UNIVERSITY OF CAPE TOWN

DEPARTMENT OF COMPUTER SCIENCE.

THE DESIGN OF AN INTELLIGENT PARKING SYSTEM USING WIRELESS SENSOR NETWORKS AND MULTI-PROTOCOL LABEL SWITCHING.

A thesis submitted to the faculty of science as a partial fulfillment of the requirements for the award of the Degree:

MASTER OF SCIENCE IN INFORMATION TECHNOLOGY

BY

ANTHONY MWEBAZE

SUPERVISED BY

DR. ANTOINE BAGULA

August 2009

Plagiarism Declaration

I know the meaning of plagiarism and declare that all of the work in this document, except for that which is properly acknowledged, is my own. The content of this thesis has not been presented to any institution elsewhere for the award of a Master of Science In Information Technology. I have used the ACM citation and referencing style. I have not allowed and will not allow anyone to copy my work with the intention of passing it off as his or her own work.

Candidate: Anthony Mwebaze

Signed: …………………….

Date: ……………………….

Supervisor: **DR. ANTOINE BAGULA**

Signed: …………………

Date: ……………………

Acknowledgement

This study would have hardly reached fruition if it wasn't the helping hand of dependable persons. I express sincere gratitude to my supervisor Dr. Antoine Bagula who has guided me through the tough phase of research. You have given me a strong insight of further understanding of researching in new areas.

To the MIT course convener Dr. Audrey Mbogho, you have played a pivotal role in helping me find the right direction, thanks a bunch.

To the department of computer science, I appreciate your undying availability to help whenever I popped in with endless needs.

To Mr. Magezi Alex and Dr Mbabazi Christine, your support has been of paramount importance in the success of this study.

To all my close allies that I cannot fully mention here, thanks you so very much for all the support.

Much appreciated!!

Abstract

The challenge of parking management has increasingly posed the need for smart solutions. Motorists in today's busy world seek the best option in locating available parking points. The need for an efficient parking system stems from increased congestion, motor vehicle pollution, driver frustration and fatigue to mention but a few.

This study was conducted at a time when the world was experiencing a financial crisis and more than ever motorists needed intelligent parking systems to reduce the cost of gas spent driving around to find parking. Indeed, the time spent driving around would be beneficial if used to do work that would put one at an advantage in the credit recession.

The study was also conducted at a time when South Africa was preparing to host the 2010 soccer world cup. In the preparation to manage motor vehicle congestion, this study was a viable solution to manage the expected challenge of parking.

This study presents the design and illustrates the performance of an intelligent parking system based on an integrated architecture where (1) Wireless Sensor networks (WSNs) using Small Programmable Object Technology (SPOT) motes are launched into parking places to monitor the activity of the parking area through light intensity sensing and (2) the sensed information is gathered and channeled through a gateway into databases used for parking space visualization and information dissemination over the World Wide Web technology and mobile devices via a Multi Protocol label Switching (MPLS) network. Using an illustrative simulation model of a small parking system built around a new generation of SUNspot motes, the study demonstrates how a real life smart parking

system can be deployed to benefit motorists in today’s busy World and serves as a foundation to future work on how this emerging generation of motes can be used to provide better ways of finding parking.

I dedicate this piece of work to my most beloved parents, brothers and sisters.
May you be blessed abundantly.

Abbreviations

SPOT: Small Programmable Object Technology.
JDK: Java Development Kit.
SDK: Software Development Kit.
IDE: Integrated Development Environment.
CMS: Changeable message signs.
PDA: personal digital assistant.
TCP: Transmission control protocol.
MPLS**:** Multi protocol label switching
VMS: Variable Message Signs
RFID: Radio Frequency Identification
GPRS: General packet radio service
APIS: Advanced Parking Information system

Table of contents

LIST OF FIGURES

LIST OF TABLES

CHAPTER ONE:

INTRODUCTION

1.1 Background

In today's fast paced working environment, people (motorists) greatly depend on auto mobiles to commute to their destinations. Automobiles include: motor vehicles, motor bikes, trucks to mention but a few.

The use of these auto mobiles has increasingly posed a demand for infrastructure to manage the parking. All around the world, parking spaces have been constructed and control points put in place. For example, in shopping malls and airports, some control points are automated whereby users can do a self service in the use of the parking space while others are manned by control personnel. On the other hand, parking attendants have been employed in phycally controlled parking bays to direct drivers where parking is empty.

The systems in place today for managing parking areas have helped a lot in ensuring that motorists easily park and easily leave their destinations. However, the demands of motorists in the fast working environment raise a need for a next generation of parking systems to match the pace at which they work. It is expected that such next generation parking systems will enable remote parking reservation and exhibit some of the features of modern real-time systems such as cell phone payment and car identification using Radio Frequency Identification (RFID) technology. It is predicted that such smart parking systems will play an important role in the transport field in terms of environment impact on climate change and commuter's savings and time management. According to ITU [22] recommendation, the application of ICTs to transport through the development of intelligent transport systems (ITS) can be used to reduce the environment impact of transport systems by using for example parking guidance systems to lead motorists to the most appropriate parking space and thus reduce the hassle of finding parking space. The efficient management of parking lots using smart systems is also a parameter upon which climate strategies could rely on to reduce the effect of pollution.

This study was conducted at a time when the world was experiencing a financial crisis and more than ever motorists needed intelligent parking systems to reduce the cost of gas spent driving around to find parking. The time spent driving around would be beneficial if used to do work that would put one at an advantage in the credit recession. On the other hand, efficient parking management would also reduce the emission of gases in the environment contributing to climate change due to the effect of pollution on the ozone layer. The emission of a single car may seem insignificant but with the ever increasing number of cars on the road, the gases emitted are in huge volumes

The study was conducted two years before the 2010 soccer world cup in South Africa. At that time, preparations were under way to improve services to accommodate a high population. Among the services to be provided in the transport sector was parking management in particular. For example, there would be more tourists than ever in the country, thus a need to expand the infrastructure and the role of smart parking. Parking for motorists is one of the challenges that would be faced by the 2010 soccer organizing committee, the tourists, nationals and all vehicle users who will be in the country at the time.

Among the alternatives considered in the preparations to manage motor vehicle congestion, this study was a viable solution to manage the expected challenge of parking.

1.2 Problem statement

Users of automobiles spend a lot of time in the parking bays trying to locate where to park. In today's ever busy working environment, drivers hardly have time to spend in parking bays looking for where to park. In many places, especially around shopping complexes, universities, city centres, and many other busy working environments, finding parking has been noted as one of the major causes of stress in lives of individuals who drive. The traditional method of finding parking by the naked eye has a number of irritating situations. In situations where a driver is walking towards a car or is in the car, the other drivers waiting to find parking often make signs, or whistle or try to do something intending to ask the other whether they are pulling out. Though this kind of asking might help most of the times, it leads to situations which are often inconveniencing to other drivers.

In busy towns and cities, parking management still poses a challenge that keeps growing more complex. The need for efficient parking management systems can't be emphasized enough for such cities. This study thus seeks to provide a solution to the issues above using the latest sensing and telecommunication technology.

1.3 Related work

In an endeavor to solve the parking management issue, different technologies have been developed in various parts of the world and research was conducted to develop efficient parking technologies. The places looked at in this section include cape Town International airport, Cavendish shopping mall in Claremont (Cape Town) and other places in the world such as central London parking bays.

Some Parking System Technologies.

A number of technologies have been deployed in the endeavor to solve the parking problem. While some parking solutions are deployed as stand-alone technologies in some situations, in other situations multiple technologies are combined to achieve a given goal. These technologies include (1) Camera-based sensor technology (2) infrared sensor technology and (3) ultrasonic sensor technology.

1. **Camera-based sensor technology**

Access control systems have been developed and installed in various parking lots. At the airport in Cape Town, an access control point was put in place. It works in such a way that a driver requests for a ticket at the entrance of the parking by pressing a button. When a ticket is issued, the driver is granted access to enter the parking bay [5]. The ticket issued is marked with a code that records the time of entrance to the bay. At the time of exit, the driver has to pay for the time the car spent in the bay. Payment is done by inserting the ticket in another machine which reads the code on the ticket and calculates the amount of time spent. Then a parking fee is charged according to the time spent. Sensors are used at entrance and exit points to keep track of the cars in and out. A display board (variable message sign) is used at the entrance to show whether the bay is full or still has unused slots. This technology helps with the management of payment of use of the parking bays. It also gives users a general idea of the number of available parking slots by displaying a message like “level 1 full”. However, it doesn’t give much help to the driver in finding an exact parking space in the shortest time possible nor allows localization of his car in a given parking spot using for example Radio Frequency Identification (RFID) technology.

2. Infrared sensor technology

In Cavendish shopping mall in Claremont, Cape Town, the access control system is used as well as another system that checks the status of each parking point. The system uses an infrared sensor to detect if a car is parked in a place. A light is fitted above each parking point to show the status. The light operates in the same way as the traffic lights on the road. However the meaning attached to the light colors is slightly different. A red light means that there is a car parked while a green light is used to express a free parking. There is a section designated for parents with kids. This section has an orange light to show that parking is meant for parents with children. This section also uses the red and green lights to show whether the parking spot is free or used. Another section is reserved for the disabled. It is marked with a blue light. This section also uses the red and green light to show the status of the parking. This system is helpful to the drivers once they go through the entrance. It is relatively easy for the driver to look ahead and figure out where there are green lights in the bay. However, this is only possible for the corridor in which the driver is moving through. The driver has to keep looking until they can land on a green light. However, there is still a likely flaw that a driver may see a green light yet find it red and in-use by the time he reaches the spot.

3. **Ultrasonic sensor technology:**

Siemens AG developed a similar system (SIPARK) that uses ultrasound sensors. These are installed above every single parking to determine whether the parking space is occupied or not [12]. This has been used in more than 70,000 parking spaces. The system is being used with better pay points that accept a wide range of payment options ranging from cash to non-cash options such as credit cards.

In a report by M. Crowder and C. Michael Walton, the need for intelligent transportation systems (ITS) is examined across a wide range of places [8]. These include business districts, airports, and transit stations. The report emphasizes that university premises are yet a new pressure zone with an increasing need for a parking solution. This report further details the need for real time provision of information to motorists as a remedy to reduction in traffic congestion.

Among the systems examined in this report are: Saint Paul Advanced Parking Information system (St Paul APIS) in Minnesota, Phoenix Arizona intelligent parking system, Seattle center APIS, and a few others.

In the St Paul APIS [8], static signs are placed in town to direct motorists to where the parking bays are located and the general condition of the road. The diagram depicted by Picture 1 below shows a VMS sign in St Paul town notifying drivers of the road condition. These signs variable message signs(VMS) have the ability to display different messages as set by the traffic control staff.

Figure 1: Variable Message Signs[24]

This very same sign can be used to display the availability of parking in the town.

The variable message signs are used to show the general status of parking bays. Electronic signs show messages such as full, closed and open. This system keeps a count of the vehicles that enter and leave the parking bay. Picture 2 depicts a parking guidance and information system (PGI) at the entrance of a multistoried parking bay.

Figure 2: Display Board at Parking Entrance[25]

This system is designed to minimize disruption of the driver and improve on the drivers' focus to quickly find parking. According to the picture above, the driver saves a lot of time by driving straight to the levels that still have open parking spaces. This system presents some similarities with the system installed at Cape Town airport where drivers are informed of the status of the different levels of the parking.

In Phoenix, Arizona in particular, a phoenix event parking and traffic Information system was designed similar to that in St Paul city. It's an improvement on the system that was in place earlier on called sunburst which was operated by the police to monitor down town events. It uses Variable message signs to disseminate real time information.

In Seattle Washington, an advanced parking information system (APIS) is used as an intelligent parking system of the larger Seattle smart Trek Model deployment initiative designed to improve overall efficiency and performance of the roadways and transportation systems to meet growing demands. Basic architecture of this includes five elements:

(i) Parking facility-monitoring subsystems: monitoring ingress and egress of vehicles at all three parking facilities.

(ii) Parking information signs: provide parking lot occupancy and directional information to drivers at decision points near the Seattle Center.

(iii) Communications subsystem: facilitates communication between the central computer, vehicle detector system and the parking information signs.

(iv) Central computer system: calculates lot occupancies from vehicle detectors, commands signs to show required message, and allows operator intervention.

(v) External interfaces: facilitates communication between the central computer and external systems including Smart Trek VMS.

This system focuses on information dissemination to help travelers with pre-trip information of parking garages [8]. This system was implemented to provide traffic information to the University of Washington Cable TV, commercial radio and local transit authority.

All the systems described above have made an immense contribution to the solution of increasing parking needs in today's ever busy environment. In general, the state of the art of the discussed systems mainly consists of: parking facility equipment, central computer and connections as well as signage. The parking equipment consists of vehicle counters and space monitors on site to monitor activity on site. The central computer then controls VMS by displaying information to help direct the traffic. *The computer can also be configured to send simultaneous messages by radio frequency, dedicated phone line; Ethernet connection to the local radio, television or Internet* [8]. However, a lot remains unaccomplished in the efficiency and dissemination of real time information for motorists.

Synchronization of the Access to parking lots:

The synchronization of the access to parking lots is an important research issue upon which the efficiency of a parking system depends. Rachid,G and Sebastian,B [9] report on work done on synchronization of the access to parking lots through integration of the driving philosophers paradigm into the car park simulation. The driving philosophers paradigm looks at the situation as a problem that involves a set of processes P and a set of resources R. For example P = { P1, P2….. (Pn-2), (Pn-1), pn} and R = {R1, R2, …… (Rn-2), (Rn-1), Rn}. The processes and resources are viewed to be happening in a round about where the cars are the processes and the parking lots are resources. Two types of processes are considered. They include (i) init processes which represent the processes that have not acquired a resource in the round about and (ii) Transit process. The driving philosophers' paradigm seeks to solve the following problems. To ensure that (1) each resource of the round about is owned by at most one process, (2) that no process will starve for too long a period of time, and (3) that no deadlock will occur in the roundabout.

In this approach, an algorithm was developed based on the idea that processes interact in a synchronous way based on a round computation. Each process goes through three (3) phases:

1) Send phase: the process requests resources from other participants.
2) Receive phase: the process receives and stores the received information in order to execute the next phase.
3) Compute phase: the process computes the message to send in the next phase.

Different algorithms were developed using the above mentioned phases. These include:

The serial algorithm: This allows a single process at a time in the round about where the process that has been waiting for the longest time has priority over the other process.

Concur1 algorithm: This gives to processes in transit the priority over the new processes whenever there are two concurrent algorithms for the same resource.

Concur2 algorithm: This gives the new process the priority over the process in transit whenever there are two concurrent requests for the same resource.

The implementation of this system is done by a car park simulator. A car park simulator is an application where cars and car parks interact with each other so that each car finds the closest car park to a target point that has free parking lots. The car park simulation takes place in an environment described by a map. The system has the following components:

i) **The car:** which needs to find parking.

ii) **Car Park:** The parking bay which has a certain number of parking spots. The car park broadcasts its availability and only cars that park in paying car parks receive the information.

iii) **Round about:** This is a new entity that was added which represents a round about in the map. It interacts with the cars that are entering in and exiting the simulation in order to ensure the synchronization between the cars.

iv) **Simulation center:** This doesn't correspond to any physical object in reality. Its main purpose is to display an overview of the system with all cars and the car parks

In terms of communication, the components previously interacted with each other using transmission control protocol (TCP) and multicast channels. Currently communication is done within a limited range. That is to say: Car A receives the message of car B if and only if B is within range of A.

In the above-described work, the processing that takes place is rather tedious. Cars need to be in range in order to communicate. The phases involved also make it hard for quick information access on parking. Thus, there is need to improve on intelligent parking with a system that reduces processing time.

A Smart Management Field Test

Looking further at other systems, a smart management field test is described in [22]. This system involves two real time user interfaces with changeable message signs (CMS)

i) One displays availability information to motorists on the adjacent corridor along the road.

ii) The second one is a centralized intelligent reservation system that permits commuters to check parking availability and reserve a space via telephone, cell phone, internet and personal digital assistant (PDA)

This system integrates traffic count data from entrance to exit to provide accurate up-to-minute counts of the parking availability. 50 parking spaces were used in this test. Those who use the system for en-route reservations call in their license plate numbers via cell phones when they park in the smart parking lot. Users needed to have advanced reservation parking permits and license plate numbers that match those provided to the enforcement personnel in real time via PDA for en-route reservations. This field test further carried out a number of tests on the behaviors and responses of the motorists, such as willingness to pay, feedback on parts of the system and so many others.

1.4 Proposed solution

This study seeks to develop an intelligent parking system that efficiently allows users to locate empty parking bays in the shortest possible time and keeps track of the activity of the parking bays. The system should be able to record, process, store and channel parking activity data to a central gateway where visualization and information dissemination are performed to improve management of the parking bay.

With the help of the Internet and the ever-improving levels of technology, the information dissemination may make the status of parking bays available to mobile users. This would enable drivers who have devices that can connect to the internet to, through GPRS for example to figure out which parking bays are free. More to that, they can even find out the exact parking points that are not being used at a given time and reserve a parking spot. The system should be able to update VMS signs at the parking entrance as well as update other output facilities available for information dissemination.

1.5 Objectives

General objectives

- Improve on the current parking management systems.
- Enable drivers to locate and reserve empty parking bays at the entrance and remotely. This should be provided by a small display panel at the entrance of the parking bay and using mobile devices such as PDAs, smart phones for remote access and reservation. It should show the parking number and the state of the parking. For example 345: used, 346 open, and so on.

Specific Objectives

- Monitor the state of a parking bay. The parking bay may be fully in use or free. This system seeks to constantly monitor what is happening at the bay to identify parking spaces that are available or occupied.
- Collect the time and date of activities happening in a given parking bay. Every time the state of the parking bay changes, the date and time should be captured.
- Gather the data about activity and send it to a central point using a gateway. The date, time and status of the parking spot will all be sent to a base station.

CHAPTER TWO:

SYSTEM DESIGN

2.1 Introduction

The design of the proposed intelligent parking system will involve a number of fields that need to be critically considered if the intelligent parking system is to have useful results. These include, data capturing from the parking spots by sensing activity at the spot using sunSPOT motes, channelling the data through a gateway, displaying, storing, and dissemination of the information.

For data sensing and capturing, our study builds upon the intensity of light captured by sensor technology to detect the presence or absence of a car on a specific parking spot. In our system, a predefined period of low light intensity on a parking spot expresses the presence of a car on the spot while higher light intensity represents the absence of a car on the spot. The Sun SPOT motes are endowed with light sensors which are used in our study to detect light changes at a parking spot. In Figure 3 below a picture of the Sun SPOT mote is shown. [19].
Our design uses a database management system to store the sensed information in a gateway and allows users to query the database for real-time information on the spots. Furthermore, the database should provide information to the visualisation components. The visualisation of the information covers three parts. These include display of information to the parking control staff on a mainframe computer (or simply a personal computer), display of information at the parking entrance to drivers entering the parking, and finally displaying information to the mobile users who will use mobile devices to access the information.

The dissemination of information in this design seeks to use wireless MPLS network as the backbone for information dissemination and transfer.

2.1.1 Sun SPOTs

SunSPOTS are little devices of 41 x 23 x 70 mm which weigh about 54 grams. They are developed by Sun Microsystems. They are shipped from the manufacturer into two versions: (1) free range SPOT and (2) base station SPOT. The base station is a Sun SPOT without battery. It can be connected through USB to a Gateway (computer, PDA, Cell phone) which collects the information sensed by the free range Sun SPOTs and store this information into databases for local or remote processing and appropriate decision making on the parking being monitored. The free range SPOTs are equipped with battery and add-on boards[19]. They are equipped with parking management software to sense what is happening in the parking and communicate wirelessly with the base station which is connected to the gateway. A free range Sun SPOT is shown in Fig. 1 with and without the lid placed. The device's characteristics are presented below.

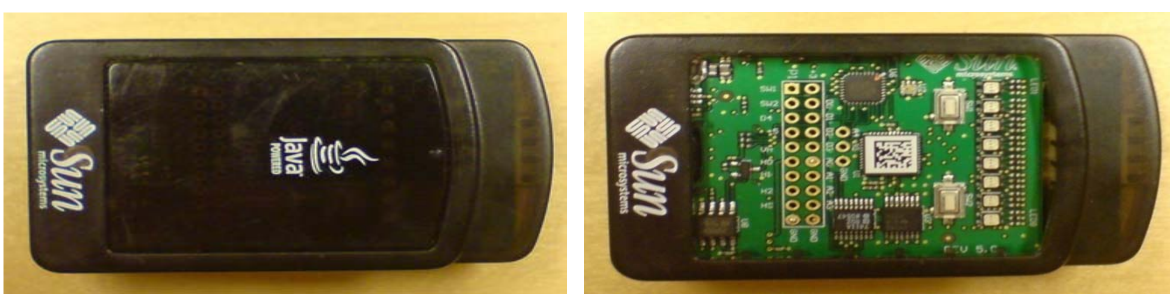

Figure 3: Free range SunSPOT -

The different parts of a Sun SPOT mote are depicted by Fig 2

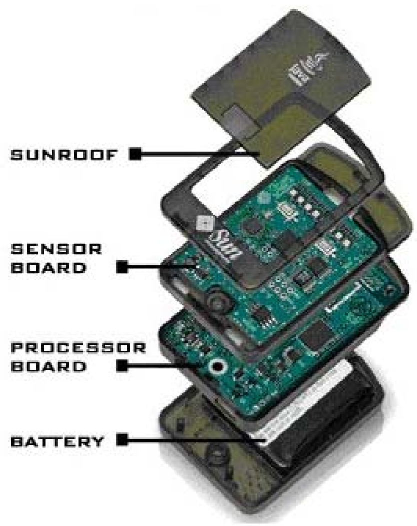

Figure 4: Sun SPOT's different parts

Sun SPOT's Hardware

A Sun SPOT device is built by stacking a Sun SPOT processor board with a sensor board and battery. The main features of a ***Sun SPOT Processor Board*** are

- 180 MHz 32 bit ARM920T core - 512K RAM/4M Flash
- 2.4 GHz IEEE 802.15.4 radio with integrated antenna
- USB interface
- 3.7V rechargeable 720 mAh lithium-ion battery
- 32 uA deep sleep mode

The Sun SPOT's processor board uses a 180 MHz 32 bit ARM92 core with 512K RAM and 4M flash. It is built around the IEEE 802.15.4 radio standard for communication in the 2.4 GHz free ISM band. The battery is rechargeable through its USB interface and goes to deep sleep mode in which it consumes 32 uA. The main features of the ***General Purpose Sensor Board*** are

- 2G/6G 3-axis accelerometer
- Temperature sensor
- Light sensor
- 8 tri-color LEDs
- 6 analog inputs
- 2 momentary switches
- 5 general purpose I/O pins and 4 high current output pins

The sensor board has a 3-axis accelerometer to measure acceleration and generally movement of the device. It also contains an ambient light sensor, eight tricolor LEDs, two push buttons, six analog input pads, four high current high voltage output pads, and five general I/O pads.

Sun SPOT's Software

The Sun SPOT's software is built around the Squawk Virtual Machine. Its main features include

- Fully capable J2ME CLDC 1.1 Java VM with OS functionality
- VM executes directly out of flash memory
- Device drivers written in Java

- Automatic battery management

Sun SPOTs are very flexible to program with very advanced capabilities for such a small device because they are programmable with J2ME CLDC 1.1 Java VM. This small J2ME (Squawk) runs directly on the processor without any other OS (Operating System).

Developer Tools

- Use standard IDEs. e.g. NetBeans, to create Java code.
- Integrates with J2SE applications.
- Sun SPOT wired via USB to a computer acts as a base station.

Sun SPOTs can be used with the standard IDEs like Netbeans to write Java code and deploy it on the device. It can integrate with J2SE applications as well.

2.1.2 Multi Protocol Label Switching.

Multi-protocol Label Switching (MPLS) [10] was proposed by the Internet Engineering Task Force (IETF) to achieve Traffic Engineering (TE) and support Virtual Private Network Networks (VPNs). TE is a traffic management technique allowing the traffic offered to a network to be efficiently routed through a routed or switched network by effecting QoS agreements between the offered traffic and the available resources. MPLS uses a connection-oriented routing model borrowed from the circuit-switching paradigm where the traffic is routed over bandwidth-guaranteed paths referred to as Label Switched Paths (LSPs). These paths are built through label swapping; a packet forwarding model where the route followed by the IP packets is defined by pre-computed labels. These labels are swapped into the core of the network by MPLS routers referred to as Label Switched Routers (LSRs) to define the next-hop to the destination. Multi-protocol Lambda Switching MPLambdaS [23] extends the label swapping paradigm to wavelength-routed networks by assigning labels to wavelengths and switching these labels to setup wavelength-switched paths referred to as Lambda Switched Paths or LambdaSPs. Generalized Multi-protocol Label Switching (GMPLS) [1,2,7,23,20] extends MPLS to allow the migration from per layer to an integrated mode of operation where a common control plane is used to control heterogeneous network elements supporting different routing/switching capabilities [23]. These include IP routers that switch packets, MPLS routers and ATM switches with layer 1 switching capabilities, SDH cross-connects with TDM switching capabilities, OXCs with wavelength switching capabilities and fiber cross-connects that can switch an entire fiber. This mode of operation allows the efficient use of the high bandwidth provided by the optical technology. GMPLS was proposed as an extension to MPLS that makes the status of each link and node of a data/optical network visible to a common network control in order to overcome the immaturity in the current generation control and data plane technologies leading to complex and time-consuming manual network planning and configurations requiring a group of ``layer experts'' to operate and maintain a data/optical network. GMPLS has opened the way for automated operation and management functionalities such as connection creation, connection provision, connection modification and connection deletion. This allows the different layers of an IP stack to be managed by a single network operator.

2.2 Architectural Design

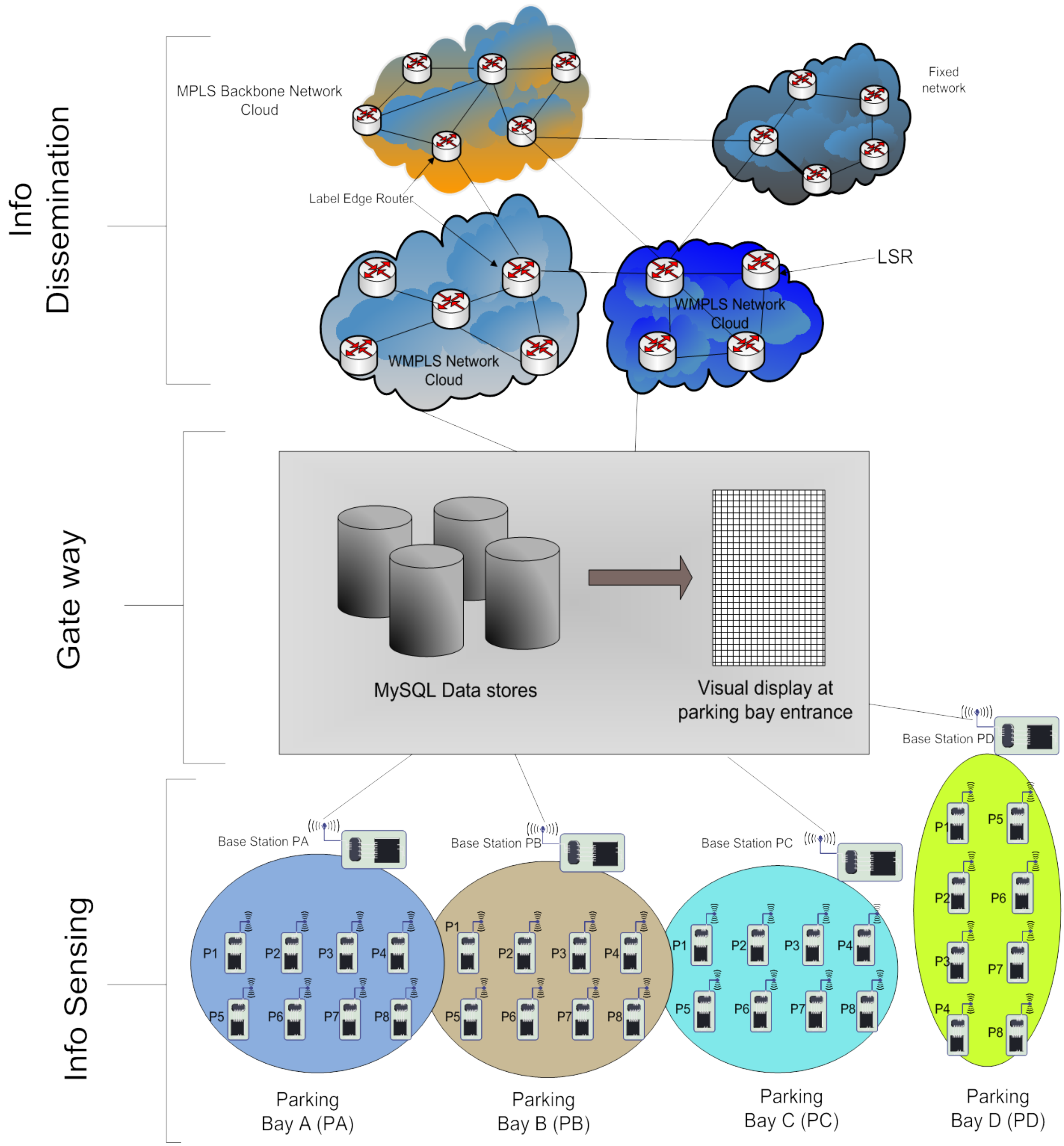

Figure 5: Architectural Design

Figure 3 above depicts a multi-layer system architecture revealing a wireless MPLS network which is layered above islands of wireless sensor networks via a Gateway layer that stores the information collected by the sensor motes into databases. These islands of WSNs are used to sense what is happening in the parking bays by using the light sensor of a SunSPOT mote to detect presence/absence of cars at parking spots. The wireless MPLS network is used to achieve information dissemination for users who either would like to get a parking place for their vehicle in a specific parking bay or reserve a parking spot through the internet or using GPRS enabled cell phones.

2.3 Information dissemination using MPLS

Multiprotocol Label Swapping is a protocol used in connection oriented networks to route traffic. Starting by the Internet backbone, MPLS has dominated the core of several operators' networks and moved into access networks under the Wireless MPLS (WMPLS) denomination. In WMPLS, packets are switched from router to router by use of radio wave. Frequencies of the radio waves in sunSPOT motes can be set and adjusted for each given mote during communication between the device and the workstation. Frequencies can be reused between the base station and the devices.[17]

WMPLS inherits all the functionalities of wire line MPLS and provides an added advantage for mobile users of the network. Devices in the gateway network connect to a base station and communicate to the base station by switching packets basing on the labels attached.

In a mobile network, devices hop from one base station to another. However for this scenario (intelligent parking system), the motes are stationery in the parking area. They use the same mechanism as those that would be mobile. The frequencies of radio communication between the motes and the base station is set prior to deploying them (motes) in the fields. However, the frequencies can be remotely adjusted using communication from a workstation through the base station and finally to the mote whose frequency needs to be changed. The higher the frequency the faster the communication [17]. This also means that there is more bandwidth available at high frequencies since there is less traffic in communication due to faster transmission.

"MPLS is a versatile solution to address the problem to being solved by next generation mobile networks. These include bandwidth, scalability, quality of service (QoS) management, and traffic engineering. WMPL can emerge as an elegant solution to meet effective management of the issues above". [16]

WMPLS provides an added advantage because it can address the issues (mentioned above) which are the same addressed by Internet protocol (IP) and asynchronous transfer mode (ATM). Yet at the same time, WMPL can exist over ATM and frame-relay networks. This is the major reason why WMPLS is said to be the smartest option in providing network efficiency.

2.3.1 Location of WMPLS in OSI model

The transfer of packets in the traditional network remains the same except the destination routing is determined by labels instead of IP addresses. [16] The Diagram below shows the placement of WMPLS.

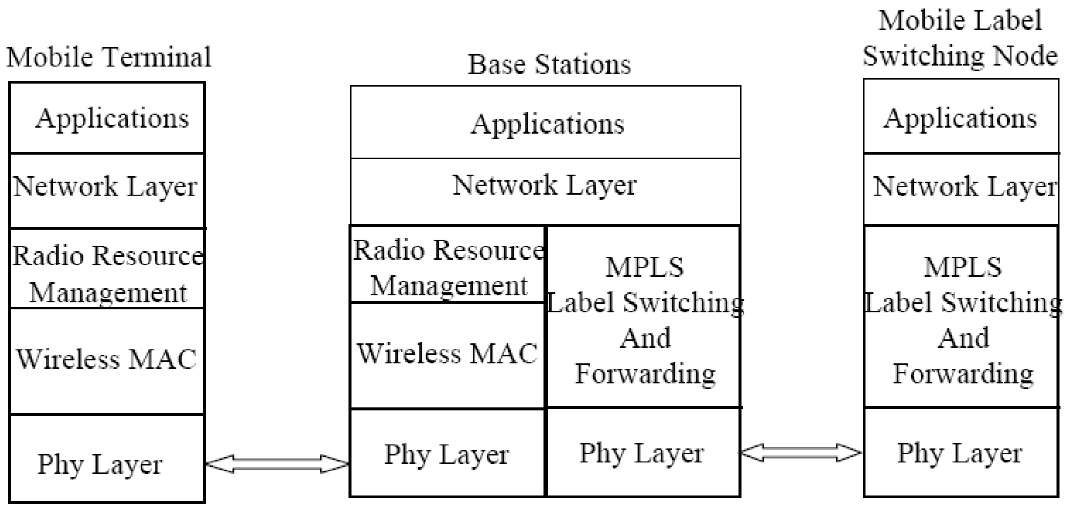

Figure 6: Location of WMPLS in OSI model

2.2.2 How WMPLS handles routing & location management.

There is need to keep track of the position of mobile users in the network. This allows easy forwarding of resources and allows network management of subscribed services. The path between the base station and the mobile device is established basing on location management.

The routing process in the WMPLS is based on the backbone network of MPLS. Packets are switched according to the labels attached to them by the label switched routers (LSR) and edge label switching routers (ELSR).

2.2.3 A brief look at the hand off.

Handoff also called handover refers to the process of transferring packets from one base station to another when a mobile devices changes location in the network.

The process of switching base stations in mobile devices involves a process of handing over of packets from one station to another. A device needs to find a path, establish a label switched router before communication takes place. When the device moves to another location and there is need to switch base station due to different area coverage of the signals, there is need to hand over packets from one base station to another. There is also need to keep the connection without breaking up. While the device is moving from one base station to another, there comes a time when its in a diversity area. This is an area where both base stations have coverage. In such situations, the packets are merged leading to label merging in order to generate a new packet stream for between the mobile device and the new base station. [4]

The design of the intelligent parking system will depend on the frame work of sunspots. A network of sunspots will be configured in such a way that motes in the parking bay will communicate to a base station. The base station will then communicate to a workstation. The motes in the parking bay will communicate to the base station using the radio function of the sunspot. The base station will communicate to the workstation using a universal serial bus (usb) connection.

The data generated by each sunspot will be broadcast to the base station. The base station will listen and receive broadcasted messages. The received messages will then be sent to the workstation/computer.

2.3 Gate way Design

In this scenario, the gateway refers to the means by which sensed information will be channeled and made available to the users of the parking bay. Various computing resources, such as protocols and programs will be integrated together to form the IPS gateway. These resources can be looked at in two categories, that is: information recording and information visualization.

2.3.1 Information recording (MySql Data stores)

The gateway involves data stores which are necessary to hold the information about the parking. This creates the need to design a database specially customized to the needs of the intelligent parking system.

IPS-Database design: In order to successfully develop the required database, entities of the IPS have to be examined, the attributes of those entities identified and the values of the identified attributes need to be agreed upon. A relational model needs to be designed to examine the flow of information, avoid redundancy and present information in a manner that favours all visualization points (that is VMS at the entrance, mobile users as well as display to the control staff).

Entities of the IPS-Database: These are mainly two, namely; parking status and system status.

Attributes of the identified entities:

i) parking status:

Mote Number	Value of X	Value of Y	Time (t)	Status	Alarm

Table 1: Database Table For Parking Status

The values of the attributes are explained as follows:

Mote number will be the position of the mote in the parking. For example Mote number P5 which represents the parking space five. The value of X and Y are light threshold values and will be further discussed in implementation phase. Time (t) will be the time a car spends in the parking area. Status of the parking area will be either used, reserved or empty. The alarm will show if there is abnormal behavior in the parking system. For example if a human being or animal covers the light sensor for a few seconds, the mote will initially detect a change in environment but if this change lasts for a few seconds then definitely its not a vehicle that's caused that change thus the alarm should be triggered.

ii) System Status:

Mote Number	Battery level

Table 2: Database Table Showing Systems Status

The values of the battery level will have values of fully charged, moderately charged, low charge and depleted charge.

Entity Relationship

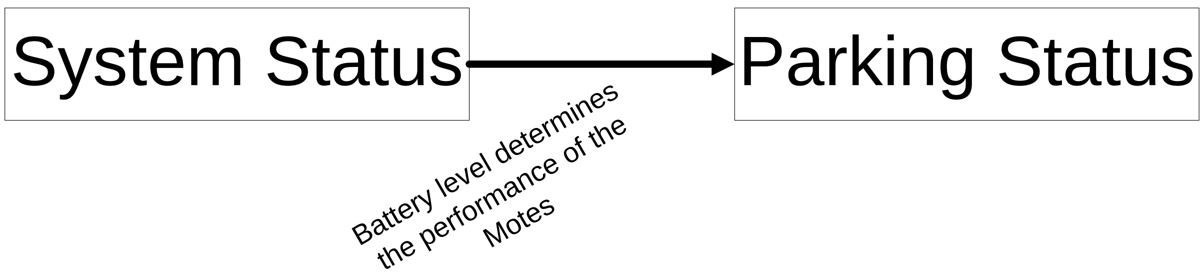

Figure 7: Entity Relationship Diagram

The relationship in the entities showed in the above relational model can be justified as follows: the parking status can only be generated if the battery levels of the motes are powerful enough to sustain the operation of the motes in the parking area. The unique identifying field of these entities is the mote number. The mote number is the best choice of a primary key for each entity. Following the database development procedures, the tables formed in the entities need to be normalized. The normalized form of the database is shown as follows in the table below:

Mote Number	Value of X	Value of Y	Time (t)	Status	Battery level	Alarm

Table 3: Normalized Database Table

This table will be developed in MySQL database to hold the information from the sensing

The code below is an extract of the SQL code that was used to create database and the table that holds the sensed data:

mysql> **CREATE DATABASE** data_store;

mysql> **CREATE TABLE** records (mote_number VARCHAR(15) NOT NULL, value_of_x INT(5) NULL, value_of_y INT(5) NULL, time INT(5) NULL, status VARCHAR(10) NULL, alarm VARCHAR(10) NULL, PRIMARY KEY(mote_number));

Connectivity of the database:

As the data store in the gateway acts as a bridge between information sensing and information dissemination, its important to examine the connections at both sides of the data store. That is the connection between information sensing and the database as well as the connection between the database and information dissemination.

Information sensing to data store connection:

The sunSPOT has the ability to send information to a database. This would happen from the base station. The base station receives data from other motes through radio connection. Then the base station sends the gathered information to a workstation where the database store resides.

Data store to information dissemination connection:

This connection is very important as it's the only way that users can get hold of the information about the parking. The management staff of the parking should be able to query the database on the workstation. This is provided for by SQL which is supported by MySQL.

An interface needs to be built with commands embedded in graphical structure so as to allow easy use of SQL without the user being exposed to the complex technical back end.

The data store should be able to automatically update the parking entrance. The online users should be able to access the database by using the web technology. This will be archived by use of a combination of tools. These include common gateway interface (CGI), HTTP, WMPLS.

The common gateway interface refers to a program that runs on the server. It can be written in various languages. However it’s increasingly being written in Perl programming language because Perl can be used on many platforms. CGI works in the following way; when a user fills a form to query the database, the content of the form is copied to the server. Then the server looks for the program specified in the form and runs the program. The program requests for content from the web page, assigns contents to the variables, and then the program searches the database to find the contents of the variables being queried. The query web page will have names of the variables that are the same names as those of the variables holding data in the database. Once the query is made, the variables are matched and the latest information retrieved. This will be done by use of MySQL commands at the database.

The diagram below shows the connectivity of database to mobile device users /internet users.

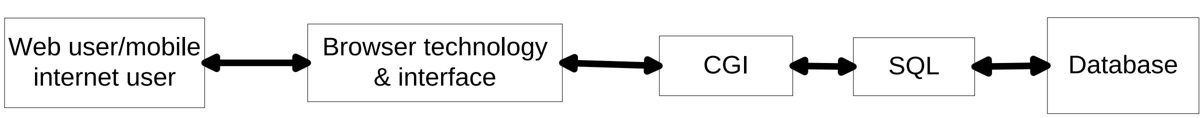

Figure 8: Databse Conectivity

Information visualization:

This will involve means of making the information visible to mobile users, users at the entrance of parking, as well as the managing staff at the parking control system. Display of information to users has mainly been discussed in information recording. However another part of visualizing this information by the staff needs to be looked at. This would include a visualization tool such as Octopus. Octopus is a visualization and control tool for sensor networks. This would mainly provide a visual image of the status of the system. This would help staff in efficiently managing the parking system and keeping it up-to-date. [6] This tool (octopus) will aid the management to locate and replace depleted batteries in the parking.

For the display at the entrance, VMS signs need to be used. These have been used before and proved to be an efficient way to communicate to the drivers.

2.4 Parking spot management using WSN

The system being designed will be implemented in the virtual environment. In this case, the only part of the system that will be real is the computer. Thus the motes in the parking area as well as the base station will be virtual. The data flow from each spot thru the radio, to the base station and finally to the workstation will all happen on the computer. This will be a direct simulation of what exactly would happen when this system is implemented in real life.

The diagram below shows the architectural design of the intelligent parking system.

2.5 Detailed Design of an intelligent parking system.

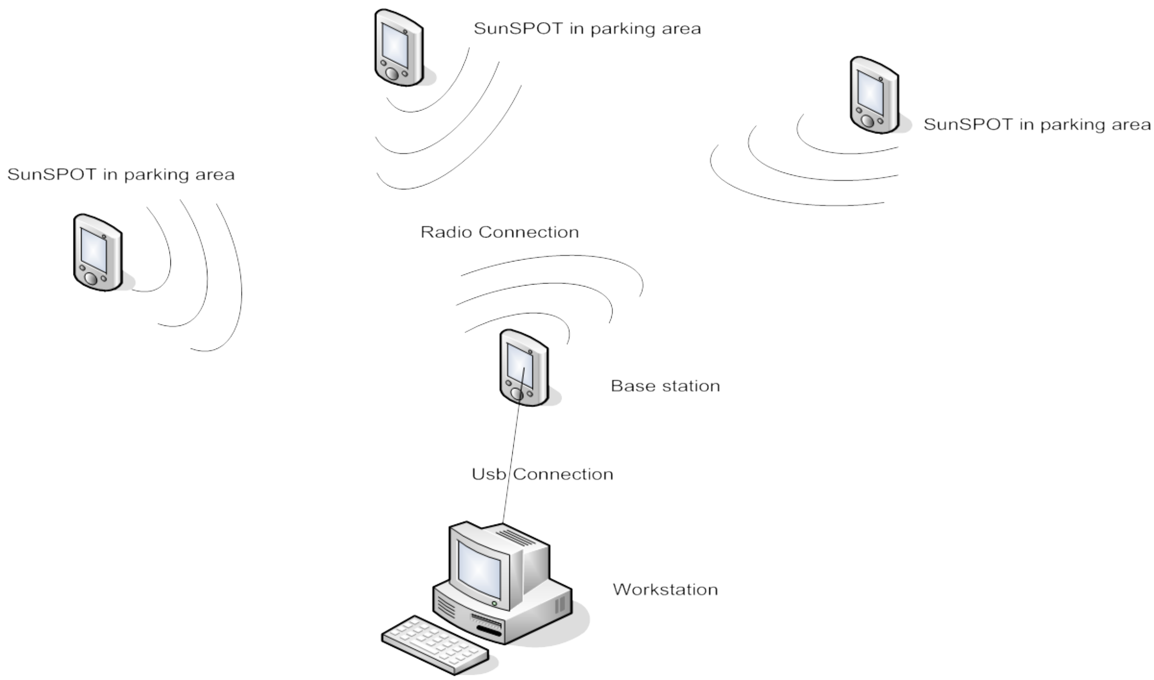

Figure 9: Design of an intelligent parking system

CHAPTER THREE:
IMPLEMENTATION

3.1 introduction

The sunspots were deployed with build files or jar files. The files were run using a java virtual machine called squawk. This virtual machine allowed processing of data using the deployed files. The jar files were developed in net beans, compiled and deployed onto the virtual sunspot. The sun SPOT development kit was used to store the jar files that were deployed on the sun spots. The diagram below shows the frame work of the development environment.

3.2 Development environment

The diagram below shows the various parts of the development environment that were used in the implementation of the system described in the design stage.

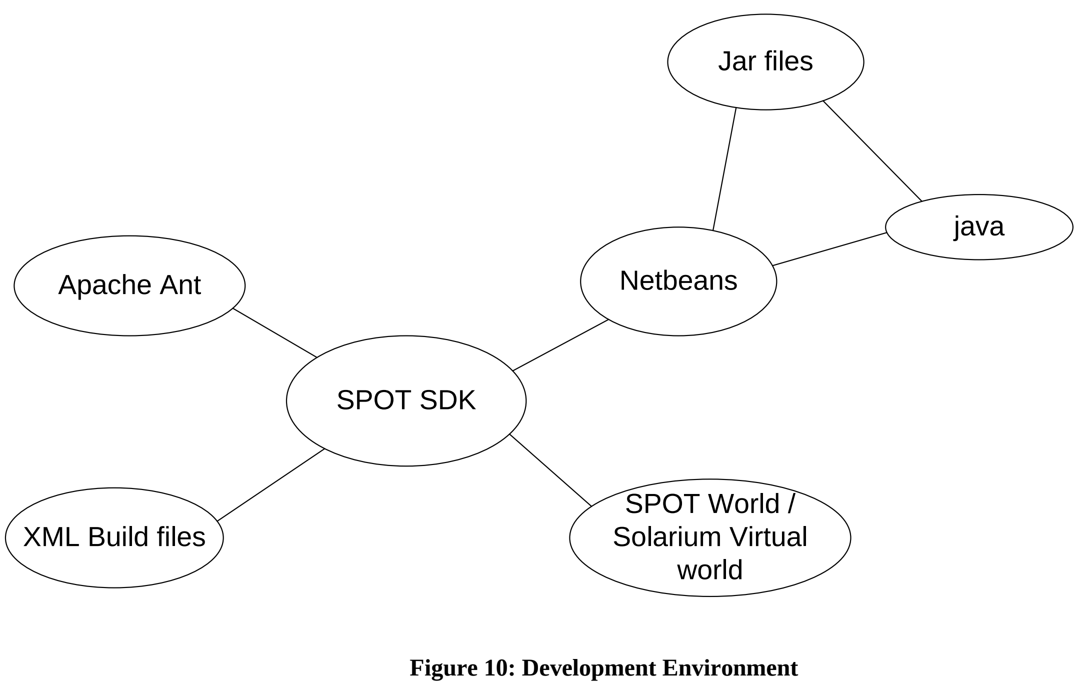

Figure 10: Development Environment

Apache Ant enabled command line use of the sunSPOT. Deploying of files on the sunspot could be done either in command line or by using graphical interface.

3.3 Algorithm implemented by the motes.

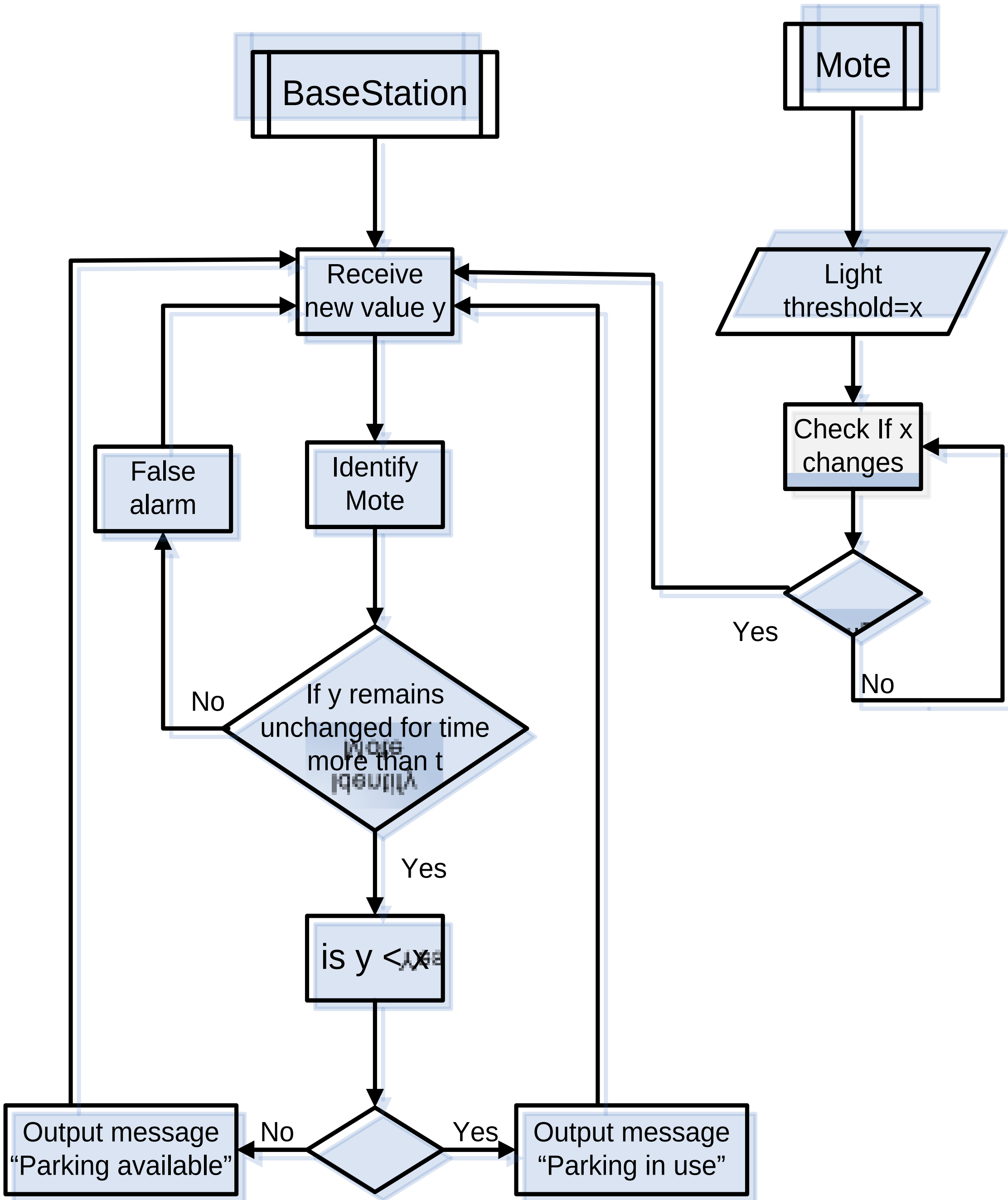

Figure 11: Algorithm for Intelligent Parking

The algorithm above was executed as follows.

Initially, light threshold of the light sensor is set to a value x. The mote then:

1) Listens for any changes in the threshold.
2) If threshold changes to a value less than x, let the new value be Y. If value = y for less than five minute, assume that some moving object was at the parking place but not necessarily a car.
3) If threshold changes to a value less than X and keeps at new value for more than five minutes,then that parking slot has been occupied by a car.
4) If step three is true, proceed to step five, Else ignore activity, take no action, return to(1).
5) Record time t when the change happened.
6) Record date dd/mm/yyyy when the change happened.
7) System.out.print (the parking slot was occupied at time t on date dd/mm/yyyy)
8) Listen for changes in threshold again.
9) If threshold = X ISTRUE (that is to say that the threshold has returned to the initial value)
10) Record time t and date dd/mm/yy
11) System.out.print (car left at time t, on date dd/mm/yy)
12) Return to step (1)

Issues that arise with the above algorithm.

- The algorithm works during day time. If it gets dark, the system could end up recording that there is a car in parking yet its just night time. The parking bay would then need to be carefully lit to maintain the normal light threshold when the parking spaces are not in use.
- The threshold value may become less than the initial value and stay in that state for more than five minutes but it doesn't necessarily mean that there is a car. It could be something else covering the sport. For example a leaf that fell from the tree to the top of the spot.

3.4 Development Requirements.

In this project, a number of requirements were be needed for successful development, implementation and deployment. These were categorized into software, hardware, inputs/outputs as well as the expected size of the system.

3.4.1 Software

In the category of software requirements, the following needed to be in places:.

a) Java development kit (JDK): This environment provided the tools for developing the software that will run on the sunspot. In this project, JDK version 6 will be used.

b) Java run time environment (JRE): This environment would allow the compiling and testing of the developed software before deploying it on the sunSPOTs. In this project, Java 2 standard edition (J2SE) will be used.

c) Net beans IDE: This provided an environment with a graphical user interface, code editor, compiler and a debugger. This provided the java programming tools in a centralized location and allowed enhanced development.

d) Apache Ant: This was installed to allow the use of a command line prompt.

The above mentioned software was a prerequisite for installing the software development kit of the sun SPOT.

e) Sun SPOT SDK: This environment was used in the development of software that would be run on the virtual machine on the Sun SPOT. The squawk virtual machine differs from the normal virtual machine of the computer.

f) The emulator (solarium): This was used to provide the virtual environment. It was generated form the sun SPOT SDK.

3.4.2 Hardware

In the real world, the hardware required was Sun SPOT motes, a base station and a computer. However, in the scope of this project, the only hardware needed was a computer (desktop /laptop). The computer specifications needed to be at least:

a) Pentium iv

b) 1GB RAM

c) Processor speed 3.00 GHz

d) At least 20 GB of hard drive space

3.4.3 Inputs & Outputs

Inputs

The system read light values, generated the date and time when the value was noted and determined the status of the parking bay depending on the light value. Therefore, the inputs of the system were light value, date, and status of the parking bay.

Outputs

The system generated a message which was broadcasted to the base station. The base station then sent the information onto the workstation.

3.4.4 Size of system

The size of the system depended on how big the parking bay was. For each sunSPOT, a jar file was be uploaded which contained the midlets needed to perform the functions of the intelligent parking system.

3.5 Project scope

The system can operate on both windows and Linux platforms. The sunspots run on a java virtual machine and execute java jar files in order to archive functionality. The jar files can be deployed from any workstation having a java development environment. This enables the sunspots to operate with operating systems such as windows and Linux .

The study mainly uses the light sensor of the sunspot to monitor the activity of the parking bays. The sunspot has other sensors, namely: temperature and movement sensor.

This project will run in a virtual environment as a simulation of the real world. This will be achieved using the emulator of the sunspot which provides the virtual environment. The emulator will provide an imitation of what will happen in the real world when the system is implemented.

Figure 12 below shows a demonstration of a single mote in action at a stage of monitoring and sensing data. The change in light threshold is recorded and the meaning of that change is processed and displayed.

Diagram of a SunSPOT sensing change in light threshold

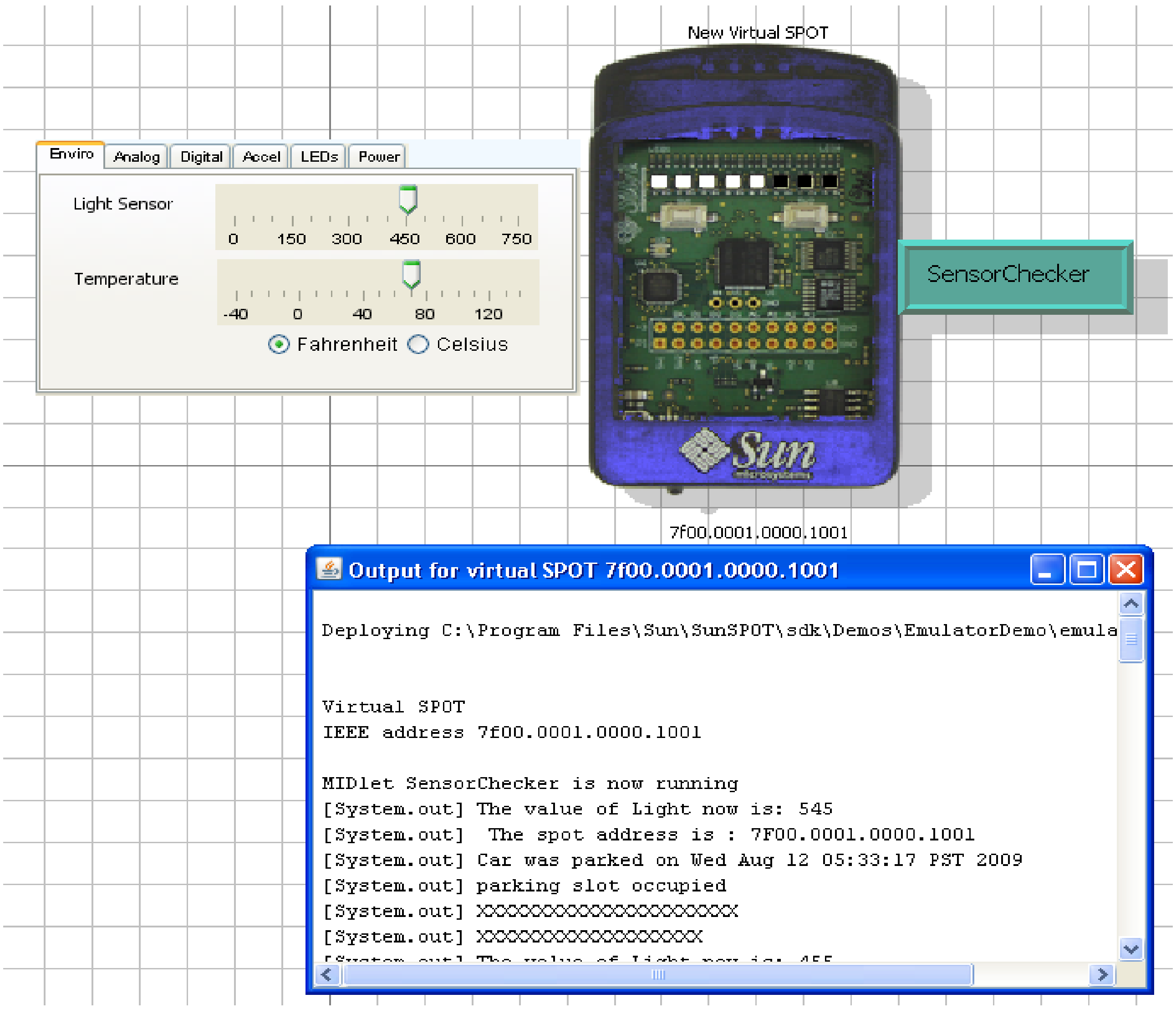

Figure 12: SunSPOT detecting used parking space.

In the diagram above, a virtual mote is shown with three main parts.

I. The sensor panel: this has sensor control knobs and switches used to adjust the light values.

II. LED lights: As seen in the diagram, there are white lights lined up near the top of the mote. To the right of those light are three black little squares. These are LEDs whose light has gone out due to the decrease of light as detected by the light sensor.

III. The output frame: Below the virtual mote is a window showing the output of the mote. The output shows details of the midlet deployed on the mote. Its then shows the status of that midlet. For example, after the midlet was deployed, it showed that sensorchecker was running. On the other hand, the virtual mote displays a green label for each midlet that is running at the time. As seen in the picture, a little green label shows SensorChecker. This is the stage when the mote is ready for monitoring the changes in light. In the output, below the line that shows that the sensorchecker is running, the mote gives outputs caused by the change in light value. As required of the mote, the new light value was captured and the addresss of the mote retrieved. The system then determines the state of the parking, and records the date when that happened. According to the scenario in the above figure, the light value changed to a value below the set threshold. Threshold was set to be 600 in this situation.

When the light value increases, the mote senses the change and a number of processes take place.

The code below gives a brief insight on the method used by the light sensor to monitor the environment. An “if statement” is used to execute the process when the light value goes below the set threshold. An “elseif statement” is joined to the if statement to execute the process when the light value returns to the set threshold or higher. The squawk virtual machine on the sunSPOT mote uses classes and methods that were built specially for the hardware peripherals on the mote circuit board. For example, the address of the SPOT is retrieved by use of the radio policy manager and the IEEE address. These are both classes unique to the mote. However, the Squawk VM uses some of the common classes and methods used by the Java virtual machine on computers. Such classes include the date and the output methods, that is system.out.print. It should be noted at this point that the functionality of the squawk VM combines a variety of resources.

```
public void thresholdExceeded(ILightSensor light, int val)
    {
        if (val< 600)
        {
            Date parkedDate = new Date();

        //getting the spot address
     IRadioPolicyManager addressInstance = Spot.getInstance().getRadioPolicyManager();
    IEEEAddress spotAddress = new IEEEAddress (addressInstance.getIEEEAddress());
         String id = spotAddress.asDottedHex();

            System.out.println("The value of Light now is: " + val);
            System.out.println(" The spot address is : " + id);
            System.out.println("Car was parked on " + parkedDate);
            System.out.println("parking slot occupied");
            System.out.println("XXXXXXXXXXXXXXXXXXXXXX"        +        "\n"
+"XXXXXXXXXXXXXXXXXXX");
        }
        else if (val > 600)
        {
            Date parkedDate = new Date();
            System.out.println("The value of Light now is: " + val);
            System.out.println("Car left parking slot on " + parkedDate);
            System.out.println("parking slot free");
            System.out.println("XXXXXXXXXXXXXXXXXXXXXX");
```

Below is figure 13 showing the same mote as seen in figure 12 but this time its the scenario when the car moves away from the parking. The Leds are seen to increase in number, which is a sign of increased light detected by the light sensor. The output changes showing the new light value and the status of the parking space. The output frame keeps listing the changes as they happen. The previous output is shown as well as the latest output.

Diagram showing a virtual sunSPOT detecting free parking space

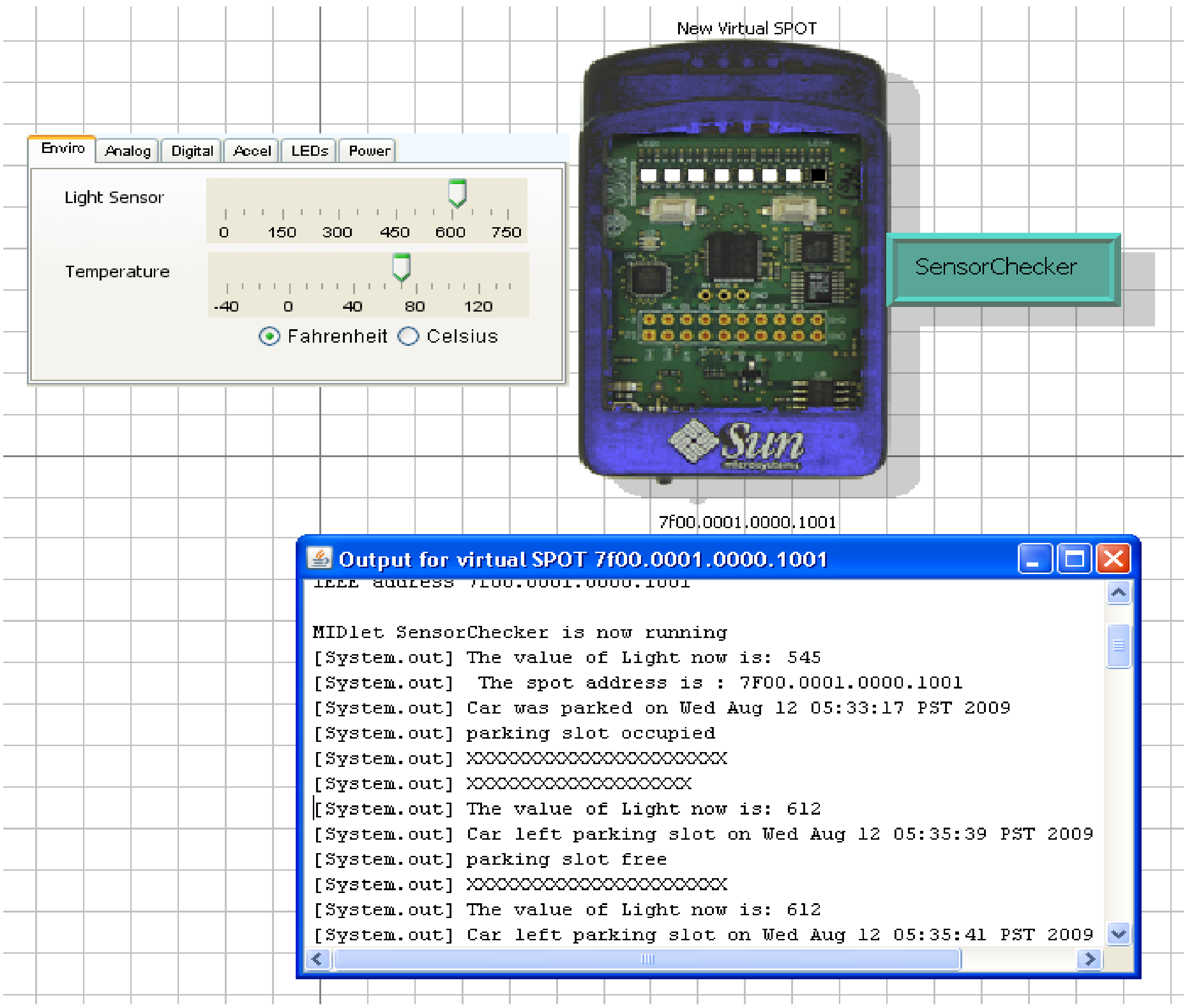

Figure 13: SunSPOT detecting free parking space

The new light value is recorded and the status of the parking determined.CHAPTER FOUR:

DISCUSSIONS & RECOMMENDATIONS

4.1 System evaluation

RESULTS

A once off observation was made for 50 parking spaces showed that the activity at the parking spaces was normal most of the time. A few false alarms were triggered while some parking spaces remained unoccupied.

A table was used to tabulate the observed results of the parking bay. The Motes were numbered as parking spots P1 to P50. The threshold value X of the light was kept constant. The new light value Y was recorded and a time t counted for how long the light value was Y. The battery level was recorded for each mote at the time of observation.

The status of each parking spot was determined through the criteria below:

Status of a given parking space = IF ($(Y < X)$ AND ($t > 5$),"then the spot is in use", "otherwise its free")

An alarm system was configured to check any malfunctions of the system. The operation of the alarm system was configured by use of the equation below:

Alarm = IF (($Y \neq X$) AND ($t < 5$) ,"Trigger Alarm", "otherwise the situation is Normal")

NB: the situation where X=Y is when no activity took place at the parking space.

Tabular results of the activities in parking Bay

Motes	Value of X	Value of Y	Time t (min)	Status	Battrey Level	Alarm
P1	550	200	0.45	free		Trigger Alarm
P2	550	108	60.00	in use		Normal
P3	550	530	140.00	in use		Normal
P4	550	268	49.00	in use		Normal
P5	550	167	180.00	in use		Normal
P6	550	359	500.00	in use		Normal
P7	550	152	0.30	free		Trigger Alarm
P8	550	200	65.00	in use		Normal
P9	550	0	3.00	free		Trigger Alarm
P10	550	315	68.00	in use		Normal
P11	550	200	21600	in use		Normal
P12	550	108	7200.00	in use		Normal
P13	550	550	0.00	free		Normal
P14	550	268	69.35	in use		Normal
P15	550	167	120.00	in use		Normal
P16	550	359	205.33	in use		Normal
P17	550	152	378.39	in use		Normal
P18	550	550	0.00	free		Normal
P19	550	0	106.84	in use		Normal
P20	550	315	0.70	free		Trigger Alarm
P21	550	206	106.88	in use		Normal
P22	550	203	106.93	in use		Normal
P23	550	550	0.00	free		Normal
P24	550	301	20.00	in use		Normal
P25	550	550	0.00	free		Normal
P26	550	308	5900.00	in use		Normal

P27	550	312	162.00	in use		Normal
P28	550	550	0.00	free		Normal
P29	550	319	378.00	in use		Normal
P30	550	322	269.28	in use		Normal
P31	550	326	106.84	in use		Normal
P32	550	550	0.00	free		Normal
P33	550	333	191.15	in use		Normal
P34	550	337	238.66	in use		Normal
P35	550	340	133.00	in use		Normal
P36	550	550	0.00	free		Normal
P37	550	347	17.79	in use		Normal
P38	550	351	11520.00	in use		Normal
P39	550	355	150.76	in use		Normal
P40	550	358	191.37	in use		Normal
P41	550	362	138.55	in use		Normal
P42	550	365	88.56	in use		Normal
P43	550	550	0.00	free		Normal
P44	550	373	87.53	in use		Normal
P45	550	376	53.47	in use		Normal
P46	550	380	29.91	in use		Normal
P47	550	383	57.95	in use		Normal
P48	550	550	0.00	free		Normal
P49	550	550	0.00	free		Normal
P50	550	394	12.79	in use		Normal

Table 4: Results

The results in the table were further analyzed in a graphical manner.

i) Parking status

The status of the motes was examined where a graph of number of motes vs status was plotted. The number of motes in each status were grouped together to have an understanding of the general performance of the parking bay. As shown in the graph below, there were few cases of false alarms. Out of the 50 motes, 46 Motes behaved in a normal condition as shown by the blue bar. This normal behavior was seen to greatly depend on the system status. If the battery levels are at least above 50% charge, then there are minimal system malfunctions. The reasons why low battery levels would cause malfunctions include: (1) system failiure due to inability to perform normal functions.and (incorrect readings due to low power.)

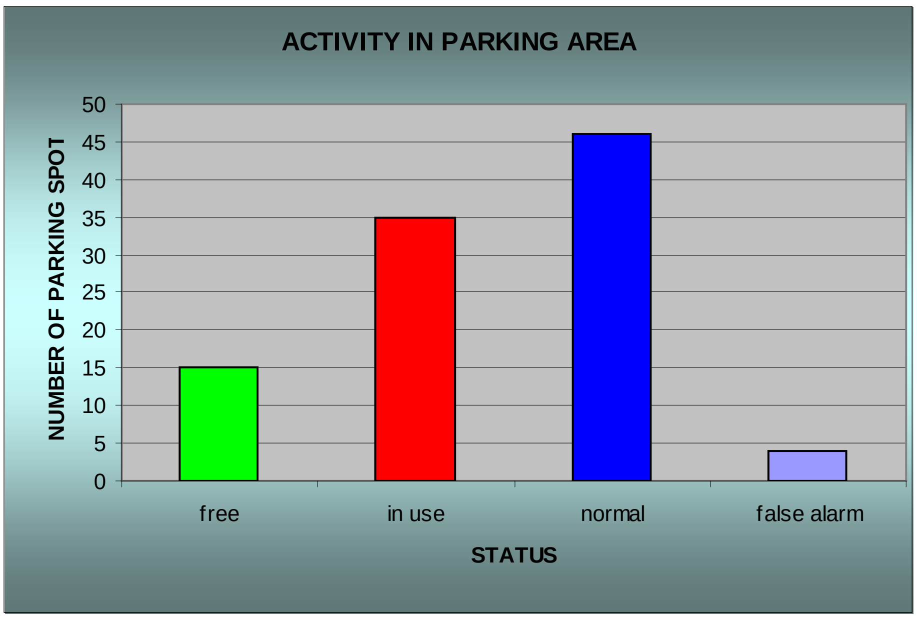

Figure 14: Graph of Results

ii) System status

The status of each mote is also checked. This is mainly the battery level of the motes. It was vital to know the battery level of the mote so that one can distinguish between system failure due to low battery and other system malfunction.

The battery level has categories in percentage of the charge. These are as follows:

Green: 80 – 100 % fully charged

Orange: 50 – 80 % moderately charged

Yellow: 20 – 50 % low charged

Red: 0 – 20 % depleted charged

The Pie chart below represents battery levels mentioned above.

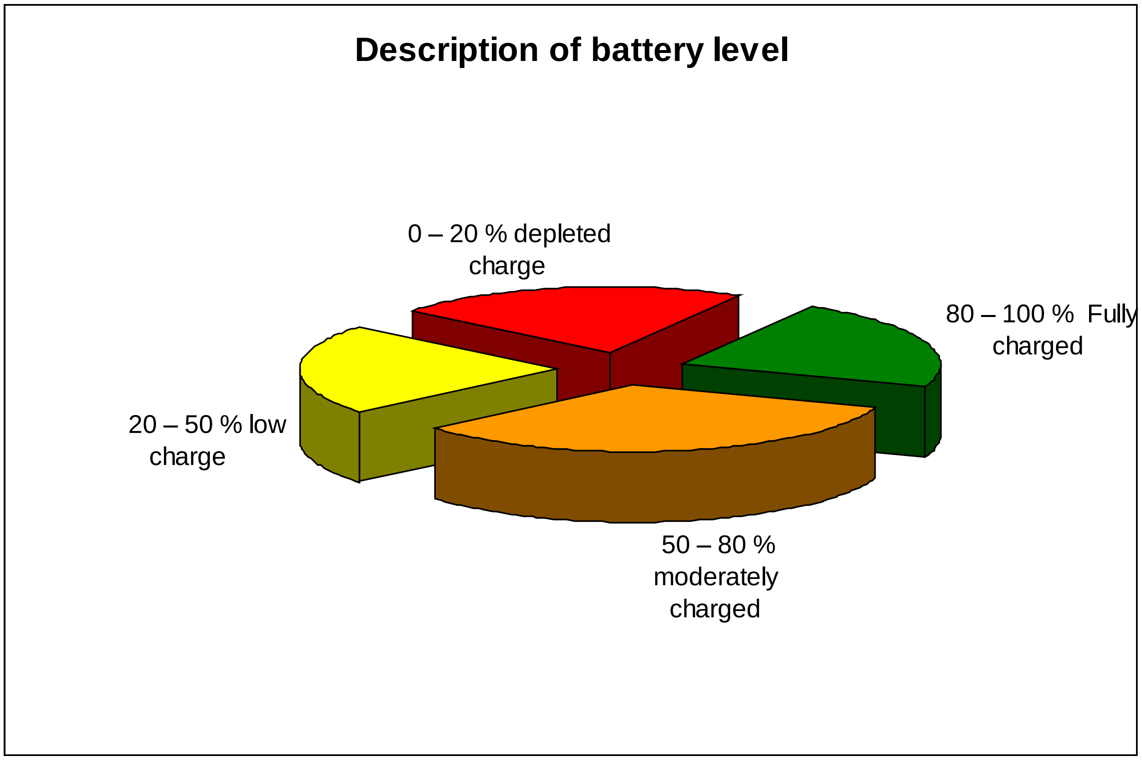

Figure 15: Description of Battery level

4.2 Achievements

A new solution to the problem of parking location was developed. This technology provides processing abilities using the sunspot circuit board which operates with a java virtual machine indirectly performing as an operating system on its own.

The system monitors a parking space, records the date and time of instances whenever there is an activity in the parking. Data is collected on a centralized location. All data is sent to a base station which the transfers information to the workstation by a USB connection.

The system has the option of being configured to provide online data to motorists. However, this option was not included in the scope of this research.

4.3 limitations of the system

The system can't distinguish between day and night. Thus it is likely to behave as if there is a car parked in the area when it's dark. This limitation can be overcome by having well measured lighting in the parking areas.

4.4 Problems encountered

The system required at least 1gb of RAM and a good internet connection. These were not readily available as my computer had only 512 MB of RAM. The internet connection was always interrupted by the university internet control system from the department of Information Communications Technology services.

This is a new technology in the field of parking management. Therefore I didn't have many people to consult with besides my supervisor. Time and again, I had to link up with the developers on line through blogs as well as consult developers in Italy.

The Squawk virtual machine resident on the sun SPOT motes presented different methods and functionality. This made a huge difference from the normal java virtual machine. Classes were accessed through categories which are classified according to the devices of the sunspots. This required further study and skill improvement in order to maximize the capabilities of the sun SPOT.

4.5 Recommendations & way forward

The system is ideal for developing countries as well as developed countries. This is true due to the wireless MPLS ability and the radio function of the sunspot. This removes the need for wires which have to be hidden in the ceiling, walls or in floor of the parking area.

One sunspot can be modified to cover three parking spaces. The sunspot sensors are made in such a way that they can be plugged in and out of the circuit board of the sunspot. This means that the circuit board can be modified to have three plug in ports for the light sensors. Thus a mote can be placed at a given parking space and one sensor elongates to the left while another elongates to the right. These sensors can then independently monitor activities of three places using one mote which then sends data to the base station.

With the right configuration, the system can be used to avail parking bay information to television, radio stations, cell phones, PDAs and many other devices. This is all possible due to the ability to provide information online.

CHAPTER FIVE:
CONCLUSIONS

In conclusion, this study suggests that there is a lot of potential in the use of a SunSPOT device which should be further examined as far as intelligent parking is concerned. There is a likelihood of developing the next generation parking systems that are highly efficient. This is all possible due to the robust functionality of the sunspot mote. It has a processing ability which can accommodate a number of functionalities. The scope of how much the sunspot can do remains a question of creativity and invention.

5.1 Bandwidth related issues

The information disseminated in this study is basically plain text. This is considered as very small size of information that should be easily disseminated. However, the aspect of data traffic in the network of the intelligent parking system can't be left out. As multiple users send requests for available parking, there is likely to be conjestion due to the messages moving back (responses) and forth (requests) in the dissemination gateway. In large parking bays, traffic may become a threat leading to system malfunction not to mention failure to provide users with the required information. In a perfect situation, users would send a request and receive feedback. That would be the end of the query. However, in the real life, users may make more than one request. Their needs may change and if there are not satisfied with the feedback, then they may make more queries. Therefore, for the intelligent parking system to efficiently deliver the required services, the bandwidth dedicated to the system needs to be sufficient. This study thus suggests a further investigation on the management of bandwidth for intelligent parking systems.

5.2 SunSPOT improvement:

Furthermore, there is need to research on how cameras can be embedded in the sunSPOT circuit board. This possibility could bring to birth a powerful system yet functioning on a small robust mote. This can provide a more diverse system that is able to monitor cars and provide surveillance of what's happening in the parking area. This kind of surveillance would further indentify cars parked in the bays by recording their registration plates. At the same time, this surveillance would provide video coverage in case of any damage caused to vehicles in the parking. Vehicle security threats such as vandalism, theft and others can then be checked. If this can be archived, a lot of costs will be cut. This kind of improvement would then provide an alternative to the present closed circuit television (CCTV) systems used today. The systems that operate with separate multiple parts such as cameras, sensors, would be combined into one. The centralizing of information and processing power would thus create a whole system that can be accessed from a common point.

5.3 User manuals:

The usability of this system can be rated easy given today's technology driven world where users adapt quickly to the trends. However, there would still be need to create user manuals for the system suggested in this study. They may not be huge manuals but could be as small as a pocket booklet with hints on how to use the system. Furthermore, there is need to choose agreeable word formarts. The study suggests that the information disseminated would need to have a formart to allow consistency and easily be understood by the drivers. There can be alternative words used to mean the same thing. Nevertheless, the question remains that which of the alternatives would be the best to provide good human interaction with the system. For example, the following words can be used; “P203-available”. But other words could also be used like “spot 203 open” These are issues can be best resolved by putting other factors into consideration such as human computer interaction. Therefore, there would be need for standards to be set for what’s easily read and understood.

5.4 Synchronizing parking space data and fee charging systems

Further study needs to be done on how the fee charging system can be incorporated in the parking space monitoring system. The combination of these two systems would enable the driver to electronically pay and book for a parking spot before embarking on the the journey to their destination. The current system use a ticket which is issued to the driver at the time of entrance to the parking bay. It would be of much convenience for the ticket to be combined with parking spot details.

In a nutshell, ground work has been done in this study. Given the fact that this is the first attempt to use the sunSPOT motes in intelligent parking systems, a lot needs to be examined on how better functionality can be achieved. With today’s drivers in busy environments, the ability to access parking information remotely will solve challenges faced. Lastly, This kind of technology is more likely to benefit developing countries but not excluding developed countries.

REFERENCES

[1] Ben Yoo, S.J. Optical-label Switching, MPLS, MPLambdaS, and GMPLS. Optical Networks Magazine. May/June (2003),17-31.

[2] Berger, L. Generalized Multiprotocol Label Switching (GMPLS) Signalling Functional Description, Request for Comments. Available from: http://www.ietf.org/rfc/rfc3471.txt, (2003); accessed 17 August 2009.

[3] Bijan, J., Rajiv, P., and Esmeal, D. Label Switched packet transfer for wireless cellular networks. In Wireless Communications and Networking Conference Available from: http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=904756 (2000. WCNC. 2000 IEEE) George Mason University (2000); accessed 10 March 2009.

[4] Jatuporn, C., Udomporn, S., and Satien, T. Smart parking: An Application of Optical Wireless Sensor Network. Available from: http://www2.computer.org/portal/web/csdl/doi/10.1109/SAINT-W.2007.98 (2007); accessed 7 March 2009.

[5] Katanga Executive Parking. How to use Katanga Valet Parking. Available from: http://www.valet.co.za/template.php?page=home (n.d.); accessed 14 August 2009.

[6] Mthokozisi, M., and Sa,L, S. Wireless Sensor Network Repairing, Honours project, University of Cape Town, 2008.

[7] Mannie, E., Generalized Multiprotocol Label Switching (GMPLS) Architecture, Request for Comments. Available from: http://www.ietf.org/rfc/rfc3945.txt, October (2004); accessed 17 August 2009.

[8] Crowder, M., and Walton, M. Developing an Intelligent Parking System for the University of Texas at Austin Research Report SWUTC/03/167229-1 Available from: http://swutc.tamu.edu/publications/technicalreports/167229-1.pdf (2003); accessed 10 June 2009.

[9] Rachid, G., and Sebastien, B. Integration of the Driving Philosophers into the Car Park Simulation (January 2005); accessed 7 July 2009.

[10] Rosen, E.C, Viswanathan,A., and Callon, R. ``Multiprotocol Label Switching Architecture", Request for Comments. Available from: http://www.ietf.org/rfc/rfc3031.txt, (2001); accessed 17 August 2009.

[11] Sabella, R. et al. Strategy for Dynamic Routing and Grooming of Data Flows into Lightpaths in New Generation Network Based on the GMPLS Paradigm. Photonic Network Communications. 7,2(2004), 131-144.

[12] Siemens AG 2002-2008. Car Park Management. Available from: http://www.industry.siemens.de/traffic/EN/products/urban/parkloesungen.htm; accessed 10 June 2009.

[13] Simply Extras Limited. Cavendish Central London. Available from: http://www.gosimply.com/airport-parking/central-london/car-parks/cavendish-square-london-parking (2009); accessed 14 August 2009.

[14] SPG Media Limited. Automated Parking System T5 Multi-Storey Car Park, Heathrow, London, United Kingdom. Available from: http://www.roadtraffic-technology.com/projects/siemensparkingsystem/ (2009); accessed 15 May 2009.

[15] Subramanain, V., and Subramanain G. An architecture of MPLS implementation in wireless networks. In Proceedings of EIT, IEEE conference, Michigan. Available from; http://www.secs.oakland.edu/~ganesan/network/MPLSforwirelessNetworks.pdf (2001); accessed 20 August 2008.

[16] Subramanain, V., and Subramanain, G. QoS Implementation For MPLS Based Wireless Networks in ASEE Conference . Available from: http://www.secs.oakland.edu/~ganesan/network/QoSImplementationForMPLSBased wireless%20Networks.pdf (April 2002), Oakland University, Rochester, Michigan; accessed 4 March 2009.

[17] Sun Microsystems. Sun™ Sun Small Programmable Object Technology Developers Guide. California, Sun Microsystems, (2007); accessed 10 October 2009.

[18] Sun Microsystems. Sun™ Small Programmable Object Technology (Sun SPOT) Kit Installation Instructions Release 2.0 Sun Labs (2007); accessed 30 June 2009.

[19] Sun Microsystems. Sun™ *SPOT Owner's Manual* Blue Release 4.0. Santa Clara, California, Sun Microsystems, (2008); accessed 20 January 2009.

[20] Sun Microsystems. *Sun™ Small Programmable Object Technology (Sun SPOT)Theory of Operation.* Available from: http://www.sunspotworld.com/docs/Purple/SunSPOT-TheoryOfOperation.pdf (2007), accessed 16 January 2009.

[21] Shaheen, S.., Rodier, C., and Eaken, A.. Smart Parking Management Field Test:A Bay Area Rapid Transit (BART) District Parking Demonstration. (2005); accessed 19 May 2009.

[22] Vetter, T., MacLean, D., and Creech, H. Sustainable Development: Theme Proposal for IGF New Delhi Available from: http://www.iisd.org/pdf/2008/backgrounder_igf_new_delhi.pdf (2008); accessed 15 July 2009.

[23] Vigoureux, M. et al. Multilayer Traffic Engineering for GMPLS-Enabled Networks. IEEE Communications Magazine 43,7(2005), 44-50.

[24] Wikimedia Commons. Changeable Message Sign. Available from: http://en.wikipedia.org/ (2006); accessed 30 June 2009.

[25] Wikimedia Commons. Multi Storey PGI VMS. Available from: http://en.wikipedia.org/ (2007); accessed 30 June 2009.