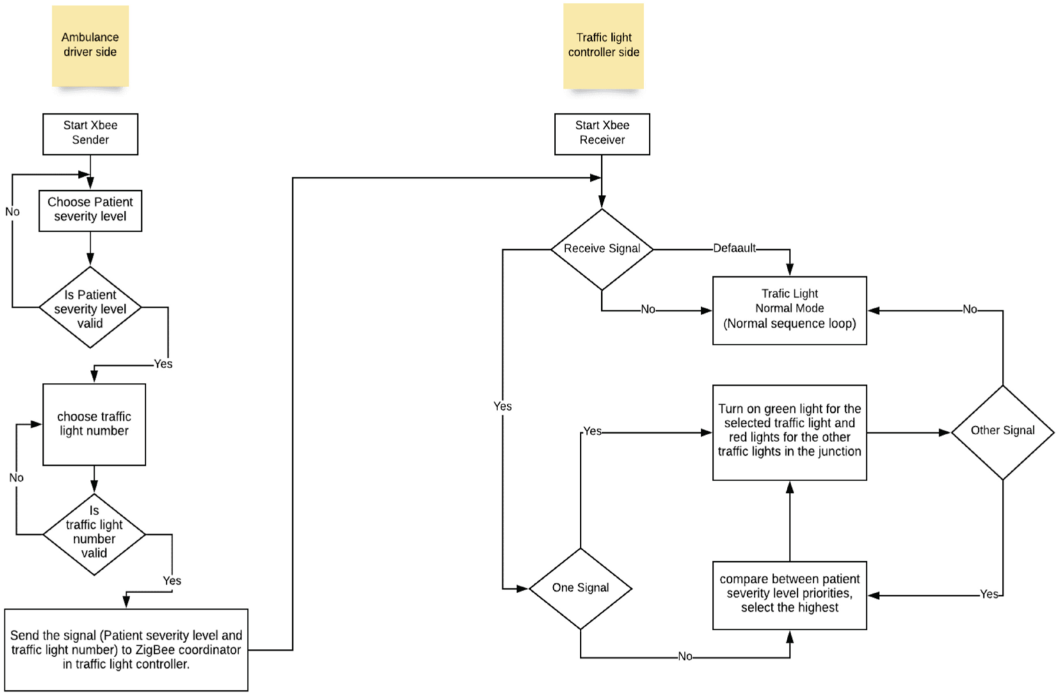


Fig. 6: System workflow.

5. System Implementation, Testing, and Validation

In this section we will describe the tools and programming languages used to build the system, and the configuration and the programming language used for the hardware (Arduino and ZigBee). The hardware used to implement this project includes Arduino UNO and mega with network shield (ZigBee). We have used C++ language and Arduino IDE to program the Arduino and the ZigBee. For ZigBee hardware configuration, XCTU software is used. The implementation process which includes wiring and configuring the hardware part and coding using Arduino IDE.

Traffic light controller unit implementation stages:

- Connecting the traffic light module to the breadboard.
- The traffic lights are implemented by setting a specific time for each LED light starting from red to green.

4 Traffic light controller units implementation stages:

- Connecting four traffic light modules to the breadboard.

- To simulate four lane traffic lights, setting time for each traffic light needs to be synchronized with other traffic lights in the junction. So, only one traffic light is set to green and the other traffic lights in the junction are red. Fig. 7 shows the traffic lights unit

LCD screen unit implementation stages:

- Wiring the LCD screen needs welding the wires embedded in the screen, that was difficult to discover and to implement. Then, connecting the screen to the breadboard.
- Coding the LCD
- LCD brightness issue: after the LCD screen was connected to the Arduino the screen brightness was not working but it displays outputs, this problem solved by serial i2c LCD display adapter.

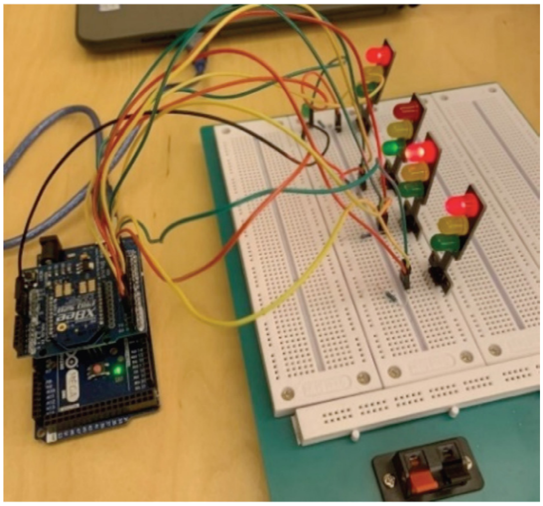


Fig. 7: Traffic Light Unit.

Keypad unit implementation stages:

- Wiring the keypad to the Arduino. This is a bit confusing, because every wire was connected to a specific set of characters. So, when wiring a wrong wire, it results in displaying a wrong output.

Configuring the ZigBee

The proposed system needs two ZigBee nodes in the ZigBee network for communication: the transmitter and the receiver. As a result, each node will be configured differently. ZigBee has two modes of configuration: AT mode and API mode. In this system, AT mode is chosen to configure transmitter and receiver nodes. In this system, the transmitter is the ambulance driver side, where the receiver is the traffic light controller side. The configuration of ZigBee nodes is made using XCTU software. Transmitter is configured as AT router and the receiver is configured as AT coordinator, see Fig. 8 and Fig. 9. For both the transmitter and the receiver nodes the network ID and baud rate are the same so they can communicate with each other. The coordinator (which is the traffic light controller unit) is the node that is on all the time will receive any signal at any time. Moreover, the coordinator is checking periodically for any signal available. The router (which is the ambulance driver unit) is only transmitting signals when data is available.

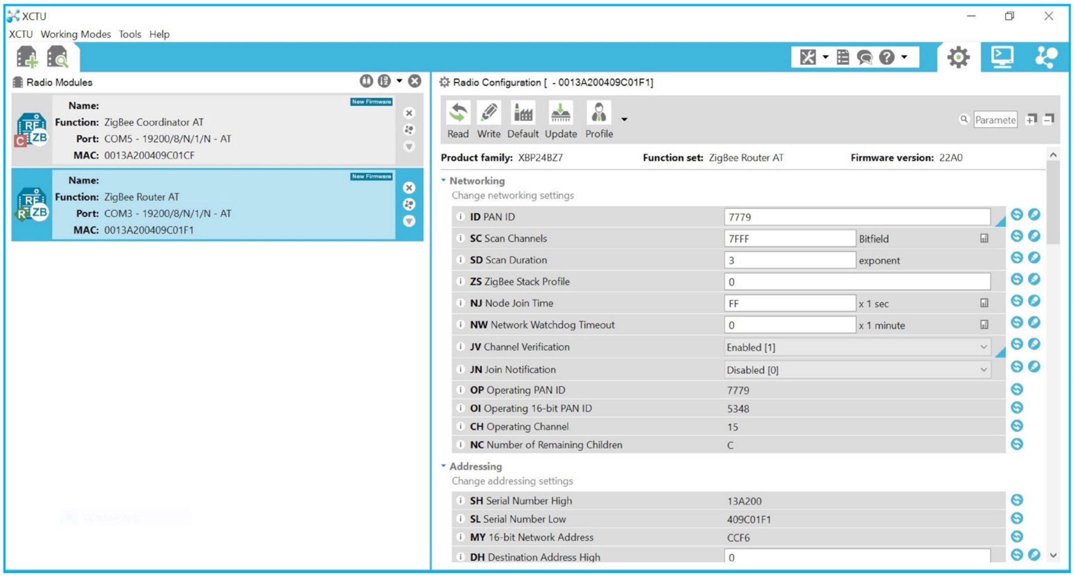


Fig. 8: ZigBee coordinator configuration.

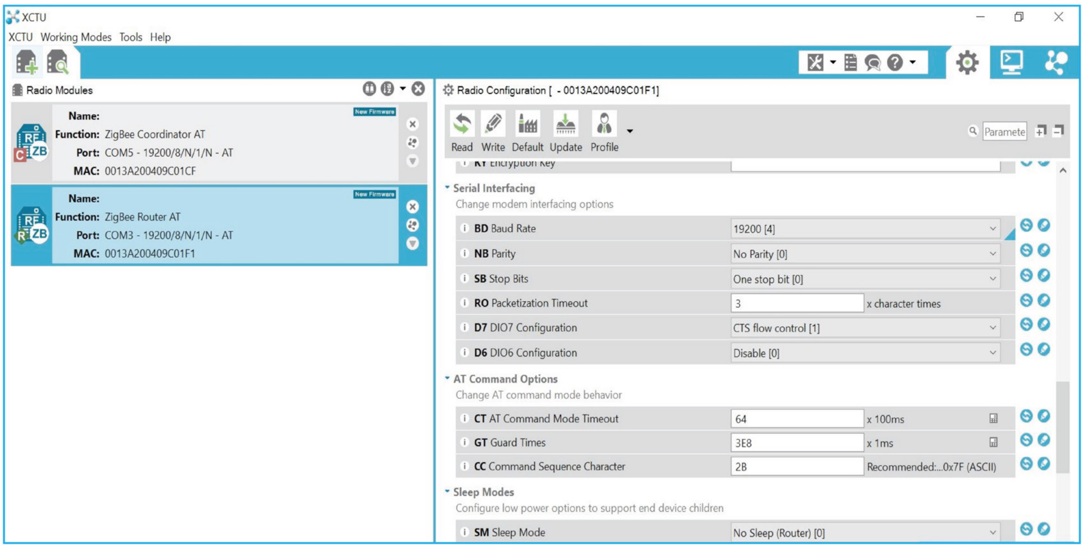


Fig. 9: ZigBee router configuration.

Coding the ZigBee

In ZigBee communication, we faced many issues from the configuration to dealing with data sent and received via ZigBee signals. We used Serial Software library to handle processing data (sent and received data). The format of sent and received data was ambiguous. It always sent characters and received different characters or receive data in unreadable form. This is due to the fact that the ZigBee is dealing with data as bytes so casting to character data type is required. Another issue is raised, when the transmitter sends data the coordinator could not receive these data. This issue is resolved by changing the baud rate in the serial monitor in Arduino IDE to the baud rate specified in the C++ code and data is finally received correctly. The system has been successfully designed and tested. We have tested the system in different way. First, we have tested several units of the system separately, then we performed integration testing to test the associated units together to check their performance. Then, we have tested the whole system with all possible scenarios to check the interaction between all the system's units. Finally, usability testing is applied to test the system from the user side. The system interface implementation is shown in Fig. 10.

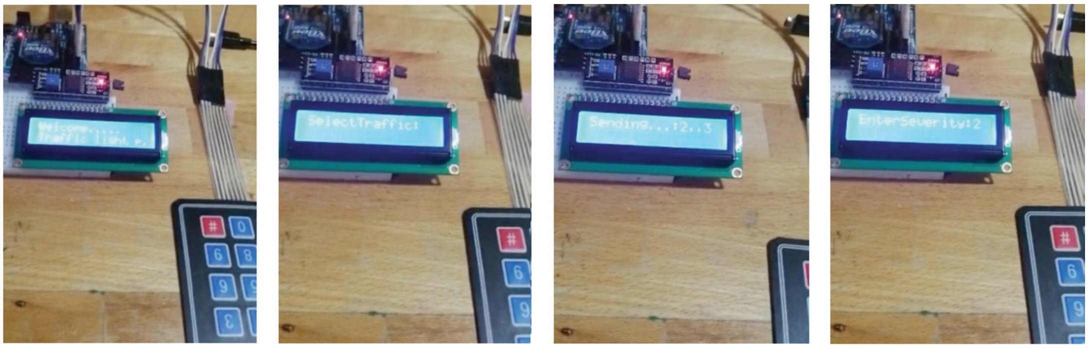


Fig. 10: System interface.

Table 2 compares the estimated time for an emergency vehicle to travel from point A (the hospital) to point B (the patient location) with and without using the proposed system. The first column shows the estimated time depending on the GPS, the times are without any traffic delay because we took the time of the distances after midnight which means in a quiet part of the day. The second column indicates the number of the traffic lights that the emergency vehicle will cross, the third column represents the total duration time for each red light between A and B (average duration time for the red light =1), and the fourth column is the time the emergency vehicle will take to travel using the proposed system.

The proposed system would take 00:12.21 seconds to accomplish its task, which starts when the driver selects the severity level and the target traffic light and ends when the traffic light turns to green. If we exclude the time needed to select the severity level and the target traffic light values by the driver, the system would take 00:2.32 seconds. Actually, the delay which is caused by the user input would not affect the performance of the proposed system since the driver would select these values while she/he is driving near to the traffic light. Regarding to the distance factor, we measure the complete time that the proposed system takes from selecting the values until turning the green light on using different distances between the emergency vehicle and the traffic light, see Table 3.

Table 2. The estimated time for an emergency vehicle to travel from point A to point B.

The time to travel from A to B (using the GPS)	No. of traffic lights on the way	The total estimated duration time for each RED light between A and B	Actual Time to travel **System not applied**	Actual Time to travel **System applied**
10 min	2	2 min	10 min	8 min
6 min	1	1 min	6 min	5 min
4 min	0	0 min	4 min	4 min

Table 3. The estimated time for the proposed system considering different distances between the emergency vehicle and the traffic light.

Distance (m)	Close to each other	4 m	8 m	12 m	16 m
Time (s)	00:12.21 s	00:12.58 s	00:13:03 s	00:13.30 s	00:20.01 s

6. Conclusion and Future Work

In big cities like Jeddah, due to the traffic congestion, emergency vehicles such as ambulance, fire engines are affected by traffic jams and consequently many people could lose their lives because of an ambulance delay. This project proposed a solution for this problem by proposing an emergency mode to the current traffic light system, which gives ambulance the priority to pass the traffic light to arrive to patients and hospitals smoothly.

The system consists of two parts, part will be placed in the traffic light controller and the other part will be placed in the ambulance. The traffic light part consists of Arduino microcontroller and network shield (ZigBee). While the ambulance part consists of Arduino microcontroller, network shield (ZigBee), Keypad and LCD screen. The driver interacts with the system by two clicks, the first one is to select the severity level, the second one is to select the target traffic light. The severity number is used in case that there is more than one ambulance arrived in the same junction and requested to turn the traffic light to green, the system while handle the request with the highest severity level. The screen displays two questions that asking the driver to select the severity level and the traffic light using the keypad, after a valid data input the ZigBee will send these data to the other ZigBee in the traffic light part. Then the Arduino microcontroller in the traffic light part will turn the target traffic light or the traffic light with highest severity level (if there is more than one ambulance) to green. The proposed system will facilitate the ambulance movement, unlike the current system where the ambulance gets stuck in the traffic. Also, the ambulance will not be forced to pass the red light, which may cause an accident. Eventually, the system will be a very helpful to save lives. Many desire adaptations have been left for the future due to lack of time. Future work aims to support Arabic language, encrypt the sent signal to enhance the security of the system, and create a website to enable the administrator in Traffic Police Station monitoring the changes of the traffic lights in emergency mode.

7. References

"Arduino documentation.» , 2020.[Online]. Available: https://www.arduino.cc/

"Raspberry Pi documentation.» [Online]. Available: https://www.raspberrypi.org/

Azalghamdi01, "Population in saudi arabia by gender, age, nationality (saudi / non-saudi) - mid 2016 a.d,» Apr 2017. [Online]. Avalable: https://www.stats.gov.sa/en/5305

"Enterprise ready mqtt to move your iot data.» [Online]. Available: https://www.hivemq.com/

J. Osborne, "Internet of things and cloud computing," Internet of Things and Data Analytics Handbook, pp. 683-698, 2017.

M. Maksimovi_c, V. Vujovi_c, N. Davidovi_c, V. Milo_sevi_c, and B. Peri_si_c, "Raspberry Pi as internet of things hardware: performances and constraints,» design issues, vol. 3, no. 8, 2014.

G. Beri, P. Ganjare, A. Gate, A. Channawar, and V. Gaikwad, "Intelligent ambulance with traffic control,» International Journal of Electrical, Electronics and Computer Systems, Feb 2016.

D. S. Reddy and V. Khare, "A smart ambulance system,» International Journal of Innovative Technologies (IJITECH), vol. 5, no. 02, pp. 0224-0227, 2017.

R. Sundar, S. Hebbar, and V. Golla, "Implementing intelligent traffic control system for congestion control, ambulance clearance, and stolen vehicle detection,» IEEE Sensors Journal, vol. 15, no. 2, p. 1109-1113, 2015.

H. Singh, K. Kumar, and H. Kaur, "Intelligent traffic lights based on rfid,» International Journal of Computing and Business Research, vol. 2012, pp. 110, 2012.