ASHESI UNIVERSITY

CAR PARKING MANAGEMENT SYSTEM

APPLIED PROJECT

B.Sc. Computer Engineering

Jason Mawuena Kwaku Agbobli

2020

ASHESI UNIVERSITY

CAR PARKING MANAGEMENT SYSTEM

APPLIED PROJECT

CAPSTONE PROJECT

Capstone Project submitted to the Department of Engineering, Ashesi University in partial fulfilment of the requirements for the award of Bachelor of Science degree in Computer Engineering.

Jason Mawuena Kwaku Agbobli

2020

DECLARATION

I hereby declare that this capstone is the result of my own original work and that no part of it has been presented for another degree in this university or elsewhere.

Candidate's Signature:

……………………………………………………………………………………………

Candidate's Name:

…… Jason Mawuena Kwaku Agbobli………………………………….

Date:

…Friday 29th May, 2020……………………………………………………

I hereby declare that preparation and presentation of this capstone were supervised in accordance with the guidelines on supervision of capstone laid down by Ashesi University.

Supervisor's Signature:

……………………………………………………………………………………………

Supervisor's Name:

……………………………………………………………………………………………

Date:

……………………………………………………………………………………………

Acknowledgements

I have put effort into this project but none of this would have been possible without God, the kind support of certain individuals and Ashesi University as a whole. I would like to express my deepest gratitude to my supervisor, Dr. Nathan Amanquah, whose suggestions and guidance helped me in the undertaking of this project. I would also like to thank my parents for their constant encouragement and prayers that have carried me through these four years. The other members of the Engineering faculty, who provided critique and guidance during this project also deserve my appreciation as well as my friends who supported and helped me.

Abstract

Finding a vacant parking spot is important to individuals to save time which can be spent on other things. The increase in the number of cars in the country has led to a difficulty in locating vacant parking spots and the increase in importance of car parking lot and structure management. To make it easier for drivers to locate vacant parking spots within structures or parking lots, some of the most common technologies were explored and applied to come up with a solution to this growing problem. This project provides a solution to this problem using sensors for car detection and displaying the status of the parking slots to the driver.

Table of Content

Chapter 1: Introduction

1.1 Background

According to Graphic Online, more than one million vehicles of all types were imported into Ghana between 2005 and 2016[1]. The large number of cars has led to parking for long hours every day and this has made parking an important factor within the transportation system[2]. With the increasing number of cars in the big cities, difficulty in locating vacant parking spaces has become a huge problem.

1.2 Objectives/Motivation

Car parking slots, whether in parking structures or parking lots can pose difficulties when a driver is in search of an available space. The aim of this project is to reduce the amount of time wasted and make it easier and more convenient for drivers to locate vacant spots. The main objectives can be found below:

- To help drivers know the availability of a parking spot and the exact location.
- To minimize cost associated with car parking solutions and to create a solution tailored to the Ghanaian market with features Ghanaians will find useful.
- To use the most efficient way to detect the availability of car parking spots within the parking location and display to the user.
- To make a solution that can be easily integrated into parking lots and storied car parking structures.

1.3 Scope of work

The difficulty in determining or locating vacant parking spots may lead to a waste of time for drivers. Common solutions to this problem employ the use of various types of sensors, applications and displays to make it easier for the driver to locate vacant parking spots. This project will inform users of the availability of parking spaces in a parking lot or structure and where exactly they are located by displaying them at the entrance and on a mobile application for the user. However, it will not include ticketing or pricing and navigation to that parking spot.

Chapter 2: Literature Review

2.1 Research/Related Work

There have been numerous attempts at a solution to ease the stress of locating vacant parking spots for drivers to park their cars. Some of these have been highlighted below:

One of these solutions was a project on the prototype of an underground multi-storied car parking system[3]. This project made use of two circular floors, each with the capacity to contain six cars and the ability to rotate 360 degrees. This particular model was developed using a DC motor and various proximity sensors to detect and transport the car to a particular parking spot using a lift. When a car enters the lift, the proximity sensor which is installed there, detects it, and communicates to the main controller. The software then checks for the nearest available parking slot, displays it to the user and asks for a password for security. The lift is then activated to take the car to the allocated slot. The conveyor belts on the lift mechanism and on the parking slot coordinate and rotate to transport the car into its allocated space. Once the car is parked, the count will be increased by one on that particular floor. This entire process is reversible for retrieval of car. This idea is very interesting and is a very effective solution. It involves minimal work from the user, as he or she can park on the lift and the rest of the work is automated. However, Ghana is a developing country and has very few multi-storied car parks to boast of and building one from scratch could create various problems. The circular floors rotate continually about a shaft and this can cause problems because of friction. This will need continual maintenance such as oiling and changing of parts which will become an albatross around the necks of those who implement it because of the number of moving parts involved. Also, because of the number of components involved in this implementation, the cost will be very high, and the parking system proposed in this project seeks to minimise cost as much as possible.

A smart car parking system with a monitoring system[4] has also been developed. That project uses a microcontroller which communicates with various components with the help of an internet connection. The solution enables the driver's car to be parked in a garage by providing navigation by use of a smartphone with the help of GPS and Bluetooth. The car can then be monitored by a user wirelessly over a distance, using their smartphones. It consists of Microcontroller (for processing and control of the whole system), SONAR and Accelerometer (for navigation), and other components. The main aim is to provide navigation and tracking of user system. The monitoring of the system provides unique data to the administrator and can be used to integrate other functionalities that will be helpful to the driver. However, the dependence on an internet connection and use of GPS may not be convenient for all. Not everyone in Ghana has access to a reliable internet connection and the lack of fully accessible roads could create problems.

A real time car parking system using image processing[5] was also built. In this project, images of cars entering the parking lot are captured in intervals of two seconds. These images are saved as references and a camera capturing in real time only allows the car in if it matches the reference image and depending on the occupancy of the parking system. The parking lot has a fixed occupancy and therefore the number of images are fixed. When one side of the parking lot is full, the car is directed toward the opposite side and when both sides are full, no car is allowed to enter. For the image processing, RGB to grey conversion is done and image enhancement to bring image in contrast to background to select a proper threshold level. The image is then matched using edge detection. The use of cameras would be cost inefficient as a camera costs way more than other sensors such as ultrasonic, electromagnetic and smaller sensors, which they argue will have to be embedded in the ground which they claim is complicated. Even though this is true, one could argue that using cameras incurs a higher one-time cost, Digital Signal Processing (DSP)-capable microprocessor systems are comparatively more expensive than their counterparts and the concept of image processing is relatively complex. Using image processing could reduce the accuracy of determination of parking spots if the right algorithm is not used and the angle of view of the camera could also leave blind spots.

In Ghana, Response One Limited has a parking facility management system that controls access to the car park using a boom gate, which only opens when there is a vacant parking slot available[6]. It also has automatic pay station and manual cashier features and acts as a central control station. The system can be integrated with CCTV cameras to improve management. The product being used by Response One focuses mainly on ticketing and is not optimised to properly manage the number of available slots and congestion could become a problem.

There has also been an attempt at a solution from the Kwame Nkrumah University of Science and Technology. The project employs the use of an image processing algorithm. It captures empty parking spaces from aerial images and saves them as reference[7]. That way, if the current image does not match then it knows that the spot is occupied. The images are sent from the cameras to the microcontroller via Wi-Fi. The issue is that the likelihood of a Wi-Fi or internet connection in a car park is quite low especially in Ghana and many people may not see the use of deploying it for the sole purpose of car park management. Also, it may require powerful processing to be able to effectively use image processing. Parking in the night could also pose a problem, as visibility is significantly reduced.

Another smart parking system uses the concept of reservation. It was developed in such a way that the user books using the Short Message Service (SMS). It also uses a micro-RTU (Remote Terminal Unit)[8] which interfaces the car park with the control system. This is very useful but a bit inefficient because a person who has reserved a particular parking space may decide not to show up for various reasons and that parking space would be unavailable because the person has not relinquished that space. Though the reservation lasts for a certain period of time, it can still lead to a waste of resources or at least a delay in parking assignment. Also, the Short Message Service (SMS) is more expensive than applications that make use of the internet. With more modern chat services with additional features like WhatsApp and telegram, users are likely to hardly interact with their SMS and these other services can offer more useful possibilities. These modern platforms can be sent locations and track them, can receive photos of parking spots at a lower cost and more.

A mobile parking app for shopping malls has been implemented to minimise the time spent driving around looking for available parking spaces. This was implemented using a Dijkstra algorithm called Breadth-First Search[9]. This project relies heavily on the use of a mobile application. This application can be a distraction to the driver using it. However, if it was used as a backup of some sort, that would be more appropriate, as less time will be spent using the app to find free space and could further reduce the waste of time for other drivers trying to park.

An automated parking system has also been implemented in four modules: sensor level, billing/payment, programming level and display level[10]. The sensor level used an ultrasonic sensor for detecting the presence of a car and transmits the data wirelessly using an Arduino uno and modem. The billing and payment collect information on drivers and stores on a cloud database and calculates payment based on how long the car is parked. For the programming module, the received data from the sensor is processed by an Arduino mega and transmits results to the server with the help of a modem. The final phase changes the colour of the LED in the parking slot to indicate to the user whether it is taken or vacant and displays on a mobile application in real time. This solution is very good and provided a lot of inspiration, but the use of a single ultrasonic sensor may cause reduced accuracy in the sensing of car presence. Also, the use of microchips instead of a development board could have made the solution more cost-effective.

A project proposed the use of ultrasonic sensors; two in this case[11]. The two ultrasonic sensors were placed above the car at a height of two meters and the other in front of the car. This was done to improve the accuracy of car detection in their system. The status of the parking slot was then communicated to an Arduino Mega via Xbee radio, which then turned on red or green LEDs to display the status of the slot. This was also done via Xbee radio. This project was very good, but a better alternative could be found for Xbee radio because it has short range, low data speed and transmission, as well as low network stability.

Findings during the research helped steer this project in the right direction. This project should contain sensors that will detect the presence of a car but should do so with improved accuracy. This solution should also be cost-effective, as there are more expensive solutions out there that not everyone can afford. This solution should also not be too complex or advanced so that it can easily be implemented or replicated.

2.2 Significance and contribution

This project should contribute a meaningful entry to the list of solutions to tackle the growing problem of vacant parking space location determination in Ghana. As mentioned earlier, this project seeks to provide an easy and efficient way for drivers to locate free parking spaces within car parks and structures, and a meaningful solution tailored to the Ghanaian market.

Chapter 3: Design

At the entrance of a car parking lot or structure, the user should be able to view the status of the various parking lots. The user should be able to tell this by indicator lights representing the various slots. Upon entering, the user should be able to view the status once again to refresh his or her memory or to confirm. Once the car is parked at a slot, the status of that spot should change.

3.1 Decision

After research and knowledge gained from various solutions, the system should contain sensors at individual parking spots to detect cars and send sensor data to a microcontroller (MCU). The microcontroller should communicate with a master MCU that displays the status using LEDs and a mobile application by uploading the results to a database. A general block diagram was then designed containing the necessary components for the parking system. The block diagram can be found in figure 3.0 below:

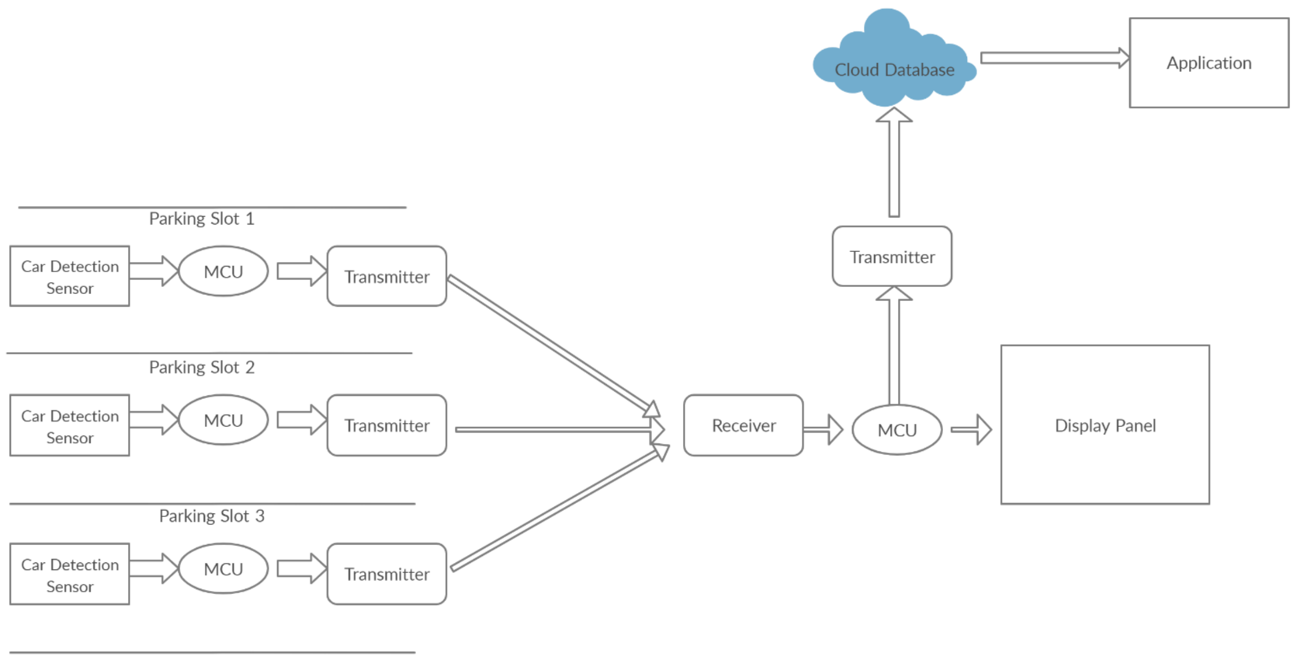


Fig 3.0 A General Block Diagram of car parking system involving three parking spots

The communication should be wireless for convenience and neatness of presentation. It is required to send only small bytes of data so it should not be a problem. The wireless communication should have a suitable range to send data to the microcontroller and the system will also need a power source since the whole system will need power to operate. There should be a display visible to the person entering the car park. In deciding the design and functionality of the system, some requirements of the system:

3.2 Functional and Non-Functional Requirements

- Should be able to view the status of a parking spot
- Should not drain too much power
- Should be able to last long before needing maintenance
- Should provide real-time information
- Display should be visible from distance
- Should be easy to understand and navigate
- Should save time for the user
- Should provide user information security
- Should be responsive to parked cars
- Should have uninterrupted power supply
- Should be power efficient
- Should be cost effective

- Should be quickly and easily deployable
- Should be tailored to Ghanaian market
- Should provide real-time information to user
- Should be easy to reproduce
- Should be able to communicate wirelessly with other components
- Should have minimal delay
- Should be able to communicate over a distance (entrance to parking slots)
- Should be available throughout the day or night
- Should work for about a year before needing to be changed
- Should not be intrusive
- Should be robust and durable (resistant to weather conditions)
- Display should be visible to user

3.3 Design Process

Using the above requirements, components were selected for the parking system. Decisions were made using Pugh matrices. A Pugh matrix is a criteria-based decision matrix which uses criteria scoring to determine which of several potential solutions or alternatives should be selected[12]. There are three scores in this Pugh matrix: ‘0’ (same as baseline), ‘+’ (greater than baseline and has a score of 1) and ‘-’ (lower than baseline and has a score of -1). After scores are tallied for all criteria, the total score is calculated, and a rank given to the alternative.

The first decision was for the sensor to be used for the detection of cars. The sensors involved in the comparison were ultrasonic sensors, weight sensor, magnetometer and pneumatic road tube, with a camera used as baseline. The ultrasonic sensor measures distance from an object using ultrasonic sound waves. It determines the distance by measuring time between the sending and receiving of ultrasonic pulse. The weight sensor works as a transducer. It converts a load into an electronic signal. Therefore, this sensor produces an output that is comparative to physical stimulus. A magnetometer is a device that measures the direction, strength or change in a magnetic field at a particular location. It detects large ferrous object by measuring change in the ambient magnetic field. A pneumatic road tube works by sending a burst of air pressure along a rubber tube as a vehicle passes over it. This pressure pulse closes an air switch which produces an electrical signal that is transmitted to an analysis software. The camera uses light from the object zooms into the camera lens. The image hits a sensor chip and is broken up into millions of pixels. The sensor measures the colour and brightness of each pixel and stores it as a number. The image from the camera is then processed to derive data. This is what is known as image processing. A few specifications of the various sensors[13] to be compared can be found in table 3.0 below:

Table 3.0 Specifications and features of sensors to be compared

Sensor	Specifications and features
Cameras	Work hand-in-hand with software for image processing. Easy implementation and can be installed on walls. Detection zones can be added by adding more cameras but there could be blind spots. Provide a wide area of detection but can be affected by poor lighting and weather conditions. The appropriate height to mount the camera for easy detection is about 50 to 60 inches.
Ultrasonic sensors	They transmit pulse waves between 25 to 50 kHz. They are useful for detecting vehicles of a certain height, can be used for multiple lane operation and are easy to install. Degraded occupancy measurement can occur due to large pulse repetition period. Weather conditions and temperature change can affect sensor performance but temperature compensation has been built into some models.
Weight sensors	Can be bending plate, piezoelectric, load cell or capacitance mat. Bending plate is accurate but costly and load cell are more accurate and cheaper. Capacitance mat can be portable or permanent, but it is the least accurate. The piezoelectric is cheapest but has weak sensitivity during weather and speed variation. They can mostly monitor up to 4 lanes and require replacement every 3 to 5 years.
Magnetometers	Can be installed both along the wall or the floor. Typically installed in a wireless sensor network. This device is portable and works within the temperature range of -40 to 85 degrees Celsius. It also has a supply voltage of 1.6 to about 3.6V.
Pneumatic road tubes	Offer a low-cost solution as well as quick installation and easy maintenance. However, they are temperature sensitive and there will be inaccuracies in axle count when the number of vehicles is high. Thy are also prone to vandalism.

The camera was used as a baseline because of the numerous existing research carried out using image processing. Based on research conducted on the various sensors, there was a comparison between them. The selected criteria were the efficient use of energy by the sensor, how durable it is, its cost, sensitivity or accuracy of the sensor, resistance to weather conditions, how intrusive its installation is, ease of use(simplicity), ability to interface with microcontroller and the range of the sensor. The Pugh matrix can be found in Figure 3.1 below:

			Alternatives						
	Criteria	Camera	Ultrasonic Sensor	Weight sensor	Magnetometer	Pneumatic Road Tube		Totals	Rank
1	Efficiency(Energy)	0	+	+	+	+		4	1
2	Durability	0	+	-	+	-		0	6
3	Cost	0	+	-	-	+		0	6
4	Sensitivity/Accuracy	0	+	0	+	0		2	2
5	Resistance(To weather conditions)	0	+	+	-	-		0	6
6	Intrusivity/Complication of installation	0	+	-	+	+		2	3
7	Simplicity	0	+	0	-	+		1	4
8	Interfacing with microcontroller	0	+	0	+	-		1	4
9	Range	0	0	-	0	-		-2	9
		Totals	8	-2	2	0			
		Rank	1	4	2	3			

Figure 3.1 Pugh Matrix for Determining Sensors

Using the above criteria, it was determined that the ultrasonic sensor was the best sensor to be used for the project. It had all the desirable characteristics needed to detect car presence.

The means of wireless communication was then decided based on the requirements. The contenders were Bluetooth, Radio Frequency (RF) module and Lora module, with Zigbee used as a baseline because it is fairly popular now. Other options were not explored because those would need internet connectivity. Bluetooth is a wireless standard used for exchanging data for short distances between mobile and fixed devices. The RF sends data over a frequency and consumes significantly low power. LoRa module provides very long range spread spectrum communication and high interference immunity. Zigbee is a low-power wireless mesh network standard targeted at battery-powered devices. A summary of the features and specifications of wireless communication[14] [15] can be found in table 3.1 below:

Table 3.1 Specifications and features of wireless technologies to be compared

Wireless Communication	**Specifications and features**
Zigbee	It transmits over a 2.4GHz frequency band and has a rate of 250kpbs. They are secured by 128-bit symmetric encryption. It has long battery life and is simpler and less expensive than Bluetooth and Wi-Fi. It has a nominal range of 10-1000m.
Bluetooth	It operates on a global 2.4GHz personal area network and has a signal rate of 720kbps. It is mostly used for device-to-device file transfers and wireless sound devices. It has a range of 10m.
Wi-Fi	It operates on 2.4 or 5 GHz frequency and has a signal rate of 54Mbps. Typically used for video, email and web applications that require high data rate connection. It has a range of 10-100m.
LoRa	It operates using license-free sub-GHz frequencies such as 433 and 868 MHz and has a data rate of 27kbps. It enables long-range transmission with low power consumption. It can range from 1 to 7km.
Radio Frequency	It operates over various license-free frequencies like 433MHz and has a data rate of 10kbps. It is typically used to send messages that require small bandwidth. It operates over a 3m (without antenna) to 100m radius.

The criteria for the Pugh matrix were the data rate, range, how much power the mode of communication consumes, probability of being a hacking target, reliability or stability of connection, cost, how widespread it is and carrying application. The Pugh matrix for communication can be found in Figure 3.2 below:

		Alternatives						
Criteria	Zigbee	Bluetooth Low Energy	Wi-Fi	LoRa Module	RF Receiver & Transmitter		Totals	Rank
Data Rate	0	0	+	+	–		1	2
Range	0	–	+	+	–		0	6
Power consumption	0	–	–	–	+		-2	9
Hacking Target	0	0	–	–	0		-2	8
Reliability	0	0	0	+	–		0	5
Cost	0	0	–	–	+		-1	7
Widespread	0	0	+	+	+		3	1
Varying application	0	0	+	+	–		1	2
	0						0	
	Totals	-2	1	2	-1			
	Rank	4	2	1	3			

Figure 3.2 Pugh Matrix for Wireless Communication

Using results from the Pugh matrix, the ideal mode of communication in this system will be the LoRa module, with Wi-Fi coming in second. However, because of cost, availability and the minimal data rate needed for this project, the RF receiver and transmitter were used.

A Pugh matrix was also used to determine the specific Microcontroller Unit (MCU) to be used in the system. The ATMEGA and PIC chip models were compared, with the MKL chip used as the baseline. The MKL chip was developed by NXP, the PIC chip was developed by Microchip Technology and the ATMEGA chip was developed by ATMEL. The MKL chip has an operating frequency of 48MHz and flash memory of 128kB and SRAM of 16kB. It has a supply voltage of 1.71V low and 3.6V high. The ATMEGA chip is a 28-pin AVR microcontroller with a flash memory of 32kB and SRAM of 2kB and a maximum of operating frequency 20MHz. It has a supply voltage of 1.8V low and 5.5V high. The PIC chip has a bus width of 8 bits, program memory of 14.3kB, CPU speed of 5 million instructions per second and a RAM size of 368 bytes. The voltage supply of 4.5V low and 18V high. The Pugh matrix can be found in Figure 3.4 below:

		Alternatives						
Criteria	MKL25Z128VLK4	ATMEGA328p	PIC16F877A				Totals	Rank
Ease of Use/Installation	0	+	+				2	1
Size	0	+	−				0	6
Cost	0	+	−				0	6
Energy Efficiency	0	+	−				0	6
Speed	0	−	+				0	6
							0	
							0	
							0	
							0	
	Totals	3	-1					
	Rank	1	2					

Figure 3.4 A Pugh Matrix for MCU

After the comparison was made, the ATMEGA chip turned out to be the most suitable MCU for the project. This was ideal because it was readily available.

After the various design decisions, the final block diagram was designed with more specifics. Two ultrasonic sensors will be placed at each parking slot to improve accuracy of car detection especially because of the bad nature of parking by some drivers. This will be done by making one of the sensors face up (90 degrees) and angling the second ultrasonic sensor to cover the length of a car. These sensors will be connected to an ATMEGA chip, which is the microcontroller in this case, and will send readings to the chip. The chip will in turn determine the presence of a car given the triggering distance for the ultrasonic sensors and transmit the status of the parking slot via RF 433MHz transmitter. Another ATMEGA chip at the entrance of the carp park or structure, receives the status of the parking slot and displays it using LEDs on a display at the entrance of the car park. It will also transmit the status of the car park to the MySQL database (which was chosen for its simplicity), which is connected to the mobile application and can serve as a portable alternate/additional display for the driver. The complete design diagram can be found in figure 3.5 below:

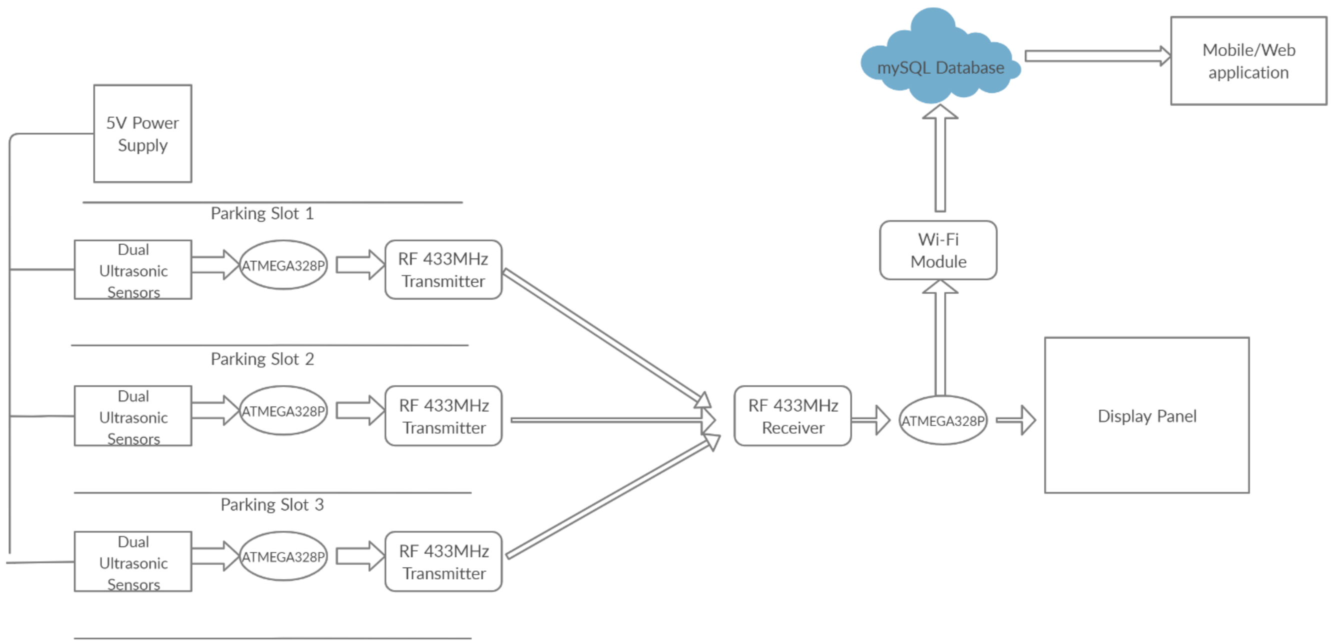


Figure 3.5 Design Diagram for Parking System involving three Parking spaces

Chapter 4: Implementing and Testing Methodology

4.1 Components

After the design process, the components for the car park management system were selected. They are as follows:

- **Ultrasonic sensors**

Ultrasonic sensors measure distance by ultrasonic waves. It emits these ultrasonic waves and receives the reflected wave back from the target. They measure the distance to the target by measuring the time between emission and reception. They transmit pulse waves between 25 to 50 kHz. It has a range of 2 to 400cm. The ultrasonic sensors will be used to detect the cars in the individual parking slots.

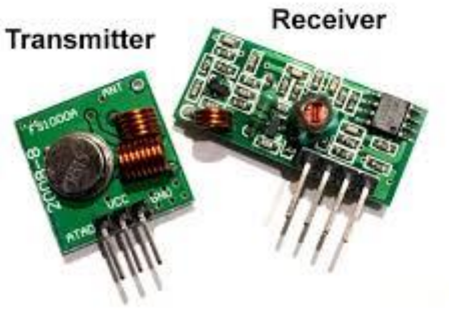
- **ATMEGA328P microcontrollers**

The ATMEGA328p chip is a high performance, low power AVR 8-bit microcontroller which executes instructions in a single clock cycle. These chips will serve as the microcontrollers for all the circuits in this project.

- **Red and Green LEDs**

A Light-Emitting Diode (LED) is a semiconductor light source that emits light when current flows through it. They will serve as indicators to users about the status of the parking spot. Green will be lit when spot is vacant and red will be lit when occupied.

- **433MHz Transmitter & Receiver**



They are a wireless transmitter (XD-FST) and receiver (XD-RF-5V) pair. These two components allow the ATMEGA chips to communicate wirelessly in the same frequency range. They will send sensor data from the various parking slots to the receiver ATMEGA chip.

- **9V Battery and battery clip**

It is a common size of battery that is used as a power source and can perform reliably across a wide range of devices. It has a rating of 9V and a capacity of 500 Milliamp hours (mAh) and a typical drain of 15 mA or when a load of 15mA is attached. This means that the battery can last for up to 33.3 hours. The battery clip is a terminal of a connecting wire that can be put on the terminals of a battery. It will be used to connect the battery to the rest of the circuit.

- **Voltage regulator**

It is a system designed to maintain a constant voltage level. It will be used to provide the system with a constant 5 volts by stepping down the 9V.

4.2 Circuits

The whole parking system is divided into two main parts which can be seen in the design diagram. The first part is made up of multiple ATMEGA chips (one per parking slot). There are 10k resistors connected from reset of each ATMEGA chip to VCC (the power source). This is done to curb the risk of random reset and electrical noise. There are two 0.1uF capacitors connected between the power pins that are used to get rid of noise and 16MHz resonator acts as a clock for the chip. A 9V battery was used as a power source and uses a voltage regulator to supply a constant 5 volts to the whole circuit. Two ultrasonic sensors will occupy one parking slot each. One will face directly up (90 degrees) and the other will be angled under the car (angle to be determined via testing) to cover the length of a car and improve accuracy. The sensor data will be aggregated and transmitted via a 433MHz transmitter to the other part of the system. A circuit of the first part of this system representing a single parking spot can be found in figure 4.0 below:

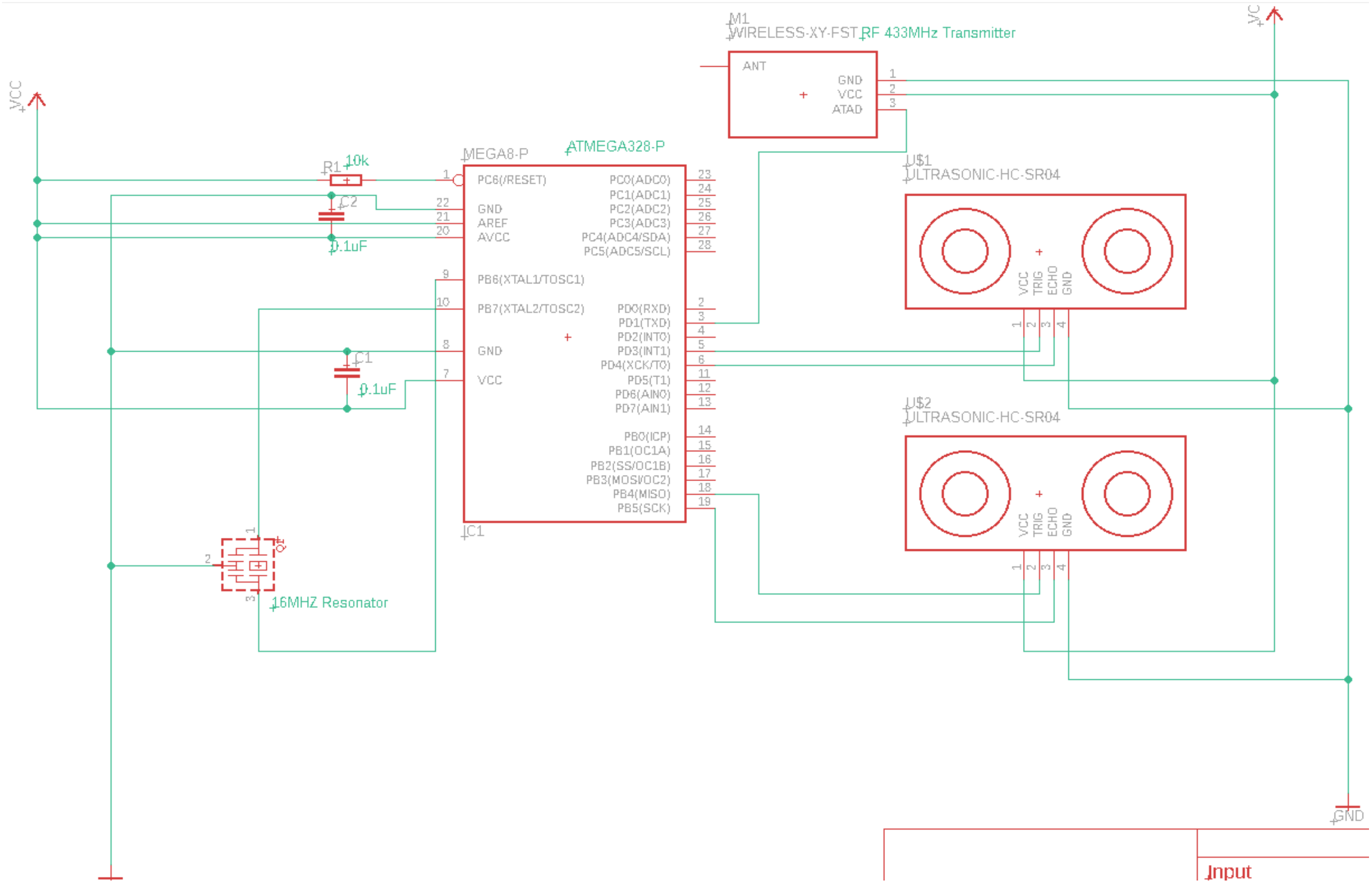


Figure 4.0 A circuit diagram of the car parking sensor for a single slot

The second part of this system is made up of another ATMEGA chip. This is a single chip that is once again connected to the 16MHZ resonator and the necessary resistors and capacitors. It has a 433MHz RF receiver to receive the status data from the individual parking spots and display the status of the slot using LEDs. There are red LEDs, to indicate to the user that the spot is occupied, and green LEDs to indicate to the user that the parking spot is vacant. There is also a Wi-Fi module attached to this circuit to upload the sensor data to the cloud database, which would in turn update the mobile application. A circuit diagram representing a circuit with three parking spots can be found in figure 4.1 below:

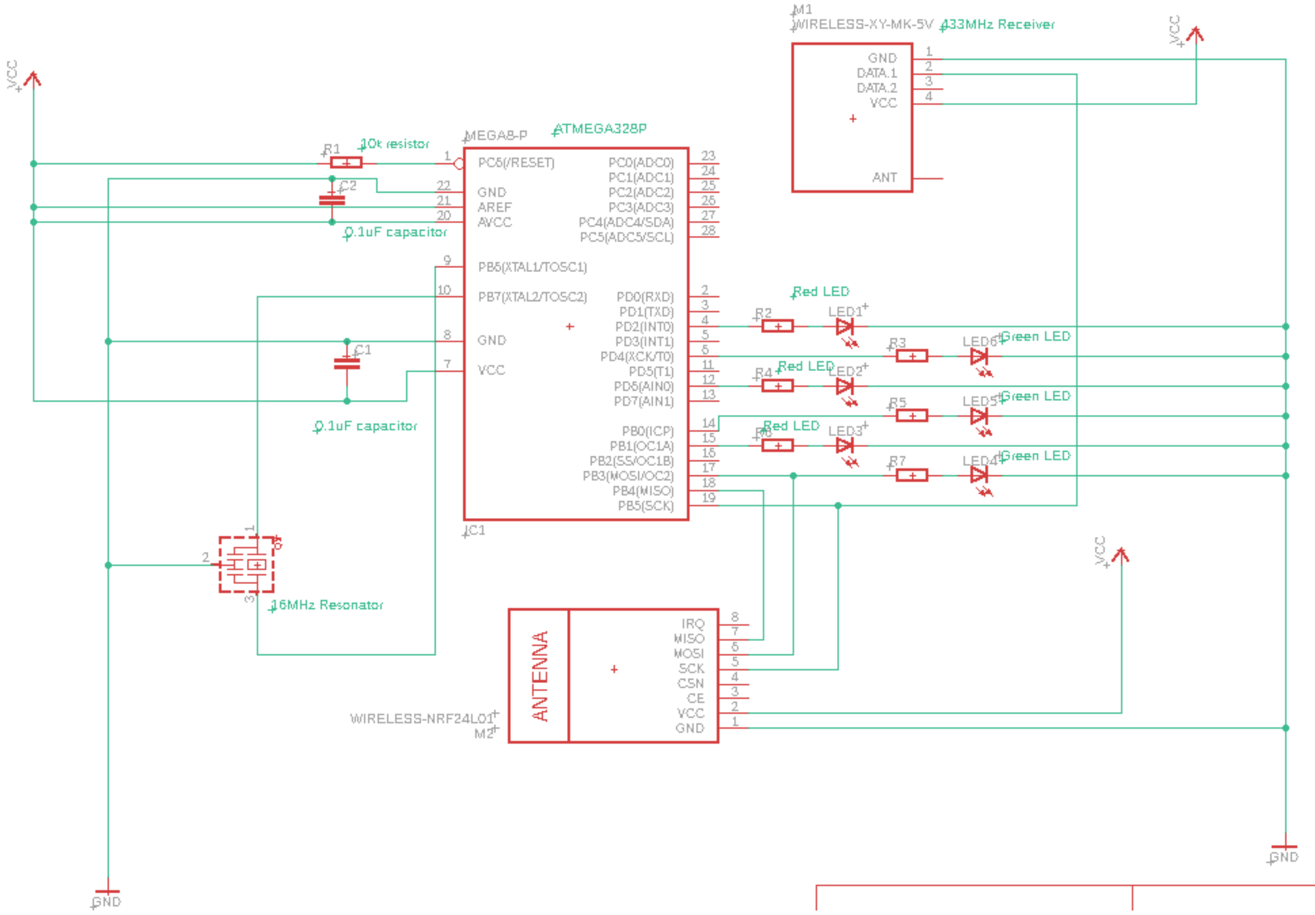


Figure 4.1 A circuit diagram of the receiver circuit and display for three parking spots

The next stage was to expand this concept to multiple parking spaces rather than just three because a three-slot car park is very simplistic. Therefore, two main ways of expanding the number of outputs were explored. The first was the use of decoders. Decoders are combinational circuits that have ‘n’ input lines with a maximum output of ‘2^n’ lines. The pins of the decoder can be activated by writing bits to the decoder for the corresponding output pins. Multiple decoders can be used to create even more output pins, but the downside is the inability to latch values. An illustration of the connection using a 4 to 10 decoder can be found in figure 4.2 below:

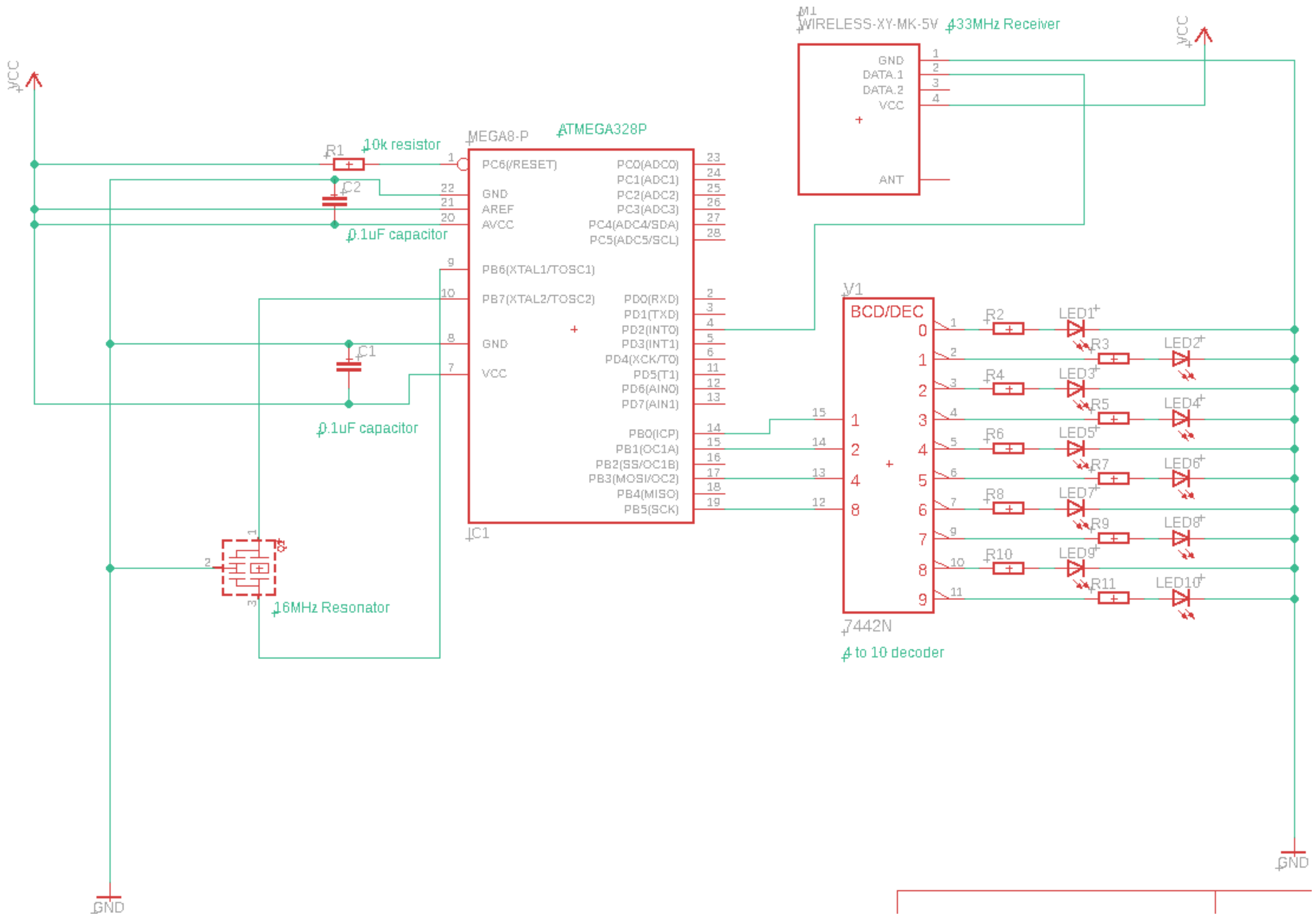


Figure 4.2 A circuit diagram of receiver and display for 5 parking spots using a 4 to 10 decoder

The decoder connection above allows for up to 5 parking spots to be represented (a pair of red and green LEDs represent one parking spot) using just 4 pins from the ATMEGA chip which would normally represent two parking spots but is still quite limited. The other method is by using a shift register. A shift register is a form of sequential logic that can store bits of information and shift them from left to right. They allow to expand output but need to be clocked. A circuit of this using two shift registers can be found in figure 4.3 below:

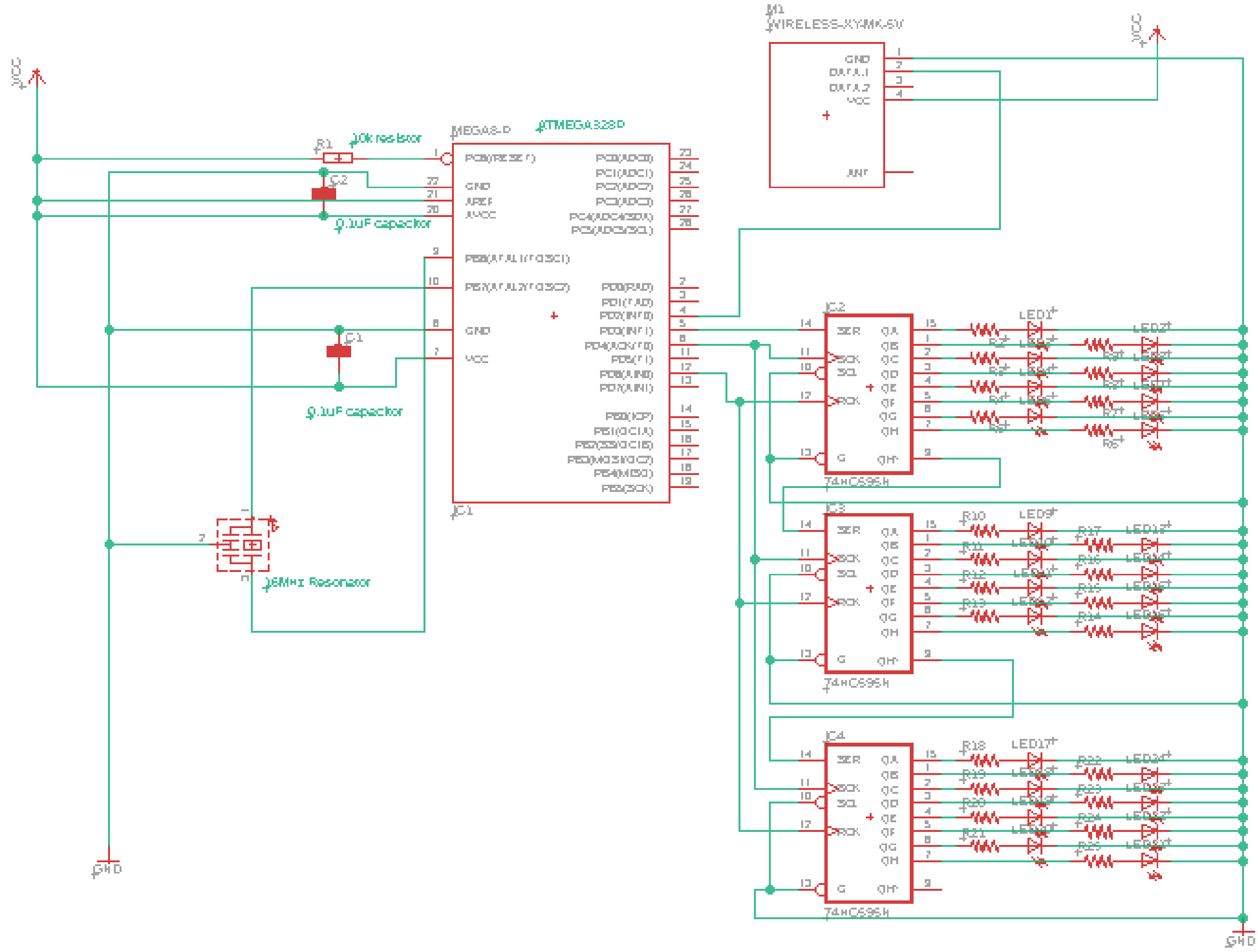


Figure 4.3 A circuit diagram for receiver and display for 12 parking spots using shift registers

The circuit above represents the receiver circuit for 12 parking spots using shift registers and uses only 3 output pins from the ATMEGA. This uses synchronous serial communication, which makes all output pins active and addressable at all times. The shift registers share the same clock and latch signal and the serial output pin of the preceding shift register becomes the serial input of the next. The great thing about using shift registers is that more of them can be added to this circuit to expand the output even further for larger car parks. It can also latch values (turn on multiple LEDs and keep them on).

To provide power to the whole system, a 9-volt battery will be used. It will be connected to a voltage regulator to provide a constant 5 volts. The numerous capacitors present in the circuit are meant to filter out noise by sending it to ground. The voltage regulator circuit that will provide a constant 5 volts to the whole system can be found in figure 4.4 below:

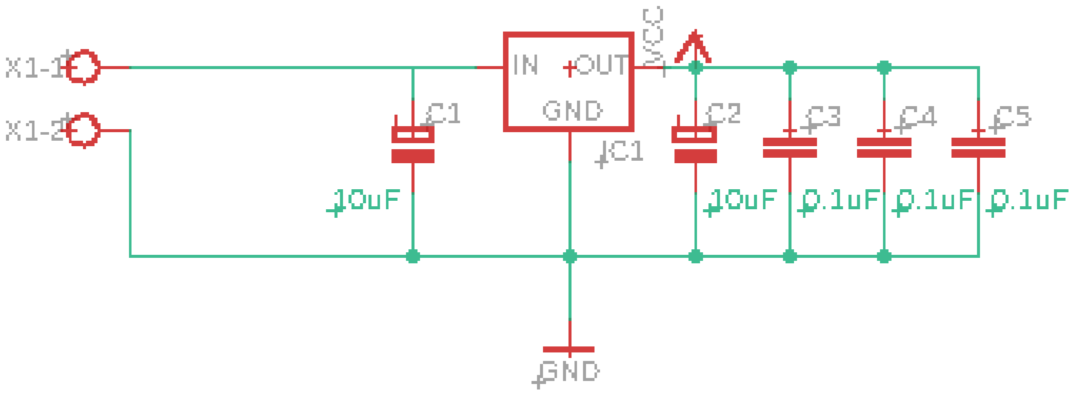


Figure 4.4 A voltage regulator circuit

The complete circuit was then constructed for the system with one slot to be used as a demonstration. The circuit on the left in figure 4.5 is the receiver. It consists of an ATMEGA chip with the components needed for setup, mentioned above, a radio frequency receiver and two LEDs, green and red. The circuit on the right in figure 4.5 is the transmitter circuit with the ATMEGA chip, two ultrasonic sensors, and a voltage regulator connected to a 9-volt battery to provide power. Both circuits are equipped with LEDs to indicate when they transmit or receive. The circuits can be found in figure 4.5 below:

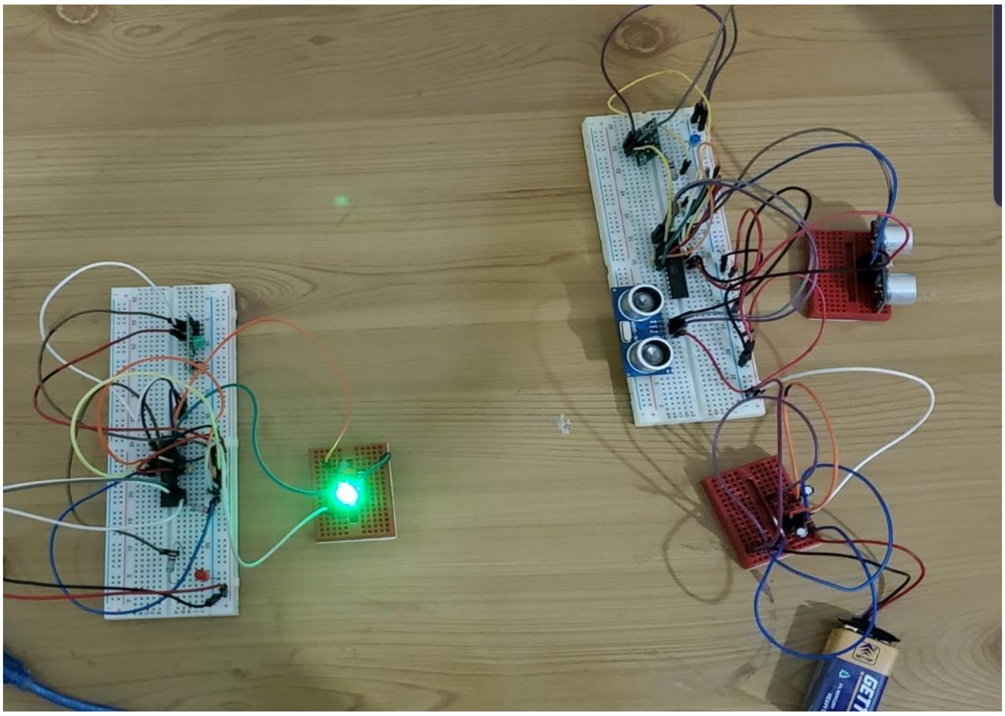


Figure 4.5 The receiver and transmitter circuit attached to the voltage regulator and power source (from left to right)

4.3 Software and Database

The database and the application to be used in the system were the next to be built. The database to be used was MySQL and the application was developed using ionic which is a tool used to develop applications.

Database

The database was created to hold the sensor data such as the ID, the status (vacant or occupied), reading and location. This database would be updated every time the status of a parking slot changed via the Wi-Fi module in the receiver circuit. This database was created to feed the app with the necessary data to update the status of the parking slot to be displayed to the user. The fields in the table can be found in figure 4.5 below:

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
	1	**SID**	tinyint(4)			No	*None*		AUTO_INCREMENT
	2	**Reading**	decimal(50,0)			No	*None*		
	3	**Status**	varchar(50)	utf8mb4_general_ci		No	*None*		
	4	**Location**	varchar(20)	utf8mb4_general_ci		No	*None*		

Figure 4.6 A screenshot of the fields for the sensor table.

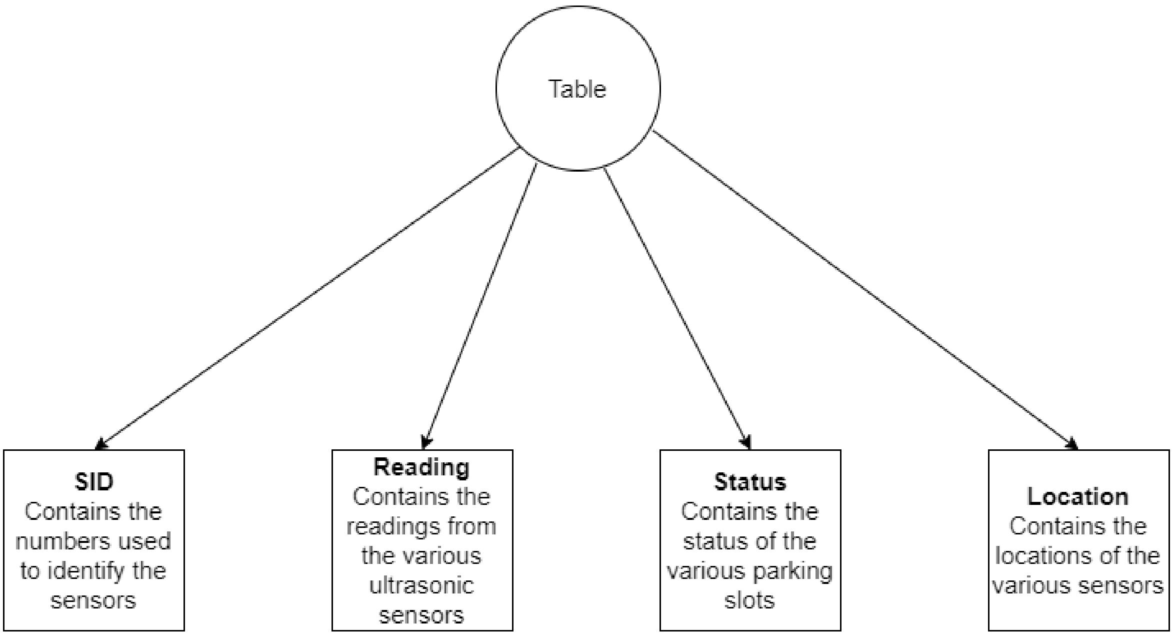


Figure 4.7 Diagram summarising the headings of the database

Web/mobile application

The application for this project was designed using ionic 4. The application required the user to sign up if the user did not have an existing account, or to sign in. The authentication for the sign up and sign in was done by linking the application to firebase, a google service used to improve the quality of mobile applications by providing authentication, security and integration with databases. This app is supposed to receive user details through login or signup and display the status of the parking slots in the parking structure or lot to the user. However, the application was not completed and only enables login or signup for the user. The dashboard of the developed application can be found in figure 4.7 below:

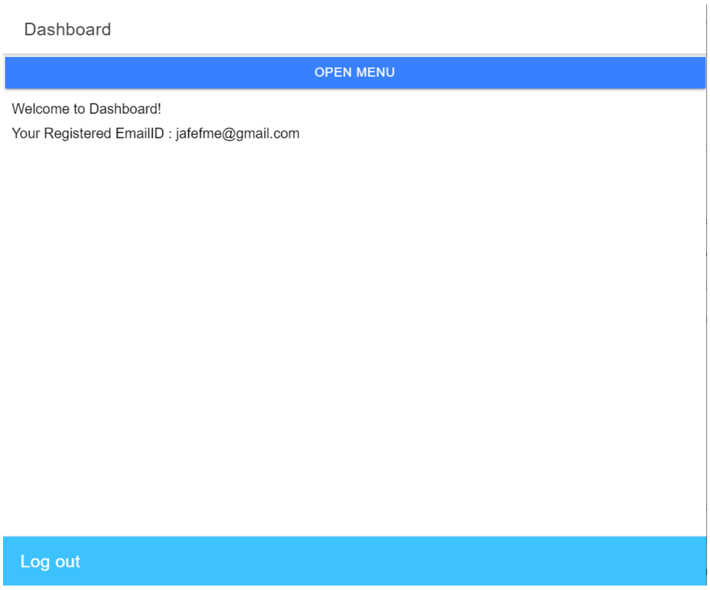


Figure 4.7 Dashboard for mobile application

4.4 Algorithm

The main cog that makes this whole system work, is the embedded system software. The algorithm was developed in the Arduino IDE and was uploaded to the individual components of the system so that the system could work properly. The status of a car park was continuously checked by the ultrasonic sensors and when the status changed, it would trigger the RF 433MHz transmitter to send a message to the receiver circuit to display the status to the user and upload to the database. An illustration of how the entire system works is in figure 4.8 below:

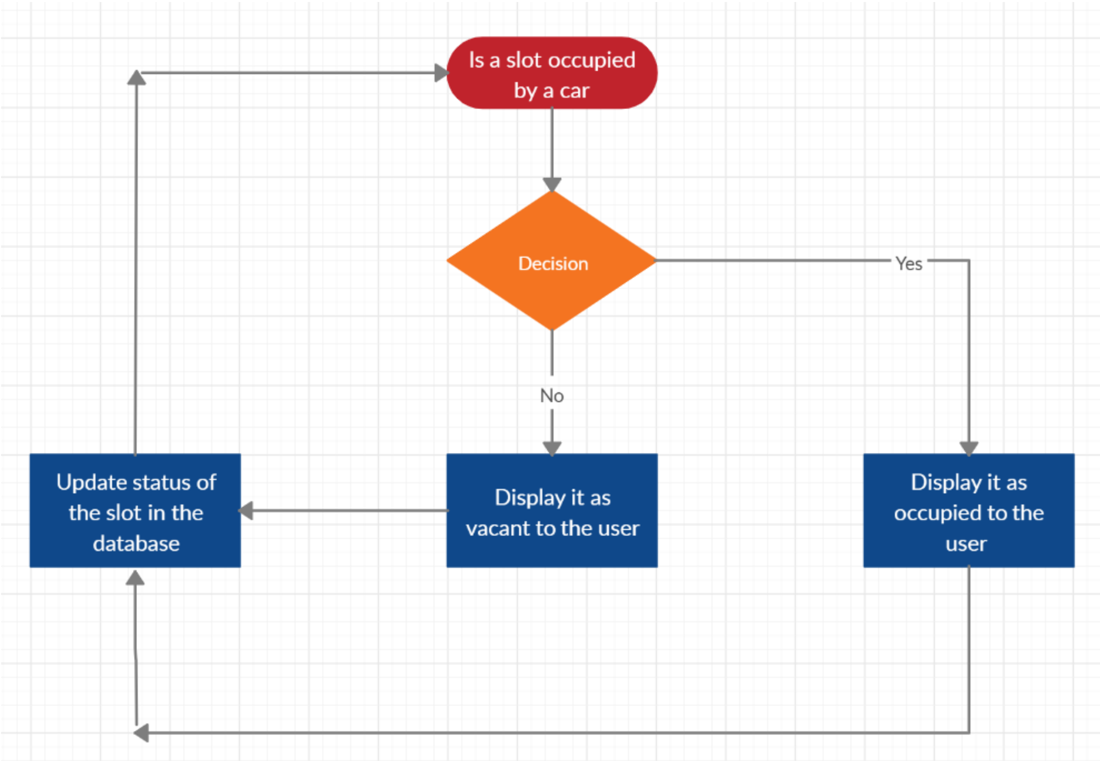


Figure 4.8 Flowchart of how the algorithm works

The entire system was divided into the transmitter and the receiver and so was the algorithm. The transmitter was equipped with two ultrasonic sensor and a transmitter. The ultrasonic sensors constantly took readings but one after the other, because the reading could not be taken parallel to each other. They were separated by a 10-microsecond delay and their distance was converted into centimetres. Immediately the ultrasonic sensor pointing up detected a distance less than 23.5 and the angled sensor detected a distance of less than 60, the transmitter sends a message to indicate that the slot is occupied.

The receiver is equipped with the various LEDs which correspond with the individual parking slots. It turns on the green LED when no message of occupancy is received to indicate vacancy and switches to red when a message is received to indicate occupancy. A delay is put in place to keep the red light on for 3 seconds before it checks again for the status of the parking slot to keep it on continuously without any fluctuation. Snippets of the code will be presented for the transmitter and receiver in Appendix A and B respectively.

4.5 Testing

Figure 4.9 Image of environment and car used for testing

The next step in this project was testing. The testing was done using a real-life situation to confirm the assumptions made in this project. The parking slot used was a sandy car park and a real car was involved in the testing. The car used was a Honda CR-V 2007 EX edition and has a ground clearance of 7.3 inches, height of 66.1 inches, wheelbase of 103.1 inches and a width of 71.6 inches. The sensor was placed under the car to confirm the triggering distance used in the algorithm and to calculate average response time of the system. The sensors were also placed at different positions under the car, to determine the best position for accurate detection.

Test 1: Test for triggering distance and angle

The average ground clearance (the distance between the road and the bottom of a car), of a Honda CR-V 2007 model is 7.3 inches (18.5 centimetres). This distance will be used with an extra 5 centimetres for error.

- The sensors were placed in the same position on the ground in the slot
- The car was driven over the sensor up to five times to determine if the sensor would detect the presence of the car.
- During this process, the second ultrasonic sensor (the one to be angled) had its angle changed in incremental steps to determine the best angle for detection.

Test 2: Test for average response time

The second test was very similar to the first but with a different goal and that was to find the average response time of the system.

- The sensors were placed on the ground in the same position so the car would be parked over them.
- The car was driven over the sensors three times.
- A stopwatch was used to determine the time used to turn on the LEDS from a distance when parking over the sensors.
- The same thing was done when the car moved away from the sensors.

Test 3: Test for appropriate position

The last test was done by continuously changing the position of the sensors underneath the car. This was done to determine the best position for the sensors to be placed at the parking spot.

- The sensors were shifted at different positions on the ground.
- First it was placed at the front part of the parking slot under the car.
- It was placed in the middle and at the back of the bottom of the car.

Chapter 5: Results

5.1 Results for Test 1: Test for triggering distance and angle

The first test was quite enlightening, and the results were as follows:

- At the beginning of the test, the sensor detected the car at almost every angle the second sensor was placed in but with a few switches between red and green. Results in table 4.0 below:

Table 4.0 Summary of responsiveness of second ultrasonic sensor at different angles

Angle (degrees)	Detection
30	Switching
45	Constant
60	Switching
80	Switching

- The sensor pointing upwards had no problems and detected the car in every situation which justified the triggering distance used.

It was determined that for the best results for the detection of a car, the angled sensor should be placed at 45°. This is because at 45°, the sensors detected the car and with no fluctuations.

5.2 Results for Test 2: Test for average response time

The main form of delay in the system is the wireless communication. The sensors are constantly detecting with a 10-microsecond delay between the two readings of the sensors.

There is also a one-second delay programmed into the system for timed, continuous transmission when a car is detected. The exit times were higher because there was a 3-second delay added to the system for the red lights to stay on to indicate occupation of the slot. The response time for the system were taken and Table 4.0 summarises this data.

Table 4.1 Summary of average response time of the system

Response time for entry (s)	Response time for exit(s)
0.94	3.20
0.95	3.21
0.85	3.44
Average: 0.91	**Average: 3.28**

5.3 Results for Test 3: Test for appropriate position

After changing the position of the sensors underneath the car continuously, it was determined that the car was detected best when the sensors were placed towards the edge of the front of the car (it detected the car just as it drove over the sensors).

- When placed in the middle, it detected the car, but the lights sometimes switched between red and green because the middle of the car is higher than the rest of the car and there is an uneven nature about the bottom of the car.
- When placed at the back of the parking slot, it detected immediately the car passed over it but switched back to green when the car parked fully into the slot.

Chapter 6: Conclusion

6.1 Discussion

The project was a success up to a point. The system was responsive to the presence of cars with favourable response time and the best detection was achieved at angle of 45°. The best position was also determined to be the front of the parking slot underneath the car. This project was able to help drivers know the status of a parking spot by the use of sensors and display and cost-effective solutions were used to make this a reality. The use of dual ultrasonic sensors to improve accuracy ensured efficiency and the solution was designed to be easily integrated into parking environments like lots and structures. The system worked very well but had some limitations.

6.2 Limitations

Some of the limitations this project suffered from are as follows:

- Access to materials and services that could have made this project better such as enclosure for the sensors, communication device to upload to database, complete application and Printed Circuit Boards (PCBs) for the circuits, were limited.
- Testing was difficult and limited because of inappropriate environment used.
- Lack of appropriate or the best devices such as the LoRa module because of cost and unavailability, stunted the potential of this project. The LoRa module would have been the best for communication because of its wide range.

6.3 Future Work

By learning from these limitations, some changes can be made to the project in the future:

- Alternative devices such as the LoRa module should be used for this project. The LoRa module will provide longer range for communication in the system.
- A more sustainable power source like solar should be used to make the system last longer before needing maintenance or change of batteries.
- Car parking structures and lots should be used for testing.
- More components should be acquired to enhance testing for multiple parking slots.

References

[1] “Graphic Communications Group Ltd. - Over 1 million vehicles imported into Ghana in 10 years” https://www.graphic.com.gh/news/general-news/over-1-million-vehicles-imported-into-ghana-in-10-years.html (accessed Oct. 13, 2019).

[2] E. Yorgri, C. Wen and L. Hong, "International Journal of Advances in Agricultural & Environmental Eng. (IJAAEE) Vol. 3, Issue 2", *Iicbe.org*, 2016. [Online]. Available: http://iicbe.org/upload/8899U0616305.pdf. [Accessed: 09- May- 2020].

[3] P. Eswaran, A. V. M. Manikandan, and S. Godha, “Prototype of an underground multi-storied automated car parking system,” in *2013 IEEE International Conference ON Emerging Trends in Computing, Communication and Nanotechnology (ICECCN)*, Mar. 2013, United States, pp. 674–677, doi: 10.1109/ICE-CCN.2013.6528585.

[4] M. Swatha and K. Pooja, “Smart Car Parking with Monitoring System,” in *2018 IEEE International Conference on System, Computation, Automation and Networking (ICSCA)*, Jul. 2018, pp. 1–5, doi: 10.1109/ICSCAN.2018.8541196.

[5] S. Banerjee, P. Choudekar, and M. K. Muju, “Real time car parking system using image processing,” in *2011 3rd International Conference on Electronics Computer Technology*, Apr. 2011, vol. 2, pp. 99–103, doi: 10.1109/ICECTECH.2011.5941663.

[6] “Response One Ghana - Parking Systems.” http://www.responseoneghana.com/parking-systems/ (accessed Oct. 13, 2019).

[7] B. Kommey, E. Addo, and A. Agbemenu, “A Smart Image Processing-based System for Parking Space Vacancy Management,” in International Journal of Computer Applications (0975 - 8887) Volume 182 - No.5, Jul. 2018.

[8] N. H. H. Mohamad Hanif, Mohd Hafiz Badiozaman, and H. Daud, “Smart parking reservation system using short message services (SMS),” in *2010 International Conference on Intelligent and Advanced Systems*, Jun. 2010, pp. 1–5, doi: 10.1109/ICIAS.2010.5716179.

[9] S. C. Khang, T. J. Hong, T. S. Chin, and S. Wang, “Wireless Mobile-Based Shopping Mall Car Parking System (WMCPS),” in *2010 IEEE Asia-Pacific Services Computing Conference*, Dec. 2010, pp. 573–577, doi: 10.1109/APSCC.2010.116.

[10] D. Azshwanth, M. T. Koshy, and M. T. Balachander, “Automated car parking system,” *Journal of Physics: Conference Series*, vol. 1362, p. 12059, Nov. 2019, doi: 10.1088/1742-6596/1362/1/012059.

[11] H. Alshamsi and V. Këpuska, “Smart Car Parking System,” *International Journal of Science and Technology*, vol. 5, pp. 390–395, Aug. 2016.

[12] C. Adams, “What is a Pugh Matrix?” https://www.modernanalyst.com/Careers/InterviewQuestions/tabid/128/ID/2159/What-is-a-Pugh-Matrix.aspx (accessed Apr. 15, 2020).

[13] M.Y.I. Idris, Y.Y. Leng, E.M. Tamil, N.M. Noor and Z. Razak “Car Park System: A Review of Smart Parking System and its Technology - SciAlert Responsive Version.” https://scialert.net/fulltextmobile/?doi=itj.2009.101.113 (accessed May 06, 2020).

[14] S. Chakkor, E. A. Cheikh, M. Baghouri, and A. Hajraoui, “Comparative Performance Analysis of Wireless Communication Protocols for Intelligent Sensors and Their Applications,” Sep. 2014, doi: 10.14569/IJACSA.2014.050413.

[15] “Anixter | Comparing Wireless Communication Protocols.” https://www.anixter.com/en_us/resources/literature/techbriefs/comparing-wireless-communication-protocols.html (accessed May 07, 2020).

Appendix

Appendix A (Algorithm for Transmitter)

```
#include <RH_ASK.h>
#include <SPI.h> // Not actually used but needed to compile
RH_ASK driver;

// defines pins numbers
const int trig1 = 9;
const int echo1 = 10;
const int echo2 = 8;
const int trig2 = 7;

// defines variables
long duration1;
long duration2;
int distance1;
int distance2;

void setup() {
  if (!driver.init())
  Serial.println("init failed");
  pinMode(trig1, OUTPUT); // Sets the trigPin as an Output
  pinMode(trig2, OUTPUT);
  pinMode(echo1, INPUT); // Sets the echoPin as an Input
  pinMode(echo2, INPUT);
  Serial.begin(9600); // Starts the serial communication
  pinMode(LED_BUILTIN, OUTPUT);
}

void loop() {
// Clears the trigPin
digitalWrite(trig1, LOW);
delayMicroseconds(10);

// Sets the trigPin on HIGH state for 10 micro seconds
```

```
digitalWrite(trig1, HIGH);
delayMicroseconds(10);
digitalWrite(trig1, LOW);

// Reads the echoPin, returns the sound wave travel time in microseconds
duration1 = pulseIn(echo1, HIGH);

// Calculating the distance
distance1= duration1*0.034/2;

// Prints the distance on the Serial Monitor
Serial.print("Distance1: ");
Serial.println(distance1);
delayMicroseconds(10);

digitalWrite(trig2, LOW);
delayMicroseconds(10);

digitalWrite(trig2, HIGH);
delayMicroseconds(10);
digitalWrite(trig2, LOW);

duration2 = pulseIn(echo2, HIGH);
distance2= duration2*0.034/2;

Serial.print("Distance2: ");
Serial.println(distance2);

//Condition for occupation
if (distance1 < 23.5 && distance2 < 60){
const char *msg = "OCCUPIED";
Serial.println(msg);
digitalWrite(LED_BUILTIN, HIGH);
driver.send((uint8_t *)msg, strlen(msg));
driver.waitPacketSent();
digitalWrite(LED_BUILTIN, LOW);
```

```
delay(1000);
}

else{
Serial.println("VACANT");
delay(100);
}
}
```

Appendix B (Algorithm for receiver)

```
#include <RH_ASK.h>
#include <SPI.h> // Not actualy used but needed to compile
RH_ASK driver;

//defines pin numbers
const int Red = 8;
const int Green = 7;

void setup()
{
  if (!driver.init())
         Serial.println("init failed");
    Serial.begin(9600);  // Debugging only
    pinMode(LED_BUILTIN, OUTPUT);
    pinMode(Red, OUTPUT);
    pinMode(Green, OUTPUT);
}

void loop()
{
    uint8_t buf[8];
    uint8_t buflen = sizeof(buf);
    if (driver.recv(buf, &buflen)) // Non-blocking
    {
      int i;
      digitalWrite(LED_BUILTIN, HIGH);
```

```
      // Message with a good checksum received, dump it.
      Serial.print("Message: ");
      Serial.println(String((char*)buf));

      digitalWrite(LED_BUILTIN, LOW);
      delay(1000);

      digitalWrite(Red, HIGH);
      digitalWrite(Green, LOW);
      delay(3000);

    }

    else{
      digitalWrite(Green, HIGH);
      digitalWrite(Red, LOW);
    }
}
```