Semaphore & Slot-Acquisition based Parking System

Vignesh Shankar

Abstract: ***As we are aware that finding a parking space in most metropolitan areas, especially during the rush hours, is very difficult. This difficulty does not arise due to unavailability of parking spaces, but due to unawareness of the available spaces at that time; even if known, many vehicles may pursue very limited parking spaces that are nearby and cause serious traffic congestion. In this paper, we design a prototype of 'Semaphore and Slot Acquisition based Parking System (SSAPS)' that allows the drivers to effectively find and reserve the vacant parking spaces. The status of each parking sector is monitored and evaluated using principle of counting semaphore and is displayed at the entrance. This status is then updated on the main server. The drivers are allowed to access this cyber-physical system with their personal communication devices. Furthermore, we study the parking systems already in existence and compare their performance. The experiment results show that the proposed reservation-based parking policy evaluated using semaphores has the potential to simplify the operations of parking systems, as well as alleviate traffic congestion caused by parking searching.***

Index Terms: MUTEX, smart parking, semaphore, ZigBee

I. INTRODUCTION

In recent times, there is lot of congestion caused due to unavailability of parking lots. But there are cases where parking slots are available but they are either isolated, or too far from the entry point or not known to drivers. This usually results in more traffic congestion and air pollution by constantly cruising around the area to find a parking space. For instance, a recent survey [1] shows that during rush hours in most big cities, the traffic generated by cars searching for parking spaces takes up to 40% of the total traffic. To alleviate such traffic congestion and improve the convenience for drivers, many smart parking systems aiming to satisfy the involved parties (e.g., parking service providers and drivers) have been deployed. The current smart parking or parking guidance systems only obtain the availability of parking spaces from deployed sensor networks, and simply publishes the parking information to direct drivers.

However, since these systems cannot guide the drivers to their desired parking destinations, even sometimes make the situation worse; they are not "smart" enough. For instance, when the number of vacant spaces in an area is limited, more drivers, who obtain the parking information, start moving towards those spaces. It can also happen that while travelling to the particular parking space, some other vehicle has already occupied it. This results in severe congestion. This problem is effectively solved by using the principle of *semaphores* which will be discussed later.

In this paper, we design and implement the prototype of a Semaphore and Slot Acquisition based Parking System (SSAPS) not only to broadcast real-time parking information to the drivers as a part of a communal application, but also to provide reservation service as a part of user-targeted service. Built on advanced sensing and mobile communication techniques, SSAPS processes streams of time-stamped sensing data from sensor network in parking lot and publishes parking availability information. The drivers can retrieve parking information and reserve their desired vacant spaces via Wi-Fi or Internet.

II. EXISTING PARKING SYSTEMS

Many parking guidance systems have been developed over the past decade [2][3]. In this subsection, we study several existing parking guidance approaches and explain their drawbacks. Furthermore, we simulate these different parking management strategies under realistic traffic and parking conditions, compare their performance, and show results in section V.

- Blind Search: Blind searching is the simple strategy applied by users when there is no parking information. In this case, the drivers keep cruising for parking spaces within a certain distance to their destination. The drivers will stop searching until finding any available space. Otherwise, the drivers will extend the searching area and continuously look for vacant spaces in the neighboring parking lots.
- Parking Information Sharing (PIS): This mechanism is commonly adopted by the current state of the smart parking system design [4]. After the smart parking system publishes the parking availability information to the drivers in certain area, the driver will decide their desired parking destination where the parking lot has available spaces, according to the obtained parking availability information. However, if the number of vacant spaces in a parking lot is very limited in busy hours, it is likely that the number of drivers in demand for these parking spaces would increase. This phenomenon is called "multiple-car-chasing-single-space", which may cause severe congestion.
- *Buffered PIS (BPIS):* To address the problematic "multiple-car-chase-single-slot" phenomenon, some designers of smart parking systems modify the PIS mechanism. They intentionally reduce the number of vacant spaces, when publishing the live availability information, to keep a buffer. Therefore, though there may be more drivers pursuing the limited available spaces, the system has some extra spaces to avoid the conflict.

Revised Version Manuscript Received on November 13, 2016
Vignesh Shankar, B. Tech, Department of Electronics, Vivekanand Education Society's Institute of Technology, Mumbai (Maharashtra). India.

But it is difficult to determine the number of the buffer spaces. If the buffer is too small, the problem of "multiple-car-chase-single-space" will not be eliminated. If it is too large, the utilization of parking spaces will be low.

As cited above, the blind search system is an open loop system, where users make decision without looking at the state of the system. The PIS and BPIS strategies allow drivers to make decisions based on the system state (e.g., parking availability information). However, the problem of 'multiple-car-chase-single-space' cannot be fully eliminated. To reduce the traffic searching for parking, we suggest a *Semaphore* based system, where actual data regarding the availability of parking space is provided. This technique also overcomes the drawback of PIS systems. Also, slot acquisition facility is provided for drivers to make slot reservations using their communication devices through parking management system. If a driver makes the reservation successfully, it guarantees an available parking space for him, updates the parking database and *semaphore*s, and the driver can park at the reserved space without searching. The reservation-based system allows drivers to select the most convenient parking space under their budget constraints.

III. RESEARCH METHODOLOGY

In this section, we present the architecture and design of proposed Semaphore and Slot acquisition based parking system (SSAPS) which implements an efficient reservation service to reduce the traffic volume caused due to parking cruise.

A. Overview

Firstly, the information regarding availability or occupancy of parking spaces has to be updated. In already designed smart parking systems, this information is obtained using various sensor networks and updated on the server. Apparently, this leads to 'multiple-car-chase-single-space' problem. One of the features of SSAPS design is to unfold this issue. This is done by making use of *Semaphore*s. SSAPS uses principle of 'Counting *Semaphores*' and 'MUTEX *Semaphores*'.

In SSAPS design, users can reserve their slots either by booking it using their communication devices and internet or by contacting the person at the entrance of the parking lot. Fig. 1 shows the view of a parking lot. It describes various conditions that the users can encounter while booking slots. This is how the display looks while booking slots or the display that is put up at the entrance of the parking lot.

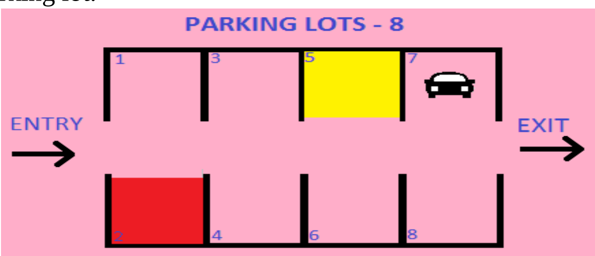

Fig.1 Top view of a parking lot

In the above figure, different colors indicated describe the current status of various parking slots. They are nothing but *Semaphore*s. It acts as a lock for the slots which are either accessed or occupied. It is clear that 7th slot is occupied already. For occupied slots, the *semaphore* value becomes 0 thereby preventing other users from accessing them. The 2nd slot which is highlighted by red color signifies that the particular slot is reserved or booked by some user. Again, the *semaphore* value for 2nd slot becomes 0 and prevents other users from accessing it. Now, the 5th slot is marked by yellow color which signifies that the slot is currently being 'accessed' (neither reserved nor occupied) by some user. Whenever the user accesses a slot for booking, the respective slot turns yellow and the *semaphore* value for that slot becomes 0. This addresses other users that the slot is currently being accessed by certain user and prevents other users from accessing it. Therefore, using the principle of *semaphore*s, it is impossible for two or more users to access or reserve a particular slot thereby preventing multiple vehicles to pursue a single slot for parking.

Initially, the value of counting *semaphore* is set to the total number of slots in the parking sector. As and when a particular slot is accessed or occupied, the value of the counting *semaphore* decrements. In this way, user can only access the slots that are neither accessed nor occupied (slots having *semaphore* value = 1). This is how counting *semaphore* acts as a key and solves the issue of 'multiple-car-chase-single-space.' Also, in some cases, the user accesses a slot for reservation but then cancels it. In such cases, the value of counting *semaphore* which was decremented when the user accessed the slot, increments thereby re-allowing other users to access the slot. The *semaphore* value of the parking slots is monitored continuously to check for its transition from 1 to 0 or vice versa and the information is updated to the main server frequently.

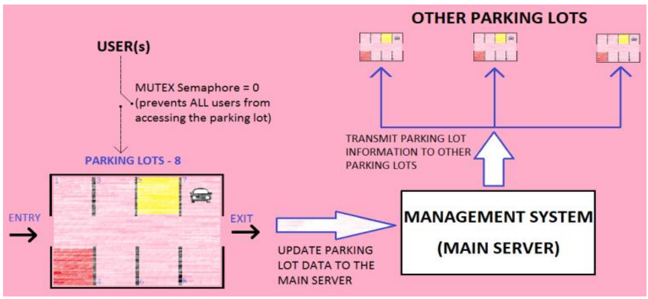

Fig.2 Principle or MUTEX *Semaphore*

Another feature of SSAPS is the MUTEX (mutual exclusive) *Semaphore* shown in Fig 2. This is a special *semaphore* which is accessed only by the main server. The main server uses this to load the parking lot data to the server and also to transmit the data to other parking lots. The main server makes the value of MUTEX *semaphore* equal to 1 and prevents 'ALL' users from accessing the parking slots. During this time, it loads the current information of parking lot to the server and transmits it to other parking lots as well. After this, the main server makes the MUTEX value as 0 and allows users to access the parking spaces. This updating of parking data to the main server is done in milliseconds and thus goes unnoticed by the users.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

MUTEX *semaphore* is used to avoid cases where while updating the information on the main server, some user may reserve a slot. This information will be ignored by the system because it is busy updating the information. In order to avoid such discrepancies in the data of users and parking system, MUTEX *semaphores* are used. This facility makes SSAPS model a smart design.

Fig. 3 shows three components SSAPS model, including parking lots, users and the parking system. The management system determines the parking prices and broadcasts live parking availability information to users (also drivers). Upon receiving parking information, the user selects a desired parking lot and reserves a space in the parking lot. As a result, the state of parking resources is changed by user's parking decisions.

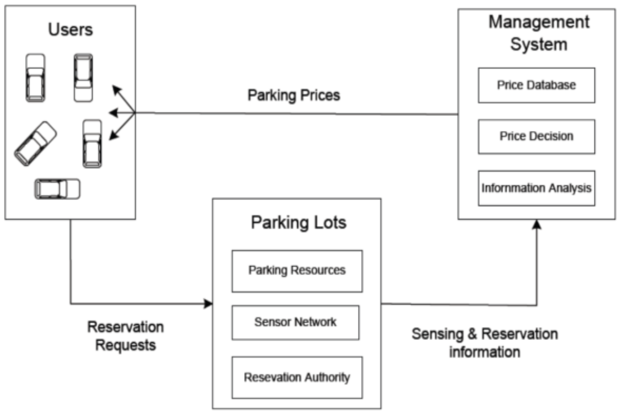

Fig.3 System Architecture

The parking lot consists of a group of parking spaces. The on-street parking can also be considered as a virtual parking lot. The state of a parking lot is the number of occupied spaces versus total spaces. Every parking lot has access to the Internet to communicate with the management system and users, and share parking information with other parking lots. In each parking lot, the reservation authority is deployed for authenticating the individual user's identity and reservation request. In this case, the reservation authority in the parking lot communicates with the specific user individually. Once the reservation order is confirmed, the reservation authority updates reservation information to hold the related space for the user. The sensor system and *semaphore*s deployed in parking lot is responsible for monitoring the real-time condition of parking lots and delivers the live aggregated sensing information (the number of available spaces or occupancy rate) to the smart parking system. The sensing information is updated on demand. Upon retrieving the parking information, the system updates the state of the parking lot. Based on the state of parking lots, the system (1) analyzes their occupancy status and congestion level, (2) determines the parking prices according to their pricing scheme, (3) broadcasts the prices to all users periodically, and (4) stores the parking information and prices for further analysis. The system serves as the centralized decision-making body in a planned economy. It makes all pricing decisions regarding the state of parking lots and user demands [5]. This system is a closed-loop system to dynamically adjust parking price, balance the benefits between users, and service providers and reduce traffic searching for parking.

B. Hardware

The system hardware is organized into three main components, the sensor network, the central server and the mobile device, as shown in Fig 4.

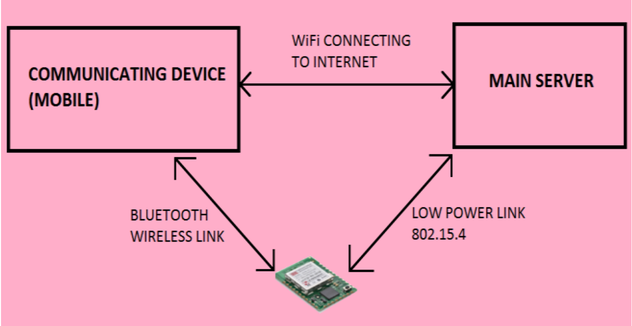

Fig.4 System Hardware Components

In our project, we developed a number of functions on ZigBee sensors that provide a continuous measure of parking status for each space [6]. Each sensor is integrated with two wireless motes. The wireless mote platform provides 802.15.4 wireless radio, 8-channel A/D and an 8 MHz microcontroller for on board digital signal processing. Mote 1 hosts light and vibration sensors, is used to detect the vehicles. In reality, the light sensor is easily interfered by light sources. So we use highly directional beam to strengthen light and reduce the interference. Mote 2 hosts the communication module of Bluetooth. As a result, the sensor bridges the communication between the ZigBee on mote and the Bluetooth module on smart phone (e.g., Android G1). In this case, the sensor confirms the identity of users when vehicle is detected in reserved parking lot.

The mobile phone is used to assess Internet, over Wi-Fi or a GSM cellular network, to obtain the information of parking availability and make parking reservation from the Internet server. The mobile phone also provides the Bluetooth module to communicate with sensors when verifying the driver's identity. Once user's reservation is authorized, the server will update the state of related parking sensor by wireless low power link, IEEE 802.15.4. [6]

C. Software

Software of SSAPS model can be implemented by two methods.

D. Remote APIs

API stands for 'Application Program Interface'. Remote APIs allow developers to manipulate remote resources through protocols, specific standards for communication that allow different technologies to work together, regardless of language or platform. For example, the *Java Database Connectivity API* allows developers to query many different types of databases with the same set of functions, while the *Java remote method invocation API* uses the Java Remote Method Protocol to allow invocation of functions that operate remotely, but appear local to the developer [9][10].

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

Therefore, remote APIs are useful in maintaining the object abstraction in object-oriented programming; a method call, executed locally on a proxy object, invokes the corresponding method on the remote object, using the remote protocol, and acquires the result to be used locally as return value. A modification on the proxy object will also result in a corresponding modification on the remote object [11].

E. Interfacing applications using iRev

Fig. 5 shows the design of system software architecture. Primary software elements are discussed in the following.

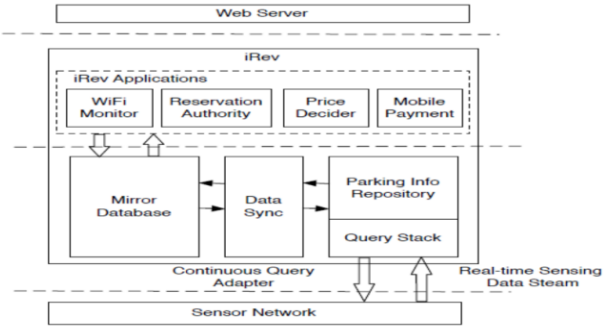

Fig.5 System Software Architecture

As the middle layer between the sensor network and web server, the iRev is the sink for all data sent from the lower sensor nodes. To simplify application development, we have developed query stack for delay-tolerant continuous query processing. Meanwhile, the parking information repository can retrieve real-time sensing data stream. To transfer data efficiently and reduce the complexity of the system, we have developed data synchronization processor that can connect and synchronize the data between parking information repository and mirror database.

Using any of the above 3 techniques, the user can check the parking price and make their reservation via mirror database, and data synchronization processor takes the responsibility to transfer related data to repository. In this case, the security of parking information is protected, as well as the information redundancy is reduced. Based on the real-time parking data, there are four specific applications in the iRev, including Wi-Fi monitor, reservation authority, price decider and smart payment. (1) Wi-Fi monitor is designed to monitor and report on Wi-Fi Access Points. It monitors and reports on performance, availability and problems. (2) Reservation authority is responsible for verifying the reservation process and driver's identity. (3) Price decider is to determine the parking price based the parking state. (4) Mobile payment allows drivers to use their mobile phones to pay parking fee.

The system dynamically updates the parking and reservation information (evaluated using *semaphores*) on the website according to the data stored in mirror database. The driver is able to obtain the real-time parking information and complete the reservation from the web server.

IV. EVALUATION

In this section, we simulate the proposed Semaphore and Slot Acquisition based Parking system (SSAPS) based on real traffic traces and real-world parking map, and demonstrate the performance of the SSAPS. In order to investigate the parking guidance policies and the proposed SSAPS, we have to develop a simulator to import the real-world map and traffic traces, simulate users' parking behaviors and implement related parking strategies.

- Import from Real Map: This simulator allows us to import a real-world map as the target area, and acquires the information from the map, e.g., distance and paths. Given the map, let G = (N; A) be traffic network defined by a set N of nodes and a set A of edges, where N and A represent the set of blocks and the set of roads connecting blocks. With the parking map, we aggregate the parking lots in one block as a virtual parking lot. Therefore, each node has a (virtual) parking lot attached, and an edge has a specific value assigned to represent the distance between two blocks.
- Parking Demand Modeling: In the simulation, we use the real-world traffic traces to generate the parking demand. Here the parking demand is the number of drivers who need parking spaces in the target area. However, in reality, it is difficult to collect the traffic traces for parking in the target area. Although the sensor network is deployed to monitor the incoming and outgoing traffic for parking in individual parking lot [4][7], the traffic data from individual parking lot cannot represent the total traffic traces for parking in the whole area. Fortunately, we can employ the highway traffic traces to estimate traffic for parking, which are available from the Performance Measurement System (PeMS) at the University of California, Berkeley. Here we make a general assumption that real total traffic for parking is proportional to the highway traffic. Although not all of traffic pursuing parking spaces in target area is from highways, and not all highway traffic need to park in the target area, the highway case can simulate the state of total traffic for parking. We classify the total highway traffic into incoming traffic and outgoing traffic, which represent the traffic approaching to and leaving from the target area. The incoming traffic serves as the reference of parking demand. We use the driving distance within the target area to measure the traffic for parking. For calculating the driving distance, the vehicles begin to run the distance meters, once they enter the area, until reaching the selected parking lot.
- Simulator Implementation: In order to implement different parking policies in the simulation environment, we use object-oriented design to realize the interactions between objects (e.g., users and parking lots). Although there are thousands of autonomous drivers who behave differently, they have the same common concerns about the parking selection, including convenience degree and parking price. So the object-oriented design provides us a cost-effective way to implement different behaviors and common concerns of objects through structures within the program. Meanwhile, by providing the interface of the objects, it is easy to extend the simulator to other applications.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

Therefore, the simulator is developed not only for reservation policy and dynamic pricing scheme, but to simulate general strategies of parking selections.

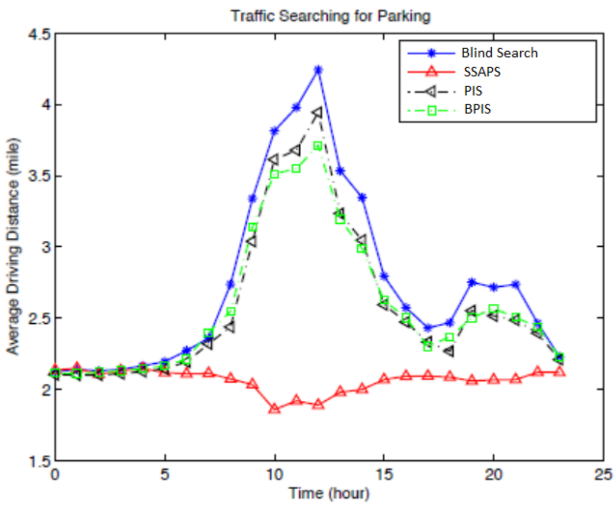

Fig. 6.Traffic searching for parking comparison under different parking guidance strategies

V. EXPERIMENTAL RESULTS

The following estimated experimental results illustrate the efficacy and feasibility of the proposed Semaphore and Slot Acquisition based Parking System (SSAPS) in a cost-effective way. We evaluate the effectiveness of reservation policy in terms of following perspective:

- Traffic Searching for Parking: Fig. 6 shows that the driving distance under blind search is the worst, especially during the peak hours; PIS and BPIS are better than blind search when traffic flow increases; and the reservation policy is the best compared with others. Note that, in this simulation, there is no pricing scheme implemented in reservation policy. An interesting observation of reservation-based policy is that the average driving distance is decreasing at peak time, rather than increasing. That is because, after users learn the states of parking lots, they tend to reserve the nearest parking lot to their destination.

During the peak hours, most parking lots are almost fully occupied in central area. Consequently, users have to select the parking lots in surrounding area, which are near to their start points. Therefore, it results in the reduction of average driving distance during the peak hours.

VI. RELATED WORK

Currently, most research work on smart parking is from the perspective of system design, which focuses on implementing a wireless sensor network to detect parking information and provide real-time parking service. In addition, we introduce the pricing-related topics in networks, which provide us a powerful tool to manage parking lots.

In [7], the authors present a smart parking management system based on wireless sensor network technology, which provides remote parking monitoring, automated guidance and parking reservation service. They demonstrate this system architecture can help commuters to find vacant parking spaces. Rongxing Lu et al [4] introduce a new *Smart Parking (SPARK)* scheme, which is based on *Vehicle Ad Hoc Net-works (VANET)*, provides drivers with accurate and convenient parking services in large parking lots, including real-time parking navigation, intelligent antitheft protection, and friendly parking information dissemination. In [8], authors proposed a scalable information dissemination algorithm for discovery of free parking spaces via VANET.

VII. CONCLUSION

In this paper, we have developed a new prototype of Semaphore and Slot Acquisition based Parking System (SSAPS) to optimize parking management. In this system, we implement parking reservation policy to balance the benefit of service providers and requirements from the users. Moreover, we have presented the detailed design, implementation and evaluation of the prototype. Based on the obtained results from our simulation study, we conclude that the proposed reservation-based smart parking system can alleviate traffic congestion caused parking searching and reduce the amount of traffic volume searching for parking.

REFERENCES

1. International Parking Institute (IPI) 2012 Emerging Trends in Parking Study.
2. One of the parking guidance systems developed until now. ”http://www.omnitecgroup.com/Solutions/parking-controls/parking-guidance-system”
3. One of the parking guidance systems developed until now. “http://www.autopassindia.com/parking_guidance_system.html”
4. R. Lu, X. Lin, H. Zhu and X. Shen, “SPARK: A New VANET-based Smart Parking Scheme for Large Parking Lots,” in Proceedings of IEEE INFOCOM’07, 2007.
5. Sullivan and S. Sheffrin, Economics: Principles in action, Pearson Prentice Hall, 2006.
6. Meng-Shiuan Pan and Yu-Chee Tseng, Computer Science Department, National Chiao Tung University, Taiwan. “http://people.cs.nctu.edu.tw/~yctseng/papers.pub/book-zigbee”
7. J. Chinrungrueng, U. Sunantachaikul and S. Triamlumlerd, “Smart Park-ing: an Application of optical Wireless Sensor Network,” in Proceedings of Application and the Internet Workshops, 2007.
8. M. Caliskan, D. Graupner and M. Mauve, “Decentralized Discovery of Free Parking Places,” in Proc. of the Third ACM International Workshop on Vehicular Ad Hoc Networks (VANET 2006), 2006.
9. Bierhoff, Kevin (23 April 2009). "API Protocol Compliance in Object-Oriented Software." CMU Institute for Software Research. Retrieved 29 July 2016.
10. Wilson M Jeff. "Get smart with proxies and RMI" JavaWorld. Retrieved 29 July, 2016.
11. Henning, Michi; Vinoski, Steve (1999). "Advanced CORBA Programming with C++". Addison-Wesley. ISBN 978-0201379273. Retrieved 16 June 2015.

AUTHORS PROFILE

Vignesh Shankar, currently a student of class D16A, B.E. Electronics. I am currently pursuing my final year at Vivekanand Education Society’s Institute of Technology. I have been a CSI member since two years and have been actively participating in various events conducted by CSI. Also, I was a part of a NGO program held at Mumbai which spread awareness regarding cancer. I do not have any publications, but have made 5 to 6 projects based on Electronics. Two of them were programming based, two were related to analog and digital circuits and two were based on sensors. With due interest in my field, I intend to pursue Masters in Electronics in the next year.