AUTOMOTIVE COLLISION AVOIDANCE SYSTEM (ACAS) APPLICATION

Mohd Aminudin Jamlos1*, Jegan Moorali1, Wan Azani Wan Mustafa2, Syed Zulkarnain Syed Idrus3

1Faculty of Electronic Engineering Technology, Universiti Malaysia Perlis, UniCITI Alam Campus, Sungai Chuchuh, 02100 Padang Besar, Perlis, Malaysia
2Faculty of Electrical Engineering Technology, Universiti Malaysia Perlis, UniCITI Alam Campus, Sungai Chuchuh, 02100 Padang Besar, Perlis, Malaysia
3Faculty of Applied and Human Sciences, Universiti Malaysia Perlis, 01000 Kangar, Perlis, Malaysia

1mohdaminudin@unimap.edu.my

Abstract. Automotive collision avoidance system is a system that avoid collision when danger occur. It is also very helpful in order to alert the drivers and passengers. The system also reacts to danger automatically, faster, and more efficient. The system is implemented on a remote control car. The remote control car can automatically brake when the difference between front ultrasonic sensor and obstacle is less than 35cm. While the system applies automatic braking system a red LED will light on. Furthermore, the system is also surrounded with another 3 ultrasons which is on both sides and at the back of the prototype. In this project, this system is using 2 controllers which is Raspberry Pi 4B and Arduino Uno. A camera module is connected to Raspberry Pi which detects the corresponding object. Next, the ultrasonic sensor is used to detect distance and send the reading to Arduino Uno to calculate distance. LCD is used to display the calculated distance. This system also uses dc motor to control the movement of the remote control car through the Arduino Uno controller LED’s and a buzzer is used as alerting devices for the prototype.

1. Introduction

In the present situation, traffic collisions lead to injury and casualties involving people. About half of all road traffic accidents typically include people between 15 till 45 years of age. The incidence of road traffic accidents has been predicted to constitute around 1.5 per cent of gross national product (GNP) in intermediate-income countries and 2 per cent in great-income countries. The causes for road accidents are numerous and it includes rapid urbanization, unsteady safety practices, lack of compliance, intoxicated or sleepy driving, impact of hallucinogenic substances and alcohol, overspeed and fail to wear seat belts.

Malaysia is the next highest accident rate for road injuries in Asia and Asia, after Thailand and Vietnam. Such fatality figures are close to those in certain African nations. Transport deaths were the most numerous causes of death in Malaysia in 2019, with 27,613,120 registered vehicles in the same year, including 13,123,638 four-wheeled light vehicles, 12,677,041 two-and three-wheeled vehicles, 1,191,310 heavy-duty vehicles, 59,977 buses and 561,154 other vehicles. Due to increase in accident rate has led to the increase of need for collision avoidance techniques. The collision avoidance is a pre-crash system to increase the safety by giving alerts to the drivers using the collision avoidance techniques available. The collision avoidance focuses to reduce the collisions and fatalities of pedestrians and vehicle drivers.

In Malaysia, the incident rate tends to escalate due to lack of care or other factors, either the situation is not at all understood or there is a pause in correctly perceiving the condition.(Basma & Refai, 2009). Besides that, the estimation and understanding of the perceived condition was not acceptable to conditions. Moreover, the strategy for dealing with an entity instances and assessed situation is too slow.(Wang & Zhang, 2013). In addition, A totally insufficient action has been taken. In addition to this,
injuries are growing due to the registration of vehicles, the more speed a vehicle gets, the complex it becomes to slow down and the lack of enforcement. (Khairul et al., 2018).

The aim of this project is to achieve a collision avoidance system which provide early warning to the drivers in order to avoid collision with the pedestrians and other vehicles which able to detect vehicles using camera module and to detect distance between an obstacle surrounded by the prototype using ultrasonic sensor. It also to alert the driver on the obstacles in real time and to automatically brake the vehicle to avoid crash. Lastly, develop a prototype of collision avoidance vehicle system that functions efficiently.

This project is mainly to develop a collision avoidance system. The system introduced is basically to reduce accidents. This project used remote control car to develop a prototype using high precision ultrasonic range finder module, camera module, LED’s, DC-motor, piezo buzzer, single-channel relay, I2C LCD and 2 microcontrollers which are Arduino Uno and Raspberry pi 4B.

This project is an automatic self-controlled system which uses microcontrollers as the brain. The automatically braking system here is basically to cut off the motor rotation. This ensure when any obstacle is detected in front of the car, the motor stop rotates within a specific range of distance. The system also configured to turn on the LED’s and buzzer as to alert about the danger heading towards the remote-control car. The project has a microcontroller that act as control unit for the system. This project also be programmed to detect the cars in front the prototype suing a camera module which is controlled by another microcontroller as a processing unit.

This project uses two type of programming languages which is C programming and Python. Both programs were controlled by the project working principles and the programming languages would be studied in detail before designing the circuit. Besides learning to programme, it is also important to understand how to integrate hardware. The datasheet is important before use device. Important data that needs to be emphasized voltage and pin configure.

2. Methodology

![Figure 1 Block Diagram of Raspberry Pi interface with camera module](image_url)
**Figure 2** Block Diagram of the Arduino Uno with the components

**Figure 3** Flow chart of the system (Raspberry Pi)
Figure 4 Flow chart of the system (Arduino Uno)
2.1 Hardware and software

2.1.1 Camera Module

The Pi camera module is a light-weight camera that supports to Raspberry Pi. It communicates with Pi through the MIPI camera serial interface protocol. It is normally used for image processing, machine learning or in surveillance projects. It is commonly used in surveillance drones since the payload of camera is very less. Apart from this, Pi camera is also used as webcam and live streaming.

![Figure 5 8MP Camera Module](image)

2.1.2 RaspberryPi 4B

A microcontroller Raspberry pi 4 will do a remarkable amount of work. Amateur computer enthusiasts use Pi boards as a media hub, file servers, classic game consoles, routers, and network-level ad-blockers for example. Yet that's just a glimpse of what's possible. There are loads of projects people used Pi to create tablets, computers, phones, robots, smart mirrors, etc.

![Figure 6 Raspberry Pi 4B](image)
2.1.3 Remote-control car

A remote-control car is used as prototype for this project which has remote to control the car. The car moves by the dc motors. The remote-control cars can be controlled using a specialized remote from a distance.

![Remote-control car](image)

**Figure 7** Remote-control car

2.1.4 Ultrasonic Sensor

This HC-SR04 ultrasonic sensor is perfect sensor for distance measurement and object detection. It offers great accuracy range and provide stable readings. This sensor operates at 5V dc and has 4 pins. The pins are VCC, Trig, Echo and Ground. The ultrasonic sensor is used to detect the distance between the remote-control car and the obstacle.

![Ultrasonic Sensor](image)
2.1.5 Arduino Uno

The Arduino Uno is an ATmega328 based microcontroller module. The microcontroller has 14 digital input/output pins. 6 of the pins can used for PWM outputs, 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. The microcontroller can simply connect to the computer with a USB cable or power up using a Ac-to-DC adapter or a power bank to get started.

2.1.6 Single-Channel Relay

Relay is a component that control the voltage of another component. This single-channel relay supplies 3.75V-6V. A 5V single-channel relay breakout is applicable to use in microcontroller such as Arduino, SK40C, SK28A and SK18B. The amount of current when the relay is active is approximately 70mA. Moreover, the relay uses maximum current of 10A. It is a single relay board and control digital output.
2.1.7 Dc motor

A dc has an operating voltage of 4.5V to 9V, but a recommended voltage of 6V. its power is 5W. The motor rotating speed is about 12000rpm. The motor is able rotates at clockwise and anti-clockwise.

![Dc motor](image)

**Figure 10** Single-Channel Relay

2.1.8 I2C LCD

The I2C module, easily controlled the LCD using 2 wires connected to the Arduino microcontroller via SDA and SCL. The module attached has four pins on the left side of the board, two are voltage and ground, and the other two were I2C (SDA and SCL). This LCD had a total of 16 pins. The I2C module

![Dc motor](image)

**Figure 11** Dc motor
is attached to the LCD then was connected to the Arduino. Wiring an I2C LCD was much simpler than connecting a normal LCD. All users needed to do was attaching 4 pins instead of 16 pins.

![Figure 12 I2C LCD](image12.png)

### 2.1.9 Piezo Buzzer

A 5V piezo buzzer is used for this project. This buzzer operates 3-5V direct current. It requires pulses with resonant frequency of approximately 2300Hz to produce continuous beep sound. It requires pulses with certain frequency to produce tones. It has two connecting wire red and black. The red wire is connected to power supply and the black wire is connect to the ground. This buzzer operates in the range of 4-8V direct current.

![Figure 13 Buzzer](image13.png)

### 2.1.10 LED
In the simplest terms, light emitting diode (LED) is a semiconductor device that provides light as the electrical current passes through it. Light generated when the particles carrying the current are known as photons and the gaps are integrated into the silicon. As light is emitted within a solid semiconductor material, LEDs are known as solid state devices. The term solid-state lighting, often involving organic LEDs (OLEDs), distinguishes this lighting device from other systems using either heated filaments (incandescent and tungsten halogen lamps) or gas discharge (fluorescent lamps).

![Figure 14 LED](image)

2.1.11 Raspberry Pi OS

Raspbian is a recommended program for use on Raspberry Pi as a standard procedure. Raspbian is a free operating system optimized for Raspberry Pi hardware. Raspbian covers about 35,000 packages and pre-compiled applications bind in a friendly pattern for simple deployment on Raspberry Pi. It is a community project into active development, with a focus on improving the reliability and efficiency.

![Figure 15 Raspberry Pi operating system](image)
2.1.12 Arduino IDE

Arduino Integrated Development Environment (IDE) is a cross-platform application (for Windows, MacOS, Linux) written in the Java programming language. It is used to write and upload programs to Arduino compatible boards, but also, with the help of 3rd party cores, to other vendor development boards. Embedded system means some combination of hardware and programmable software.

![Arduino IDE](image)

**Figure 16 Arduino IDE**

2.1.13 Py-Charm IDE

Py-Charm Integrated Development Environment is a software which runs python language. Python is a high-level programming language. Python languages are easier to understand than C/C++ languages. Acquiring expertise of Python would be the secret to your future as a Data Scientist. A person skill is developed for many data science roles by learning Python. In current scenario, Python is most wanted language in industrial environment. This is because, the revolution of industry changes to automation. Most of the machines are now semi-automated and fully automated to reduce manpower.
Virtual Network Computing (VNC) is a remote desktop app that helps to remote from anywhere using IP address. This helps to control a desktop activity easily from another device relaying the graphical-screen operation. The advantage of this system is numerous clients are permitted to VNC server simultaneously and has cross-platform support. There are several suitable purposes of this technology such are intuitive remote control, file transferring, printing, chatting, and accessing file from one computer to another or vice versa. Moreover, this platform is secured and has online support from management.
2.1.15 EasyEDA

EasyEDA is an optimized browser-oriented schematic capture, SPICE circuit simulation, based on Ngspice, and PCB layout. It facilitates the design and editing of schematic diagrams, the editing of printed circuit board drawings. The ability to import LTspice schematics and symbols is a convenient way to port schematics to PCB layout without having to rewrite them from scratch.

![EasyEDA software](image)

**Figure 19** EasyEDA software

2.1.16 Fritzing

Fritzing is a hardware open-source project, which makes electronics available to everyone as an artistic material. We deliver a coding tool, a community website, and resources in the tradition of Processing and Arduino, cultivating an innovative environment that encourages users to record their designs, share them with others, teach electronics in the classroom, and design and produce professional PCBs.

![Fritzing software](image)

**Figure 20** Fritzing software
3. Results and Discussion

3.1 Cascade Trainer

In this part, the results and analysis of this project are shown and discussed the method to obtain the results with reference to the objective of this project. Initially, a folder which contains positive images and negatives are added into the Cascade Trainer GUI application as shown in Figure 16. Next, the number of negative image count is set. Then the number is stages is set 15 and the width is set to 36. The rest all remain default where it will create a haarcascade.xml by default.

![Cascade Trainer GUI input](image)

**Figure 21** Cascade Trainer GUI input

In the Figure 17 and Figure 18, shows the log of opencv trainer cascade using positive and negative images. Once done, a .xml file will automatically be created in the trained folder.
3.2 Camera module interface with Raspberry Pi

Moreover, a folder named “opencv” is created which contains libraries and python file. Next, a python file named ‘cascade_detect.py’ is created and saved in the folder. Later, the haarcascade.xml file which is created will be added in this folder.
In Figure 20, the library for opencv is declared by adding the path of the cascade file to read. Nevertheless, the code programmed is to open a window which displays the live cam and also to detect according to the sample in the haarcascade.xml which is added into the folder. Moreover, the cascade method detects an object using the grayscale of the real image or video.

Once the code is run, a window is opened showing the live cam. There will be a rectangle drawn which determined it detects a car referring to the sample in haarcascade.xml file as shown in Figure 21. The appearance of the rectangle with the colour of it is configure in the coding. the rectangle is shortly called as bounding box. The rectangle size differs each time it detects depends on the area it detects.
3.3 Ultrasonic sensor and output components interface with Arduino Uno

In Figure 22 till Figure 27, shows the full coding used and the output of distance in the serial monitor of Arduino IDE.

![Figure 26 Results of camera detect cars](image)

![Figure 27 Coding of pin define with serial monitor of Arduino IDE](image)
Figure 28 Coding of pin setup with serial monitor of Arduino IDE

Figure 29 Coding of setting LCD with serial monitor of Arduino IDE
Figure 30 Coding of front sensor with serial monitor of Arduino IDE

Figure 31 Coding of back sensor and right sensor with serial monitor of Arduino IDE
In Figure 28, it shown the working of front ultrasonic sensor and the red LED. The LCD displays the distance between the obstacle and the sensor. The LCD only displays the distance of the front sensor front the obstacle. The LED is on when the distance between the obstacle and front sensor is less than 35cm. At this moment, the buzzer is activated to give beep sound repeatedly for 5 times. During this phase, the motor stop rotates which defined as automatic braking system.

**Figure 32** Coding of left sensor and distance subroutine with serial monitor of Arduino IDE

In Figure 29, the yellow LED will on once the sensor of the remote-control car at the right side detects obstacles less than 10cm.

**Figure 33** Result of front sensor and distance display in LCD
In Figure 30, the blue LED will on once the sensor of the remote-control car at the left side detects obstacles less than 10cm.

In Figure 31, the green LED will on once the sensor of the remote-control car at the back detects obstacles less than 10cm.
As in detail, the Raspbian OS is downloaded and flashed in a 32GB micro sd. A text file is created as ssh.txt, this is to ensure that the microcontroller is connected to a wireless network automatically. Next, the IP address of the Raspberry Pi is obtained from the IP Scanner application. This can be defined as IoT implementation. Once the IP address is obtained, it is then copy and pasted in the VNC Viewer application. The problem of this is the IP address of the Raspberry Pi changes most of the time. So, the IP Scanner application is running to scan the IP address of the Raspberry Pi each start up. The VNC Viewer is an application that act as remote desktop. It able to control the activity of the Raspberry Pi. In addition, the configuration box is opened to enable camera module to integrate with raspberry pi. Once enabled, the package for the Raspberry Pi camera is installed using the terminal. The source to install the packages are referred from the internet. The camera is tested by capturing few images using the source code from GitHub website.

Secondly, Py-charm IDE is installed in the raspberry pi to do programme for the camera module to detect the vehicles. The programming language used in this is Python. The need of this software to installed in Raspberry Pi is because, there are difficulties in finding packages to installed directly by using the terminal in the Raspbian operating system. In order to detect objects, Opencv is one of the ways to make it possible. It will make the camera to process the image that captures by the camera and gives the outcome according to the programme coded. The library of Opencv is installed firstly into the Py-charm IDE. Next, a folder named ‘Opencv’ is created. After that, a python file is created and saved as ‘cascade-detect’.

After the programme coded is run, a window will open and shows the live streaming. At the same moment, there will be rectangles formed shows that it detects an object. The detected object is from the haarcascade.xml file of the cascade method. The haarcascade.xml file is saved in the same folder as the python file. This is to ensure the path of the file used to detect an object. The cascade method is created using positive and negative images. That means the object that is detected as from the positive images. This precision of detecting an image depend on the numbers of positive and negative images. Negative images should be more than positive images for better results.

Furthermore, the more challenging part was while interfacing the sensors with Arduino Uno. The ultrasonic sensors are initially tested with pre codes from the internet source. Once the sensors work
properly, then it is connected accordingly to the pin configured in the programme. The ultrasonic sensors are glued all four sides of the remote-control car. Moreover, the motor of the remote-control car is also tested and worked perfectly. The motor runs at voltage more than 5V, if the voltage is less than 5V then the motor stops rotating. Thus, the microcontroller provides 5V to the motor and there is no issue.

A relay is used to make the motor act as automatic braking system. One pole of the motor is connected to the common of the relay. This initially stated that the motor is set to normally close where it is connected to ground when the in1 is high. When in1 is signalled to low, the relay will switch to normally open where it is connected to 6V. In this phase, the motor will act as automatic brake when 5V of the motor contradicting with 6V of the relay. However, the automatic braking system applied in this project is applicable to use for short time due to the speed of the motor rotates.

Moreover, the LCD and I2C module bought was not attached together. In that case, the LCD and I2C module is soldered to ensure only 4 pins is used to interface rather than using 16 pins of the LCD to save the output pins of the microcontroller for other components. The soldering part was confronted because it should not overlay with another pin beside of the LCD. Finally, the I2C LCD is tested with some pre-code from the inter and it worked accordingly. Besides that, there is problem to use the source code from internet because some libraries that added does not get to read the programme. This problem was then encountered by going through several tutorials to get the appropriate code for the library used.

Lastly, the LED’s and buzzer are interface with microcontroller. The LED’s and buzzer are tested separately one by one before soldered to the circuit board. The LED’s are connected in active HIGH configuration. A 330Ω resistor is soldered for each LED. While soldering, the red LED and yellow LED was soldered oppositely. This affected when the distance between obstacle and front sensor distance less than 35cm, red LED should be on but yellow LED was on. Then, it is corrected by changing the pin configuration in the programme from A0 to A1 of the red LED and A1 to A0 yellow LED.

Finally, the project is working perfectly according to the programme coded in Raspberry Pi and Arduino Uno with its interfaced outputs.

4. Conclusion

This project develops automotive collision avoidance system (ACAS) application to apply automatic brake to avoid collision between any obstacle in front of the vehicle. This project used a camera module and Raspberry Pi to detect cars. Moreover, this project also used ultrasonic sensor as to measure distance between an obstacle and the prototype. The LED’s and buzzer are used as visual and sound warning alert to the driver and passenger in the vehicle. After running the test, the car detects is displayed on the screen and distance calculated of the front sensor will be shown in LCD. In a real situation, the camera shows the detected cars lively on a screen. In addition, the distance between the obstacle and the vehicle will display on a screen. Based on the results, this project can be said to be successful and objective was achieved.

References

[1] Aliyu, A., Kolo, J. G., Mikail, O. O., Agajo, J., Umar, B., & Aguagba, O. I. (2017). *An Ultrasonic Sensor Distance Induced Automatic Braking Automobile Collision Avoidance System*. 570–576.
[2] Amazon.com: JEYPOD Remote Control Car, 2.4 Ghz High Speed Racing RC Car with 4 Batteries, Kids Toys, Red: Toys & Games. (n.d.). Retrieved June 5, 2020, from https://www.amazon.com/JEYPOD-Remote-Control-Racing-Batteries/dp/B07V6SXHBN

[3] Arduino – Logos Download. (n.d.). Retrieved January 27, 2021, from https://logos-download.com/9905-arduino-logo-download.html

[4] Arduino Uno Rev3 | Arduino Official Store. (n.d.). Retrieved January 27, 2021, from https://store.arduino.cc/usa/arduino-uno-rev3

[5] Basma, F., & Refai, H. H. (2009). Collision Avoidance System at Intersections FINAL REPORT - FHWA-OK-09-06. 2216.

[6] Best OS for Raspberry Pi - IoTbyHVM - Bits & Bytes of IoT. (n.d.). Retrieved June 5, 2020, from https://iotbyhvm.ooo/os-for-raspberry-pi/

[7] Deotale, M. S., Shivankar, H., & More, R. (2016). International Journal on Recent and Innovation Trends in Computing and Communication Review on Intelligent Braking System. http://www.ijritcc.org

[8] EasyEDA - Online PCB design & circuit simulator. (n.d.). Retrieved January 27, 2021, from https://easyeda.com/

[9] Garethiya, S., Ujjainiya, L., & Dudhwadkar, V. (2015). Predictive vehicle collision avoidance system using Raspberry - pi. ARPN Journal of Engineering and Applied Sciences, 10(8), 3655–3659.

[10] Gokulakrishnan, P., Ragavan, M. N. V. V., & Suresh, M. K. (n.d.). SIBS: Smart Intelligent Braking System to Eliminate Human Perception and Reaction Time using VANET. Retrieved June 1, 2020, from http://www.ijpam.eu

[11] Hemalatha, Pooja, P., T, R. R., & Professor, A. (2007). Automatic Braking System for Automobiles Using IR Sensor. International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering (An ISO, 3297(5). https://doi.org/10.15662/IJAREEIE.2016.0505111

[12] I2C 16x2(1602) LCD Display for Arduno - DFRobot. (n.d.). Retrieved January 27, 2021, from https://www.dfrobot.com/product-135.html

[13] Jual PIEZO BUZZER INDUSTRIAL ALARM 3v-24v 95db CONTINUOUS - HITAM - Jakarta Barat - ARDUSHOP-ID | Tokopedia. (n.d.). Retrieved June 5, 2020, from https://www.tokopedia.com/ardushopid/piezo-buzzer-industrial-alarm-3v-24v-95db-continuous-hitam

[14] Khairul, M., Kamarudin, A., Wahab, N. A., Umar, R., Shakir, A., & Saudi, M. (2018). Road Traffic Accident in Malaysia: Trends , Selected Underlying , Determinants and Status Intervention. 7, 112–117.

[15] kirti Engineering, C. (2016). Car Collision Avoidance System. 4(Viii), 90–92.

[16] NUMEX 12 volt Dc Motor (Multicolor) Price in India - Buy NUMEX 12 volt Dc Motor (Multicolor) online at Flipkart.com. (n.d.). Retrieved June 5, 2020, from https://www.flipkart.com/numex-12-volt-dc-motor-multicolor/p/itmfl2vna2fhuc6j5
[17] *PyCharm: the Python IDE for Professional Developers by JetBrains*. (n.d.). Retrieved January 27, 2021, from https://www.jetbrains.com/pycharm/

[18] Ram, Jvs., & Kumar, Br. (2017). Automatic Braking System Using Ultrasonic Sensor. In *International Journal of Innovative Science and Research Technology* (Vol. 2, Issue 4). www.ijisrt.com

[19] *Raspberry Pi 4 Now Available With 8GB of RAM, 64-Bit OS - ExtremeTech*. (n.d.). Retrieved June 5, 2020, from https://www.extremetech.com/computing/311086-raspberry-pi-4-now-available-with-8gb-of-ram-64-bit-os

[20] *RealVNC® - Remote access software for desktop and mobile | RealVNC*. (n.d.). Retrieved January 27, 2021, from https://www.realvnc.com/en/

[21] *Resistor Details, Description & Pinout*. (n.d.). Retrieved January 27, 2021, from https://components101.com/resistor

[22] *RPI 8MP CAMERA BOARD - Raspberry-pi - Daughter Board, Raspberry Pi Camera Board, Version 2*. (n.d.). Retrieved January 27, 2021, from https://my.element14.com/raspberry-pi/rpi-8mp-camera-board/raspberry-pi-camera-board-v2/dp/2510728?CMP=KNC-GMY-GEN-DSA-TOP-MANU-RASPBERRY-Pl&mckv=sFOyYU85M_dc%7Cpcrid%7C487815930919%7Cpkw%7C7Cpmt%7Cb%7Cslid%7C7Cproduct%7C7Cpgrid%7C119081676121%7Cptaid%7Cdsa-109877826040%7C&gclid=Cj0KCQiAmLaABhDFARIsAKywVafcDvVU-GpdcsJqN3wcoMF9WuLmQMUFM2d74ZudlwOIRAE9ObAx7-Ma11pEALw_weB

[23] *Single Channel 5V Relay Breakout Board*. (n.d.). Retrieved January 27, 2021, from https://my.cytron.io/p-single-channel-5v-relay-breakout-board

[24] *Ultrasonic Sensor HC-SR04 – NIRT JO*. (n.d.). Retrieved June 5, 2020, from https://nirtjo.com/product/ultrasonic-sensor-hc-sr04/

[25] Wang, Y., & Zhang, Q. (2013). Design of an active automotive safety system. *Journal of Engineering Science and Technology Review*, 6(2), 155–159.

[26] *What is an LED?*. (n.d.). Retrieved June 5, 2020, from http://blog.sparkfuseducation.com/what-is-an-led