IoT-Based Parking Monitoring System using Magnetometer as the Sensor

A J M Hutagalung, I K Sjamsudin, D P Hutabarat*

Computer Engineering Department, Faculty of Engineering, Bina Nusantara University, Jakarta, Indonesia, 11480

*Corresponding author: dhutabarat@binus.edu

Abstract. The IoT-Based Parking Monitoring System using Magnetometer as the Sensor is developed to detect the presence of cars in parking slots and to collect all the data from the sensor to be displayed in a mobile application so that user can easily find the empty parking slot. The system consists of four types of hardware and two types of software. This four hardware are magnetometer sensors, repeaters, gateways, and LED displays. Whereas the software are the hardware programming and a mobile application. The system developed has been successful in detecting all the present and absence of the car with 100% accuracy and displaying all the information needed into LED displays and mobile application for the service purposes.

Keywords: Internet of Things, Parking Lot Sensor, Parking System, HMC5883L

1. Introduction

Parking facilities are created in many places as the needs of finding a parking spot has become a crucial factor in accessing places or facilities [1]. Yet, frankly, finding a parking spot has become a problem. According to INRIX — a company that has analysed 35 million parking spaces across 100 countries, this parking problem costs drivers huge amount of time and money. After a large-scale survey of 18 thousand drivers in 30 cities across the US, UK, and Germany were carried out, it is concluded that parking is the second biggest problem the respondents experience after traffic congestions. 90% of the respondents also want a technology that helps to know the availability of parking spots in real time [2]. In addition to wasting a lot of time, circling around to find a parking spot is also causing several other problems, such as traffic congestion, driver stress, and waste of fuel — which produces excessive greenhouse gas pollution [3].

The number of IoT (Internet of Things) devices is expected to increase in the coming years and currently, there are many applications based on IoT have been developed such as IoT for emergency response systems [4], home security systems [5], smart equipment [6], etc. The system that was developed in this paper is an IoT based application that consists of four types of hardware, a software application, and a cloud server infrastructure containing the support software and the database system. HMC5883L magnetometers will be used to detect presence of cars [7]. The data generated from the detection will be sent to the cloud server via Wi-Fi and stored in a MySQL
database [8]. A software application intended for drivers wishing to find a parking spot in the form of a mobile application will be made using Flutter — a cross-platform mobile development framework [9].

2. Research Method
As shown in Fig. 1, the system consists of four types of hardware: sensor node, repeater node, gateway node, and LED display node. The sensor node sends parking information to the gateway node via the repeater node. The gateway node will be connected to the cloud server through the internet via Wi-Fi. The data sent by the gateway node will be handled and processed by the REST API before being stored in the database. The dashboard app and the mobile app will be the two applications that are used to view the parking data. The number of hardware devices in a location are customized to accommodate the parking slots. This paper assumes that the system is deployed in the parking area of Syahdan Campus of BINUS University in Jakarta which consists of roughly 60 parking slots.

The Sensor node is the device that responsible in deciding the availability of its parking spot and it is placed on the ground — below the car — in the middle of the parking spot. Fig. 2 show the developed magnetometer sensor while figure 3 is the block diagram of the sensor node.
The Repeater node is the device responsible in delivering messages throughout the whole parking space via the mesh networking protocol used in the system. The mesh networking protocol is also used to overcome the distance limitation of the wireless communication. HC-12, a 433 MHz wireless transceiver module, is used by all the hardware devices to communicate. The block diagram of repeater node can be seen in Fig. 4.

Gateway node is the brain of the whole system on-site. It receives information regarding the changes in all slots and take necessary actions by updating the cloud server and the number displayed on the corresponding LED display node. The LED display node will help approaching drivers to see the number of currently available parking slots on-site. Fig. 5 and 6 shows the block diagram of the Gateway node and the LED display node. The connection between all devices in the system can be seen through the flow diagram in Figure 7, while the display from the mobile application can be seen in Figures 8.
Figure 5. Block diagram of Gateway Node

Figure 6. Block diagram of LED Display Node

Figure 7. Flowchart of system

Figure 8. The mobile app GUI
3. Result and Analysis

Several tests have been conducted to measure the reliability of the system. The sensor node uses a magnetometer sensor to quantify its surrounding magnetic field. The presence of a car above the sensor node changes the surrounding magnetic field, thus changes the value read from the sensor. Fig. 7 shows how the quantified magnetic field reading change due to the absence and presence of a car.

![Figure 9. Magnetic Field Detection Graph](image)

The first test conducted was to see whether the sensor node can accurately detect the absence and presence of a car. The test was conducted across two different locations in a different city — each with two different cars with 20 samples taken from each car. The test concluded that the sensor node can detect the absence and presence of a car with 100% accuracy (shown in Table 1).

| Locations | Cars     | Empty He | Parked He | ΔHe  | Accuracy   |
|-----------|----------|----------|-----------|------|------------|
| Location 1| Car 1    | 4051     | 5183      | 1132 | 100%       |
|           | Car 2    | 4078     | 5872      | 1794 | 100%       |
| Location 2| Car 1    | 5713     | 7031      | 1318 | 100%       |
|           | Car 2    | 5688     | 7363      | 1675 | 100%       |

The second test taken was to measure the maximum distance the HC-12 wireless transceiver can correctly transmit/receive. The result of the test can be used to determine how far each repeater can be placed. The test was conducted by having two devices — one transmitting, one receiving (listening). The transmitting device transmits a 20-character long message while the other one listens. The listening device is carried apart by walking in a straight line until it fails to receive a complete message. The distance between the two devices are then measured with Google Maps.

There were three environment conditions for the test — open space, below car when off, and below car when on. Open space means the test was conducted in an open space and nothing was in the way blocking the transmission path. Below car when off means the transmitting device was placed below a car with its engine turned off. Below car when on means the transmitting device was placed below a car with its engine turned on. The result of the test can be seen on Table 2 which concludes that a blocking car can reduce the transmission distance particularly when the engine is on.
Table 2. Wireless transceiver maximum distance

| Locations | Environment Conditions |
|-----------|-------------------------|
|           | Open Space              | Below car when off | Below car when on |
| 1         | 120                     | 80                 | 60               |
| 2         | 120                     | 90                 | 60               |
| 3         | 120                     | 80                 | 60               |
| 4         | 120                     | 80                 | 60               |
| 5         | 120                     | 80                 | 60               |

The last test conducted was actually a series of tests to determine the time needed for a change in the parking lot to also apply in the next devices of the data flow direction, such as the mobile app, and, the LED display node — the last device of the data flow. Each and every significant process within each of these hardware devices, software applications, and supporting software are measured. These durations can be summed up together to find the time needed within the system. Arduino millis function was used to measure the duration of process in all hardware devices. A stopwatch was used to measure the time needed for a repeater node to receive transmissions from the rest of devices.

Late in the design process, a limitation was found in the repeater node. The repeater node was supposed to listen to two different wireless transceivers at the same time but was prevented by its use of the SoftwareSerial library. The implementation of the SoftwareSerial library currently only allows one instance of the SoftwareSerial class to listen to its connected UART device. A workaround to overcome this problem has been implemented by listening to only one instance at a time. However, because of how the workaround was implemented, only 50% chance of the messages were able to be received as the microcontroller might be listening to the other instance of SoftwareSerial at the time. Fortunately, due to the messaging protocol design, the sender will keep on sending its message until it received a reply from the repeater node acknowledging that the repeater has received the message. Hence, the message will eventually be received at a longer time.

Table 3 shows the effect of the limitation on the unpredictable durations. The test was conducted on two conditions. The first condition was the ideal condition where the repeater node was assumed capable of listening to two wireless transceivers at the same time. In reality, the test was taken using only one of the wireless transceivers to simulate the ideal condition. As a reminder, the repeater node is not capable of listening to two wireless transceivers at the same time and cannot operate in the ideal condition portrayed by the test. The other condition was when the workaround was applied in the repeater node source code. This test involves one device transmitting to the repeater node.

Fig. 9 illustrates the complete duration in the system. The total time is the result of summing up the durations in the system appropriate to the data flow illustrated by the curved arrows. As seen in the figure, the sensor node detection is one of the lengthiest process. This is because the sensor node awaits 8 seconds before redoing the magnetic field reading in a more accurate manner to reduce the chance of a false detection.
Table 3. Duration of transmission — ideal vs workaround applied

| No | Duration of Transmission | Ideal | Workaround Applied |
|----|--------------------------|-------|--------------------|
| 1  | 2.24                     | 6.78  |
| 2  | 2.05                     | 7.34  |
| 3  | 2.26                     | 5.13  |
| 4  | 1.98                     | 5.92  |
| 5  | 2.33                     | 13.61 |
| 6  | 2.14                     | 10.19 |
| 7  | 1.89                     | 9.84  |
| 8  | 1.78                     | 2.13  |
| 9  | 2.1                      | 6.15  |
| 10 | 1.95                     | 8.62  |
| 11 | 2.36                     | 2.66  |
| 12 | 1.82                     | 2.3   |
| 13 | 1.97                     | 5.77  |
| 14 | 2.48                     | 17.85 |
| 15 | 2.14                     | 8.13  |
| 16 | 2.35                     | 2.56  |
| 17 | 2.13                     | 10.74 |
| 18 | 2.09                     | 9.1   |
| 19 | 1.86                     | 6.76  |
| 20 | 1.93                     | 5.12  |
| **Average** | **2.09** | **7.34** |
| **Min**   | **1.82** | **2.13** |
| **Max**   | **2.48** | **17.85** |

Figure 10. Duration of the system

In this series of duration tests, a message collision test was also conducted. The purpose of this test was to find out what happens when there are multiple devices transmitting messages to the repeater node simultaneously. As mentioned previously, a message will eventually be received by the repeater. This should also apply when multiple devices are transmitting messages — which collide to one another — to the repeater at the same time, however, with an even longer time. The duration of the colliding messages that the repeater successfully received — from two, three, and four transmissions simultaneously — was recorded by a stopwatch and plotted into Fig. 9. A polynomial equation was created from the data to predict durations from an x number of simultaneous transmissions.
As an example, the duration from fifteen simultaneous transmissions can be calculated using the polynomial equation. The result is 317.5 seconds.

4. Conclusions
The developed system works very well which is indicated by the 100% accuracy of the magnetometer sensor to detect the presence and absence of the car. The system also has a reasonable time to update the information to the LED display node and mobile application in 41 and 25 second respectively.

References

[1] Departemen Perhubungan Darat. (1998). Pedoman Perencanaan dan Pengoperasian Fasilitas Parkir. Direktorat Jenderal Perhubungan Darat, Jakarta.

[2] Cookson G., & Pishue B. (2017). The Impact of Parking Pain in the US, UK, and Germany. INRIX Research.

[3] IoT Now Transport. (18 May 2018). Smart Parking in Montpellier relieves congestion and reduces search time. IoT Now Transport, [Online]. Available: https://iotnowtransport.com/2018/05/18/67416-smart-parking-montpellier-relieves-congestion-reduces-search-time/

[4] Syahputra M.E., & Hutabarat D.P. (2020). Eco Friendly Emergency Alert System (EFEAS) based on microcontroller and android application: Paper presented at the 3rd International Conference on Eco Engineering Development. https://doi.org/10.1088/1755-1315/426/1/012160

[5] Hutabarat D.P., & Budijono S, Saleh R. (2018). Development of home security system using ESP8266 and android smartphone as the monitoring tool: Paper presented at the 2nd International Conference on Eco Engineering Development. doi: https://doi.org/10.1088/1755-1315/195/1/012065

[6] Triantoro R., Chandra R., Hutabarat D.P. (2020). Multifunctional aromatherapy humidifier based on ESP8266 microcontroller and controlled using Android smartphone: Paper presented at the 3rd International Conference on Eco Engineering Development. doi: https://doi.org/10.1088/1755-1315/426/1/012152

[7] Ahmed A., Olaniyi M., Kolo J., & Oyetoro S. (2016). Developing Smart Car Parking System Using Wireless Sensor Networks. Proceeding of at the 2nd International Conference on Computing Research and Innovations, Nigeria, 1755, 201-206.
[8] Limantara A. D., Purnomo Y., & Mudjanarko S. W. (2017). Pemodelan Sistem Pelacakan Lot Parkir Kosong Berbasis Sensor Ultrasonic Dan Internet of Things (IoT) Pada Lahan Parkir Diluar Jalan. Prosiding Seminar Nasional Sains dan Teknologi 2017. Jakarta: Fakultas Teknik Muhammadiyah

[9] Dagne, L. (2019). Flutter for cross-platform App and SDK development. Metropolia University of Applied Sciences.