Bluetooth Wireless Monitoring, Managing and Control for Inter Vehicle in Vehicular Ad-Hoc Networks

Helia Mamdouhi, Sabira Khatun and Javad Zarrin
Department of Computer and Communication System Engineering,
Faculty of Engineering, University Putra Malaysia, Malaysia

Abstract: Problem statement: The car users expect more and more accessories available in their cars, but the accessories available needed manage by driver manually and not properly manage by smart system. All these accessories are able to control by user manually using different and standalone controllers. Besides, the controller itself uses RF technology which is not existed in mobile devices. So there is lack of a comprehensive and integrated system to manage, control and monitor all the accessories inside the vehicle by using a personal mobile phone. Design and development of an integrated system to manage and control all kind of inter vehicle accessories, improving the efficiency and functionality of inter vehicle communications for the car users. **Approach:** The proposed system was based on Microcontroller, Bluetooth and Java technology and in order to achieve the idea of an intelligence car with ability to uses personal mobile hand phone as a remote interface. Development strategies for this innovation are includes two phases: (1) java based application platform-designed and developed for smart phones and PDAs (2) hardware design and implementation of the receiver side-compatible smart system to managing and interconnection between all inside accessories based on monitoring and controlling mechanisms by Bluetooth media. **Results:** The designed system included hardware and software and the completed prototype had tested successfully on the real vehicles. During the testing stage, the components and devices were connected and implemented on the vehicle and the user by installing the system interface on a mobile phone is able to monitor and manage the vehicle accessories, the efficiency, adaptively and range of functionality of the system has proved with the various car accessories. **Conclusion:** This study involved design a new system to decrease the hot temperature inside a car that affecting the health of the car driver and the car driver is able to control some of the car accessories by using mobile phone. Once the car was equipped with the Bluetooth module and control system, the car accessories is able to connect with microcontroller and control by the mobile application.

Key words: Vehicular communication, Bluetooth, VANET, monitoring, smart car, vehicle controller

INTRODUCTION

Today car drivers expect more and more accessories available in their cars, but the accessories available needed manage by driver manually and not properly manage by smart system. The smart car concept had introduced to the user and a lot of accessories are available in the market for installed in the car. All these accessories are able to control by user manually using different and standalone controller. Besides, the controller itself uses RF technology which is not existed in mobile devices.

There are some controllers using Bluetooth devices as the controller for smart car control, but the controller still a standalone controller which is only specify for controlling certain accessories. Besides, the controllers available in the market are hardwired, which is circuit design for certain purpose. If the user wants to add some device to his car, the user definitely need extra controller. It is due to the difficulty to modify the existing hard wired controller.

As the Global System for Mobile communications (GSM) network start to be popular, almost everybody have a hand phone.

With the technology available, almost all hand phones are integrated with Java platform and Bluetooth module. Hence, the innovation comes out with an idea that controlling and managing all the accessories and environment inside the car via Bluetooth which is easy to modify and adopt in newly added devices. This study

Corresponding Author: Helia Mamdouhi, Department of Computer and Communication System Engineering,
Faculty of Engineering, University Putra Malaysia, Malaysia

used the java program which is run able at most of the mobile device with java platform and Bluetooth enabled.

This system will be able to remote control some functions such as switch on or off lights, pull down or pull up the power window and turn on or off the air condition fan. In additional, this system will also be equipped with temperature sensor, air absorber and microcontroller, to make it automated. If the temperature is above a threshold temperature, the microcontroller will activate the air absorber to absorb the hot air from car to exterior and to maintain a comfortable temperature inside the car all the time. This system is a complement of the smart car idea where smart car is an automobile with some Artificial Intelligence (AI) functionality.

As overview, a Bluetooth enabled hand phone will be the controller for the system after the hand phone had installed the java programs. The mobile phone is the transmitter where it will send the control command to the Bluetooth module located at the microcontroller. The microcontroller located in the car is the brain of the system and connected to air absorber, temperature sensor and power window. The Bluetooth is the medium for the system to communicate between microcontroller and mobile phone.

Smart Car Concept: A smart car is an automobile with artificial intelligence (or "AI") functionality. As automation technology has progressed, especially in the decades after the invention of the integrated circuit, more and more functions have been added to automobiles, relieving the driver of much of the mundane moment-to-moment decision making that may be regarded as having made driving tedious[1].

To accomplish all those functions which proposed in smart car concept, there are some common technologies must be included in the smart car. Those technologies needed to be included to enable the car to do some communication and computation. Those technologies included the wireless communication and computation technology. The various wireless technologies had been propose for smart car concept, there are long range communication (Wimax, GSM and 3G) and short range communication technologies which is Bluetooth and can be extended by using Mobile ad-hoc networks or Mesh networking[2].

Recent advances in vehicle electronics have led to a move toward fewer, more capable computer processors on a vehicle. A typical vehicle in the early 2000s would have between 20 and 100 individual networked microcontroller/Programmable logic controller modules with non-real-time operating systems. The current trend is toward fewer, more costly microprocessor modules with hardware memory management and Real-Time Operating Systems[2].

Bluetooth: Bluetooth was originally developed by mobile phone company Ericsson in 1994. Looking for a unique name for the technology, they settled on "Bluetooth" after a tenth century Viking king called Harald "Bluetooth" Blatand. During his reign, he united Denmark and Norway and was well known for his ability to help people communicate. Presumably, Ericsson's hope was that Bluetooth technology could do the same[3].

Bluetooth wireless technology is a short range radio communication system designed for communication of devices like mobile phones, PDAs, notebooks, PCs, printers and headsets. It should be cheap and low power consumption for easy implantation into mobile devices. First it should mainly replace cables, now this is forming complex system for communication which is able to create pico-networks not only based on packet data transfer (ACL) but also for voice services (SCO)[4].

Bluetooth technology operates in the unlicensed Industrial, Scientific and Medical (ISM) band at 2.420-2.485 GHz, using a spread spectrum, frequency hopping, full-duplex signal at a nominal rate of 1600 hops sec^{-1}. Bluetooth technology's Adaptive Frequency Hopping (AFH) capability was designed to reduce interference between wireless technologies sharing the 2.4 GHz spectrum. AFH works within the spectrum to take advantage of the available frequency. This is done by detecting other devices in the spectrum and avoiding the frequencies they are using. This adaptive hopping allows for more efficient transmission within the spectrum, providing users with greater performance even if using other technologies along with Bluetooth technology. The signal hops among 79 frequencies at 1 MHz intervals to give a high degree of interference immunity[4].

The Open Systems Interconnection (OSI) Reference Model was developed by the International Standard Organization (ISO). It presents a generic proto communications. The 'open systems' component refers to its suitability for wired or wireless network architecture. This standard is excessive in many respects, but it provides a good comparison to the Bluetooth Protocol stack. The heart of the Bluetooth specification is the Bluetooth protocol stack. By providing well layers of functionality, the Bluetooth specification ensures interoperability of Bluetooth devices and encourages adoption of Bluetooth technology[5,6]. The radio layer defines the requirement for a Bluetooth transceiver operating in the 2.4 GHz band.
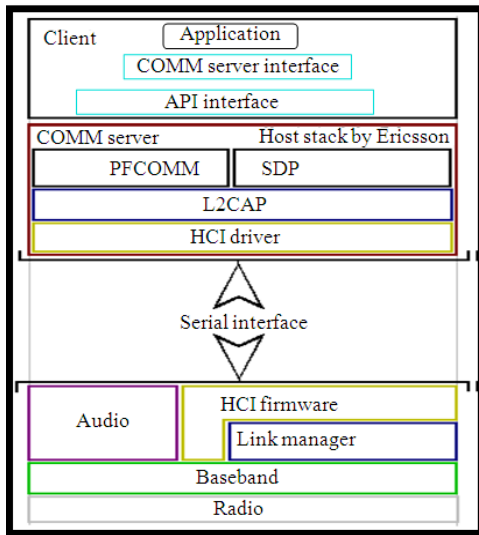
Fig. 1: Bluetooth protocol stack[7]

Figure 1 shows the detail parts of the Bluetooth stack protocol. In this stack protocol the radio module also handles transmit power levels and receiver sensitivity levels[8]. The Baseband splits the physical RF channel into slots using a Frequency Hopping, Time Division Duplex scheme. Two types of physical link exist at the Baseband level, Asynchronous Connectionless (ACL) and Synchronous Connection-Oriented (SCO). The ACL link guarantees delivery of data from the source to destination because data transfer applications cannot afford corrupt packets. Meanwhile, the SCO link provides a point-to-point circuit switched, unreliable data service. The SCO link carries voice and guarantees bandwidth rather than delivery[8]. The Link Manager (LM) performs link setup, authentication and link management using control messages (PDU). The Baseband state machine is also controlled by the Link Manager Protocol (LMP). Exchanging appropriate PDU's between peer devices establishes the Baseband physical link. The Host Controller Interface (HCI) provides a command interface to the Baseband Link Controller and Link Manager and access to hardware status and control registers[9]. The HCI provides a command interface to the Baseband controller and Link Manager and access to hardware status and control registers. This interface provides a uniform method of accessing the Bluetooth baseband capabilities[10]. Logical Link Control and Adaptation Protocol (L2CAP) supports higher level protocol multiplexing, packet segmentation and reassembly and the conveying of quality of service information[3]. The RFCOMM protocol emulates a serial cable over the L2CAP logical link established using Bluetooth.
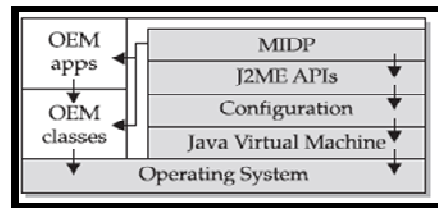

Fig. 2: Layers of the J2ME architecture[18]

It follows the TS7.10 RS-232 standard used to describe serial communication. This protocol presents the most common application of Bluetooth as a replacement to serial cables[8].

Bluetooth devices have a unique 48-bit Bluetooth Device Address (BD_ADDR) derived from the IEEE 802 standard. This address distinguishes devices during link establishment and synchronises their Baseband components during device discovery. The addresses also known as MAC addresses since it used for identify a unique device[6]. The Service Discovery Protocol (SDP) provides a means for applications to discover which services are provided by or available through a Bluetooth device. It also allows applications to determine the characteristics of those available services[3].

Java 2 Micro Edition: J2ME is a reduced version of the Java API and Java Virtual Machine that is designed to operate within the small and limited resources available in the new breed of embedded computers and microcomputers[11].

J2ME architecture consists of layers located above the native operating system, collectively referred to as the Connected Limited Device Configuration (CLDC). The CLDC, which is installed on top of the operating system, forms the run-time environment for small computing devices. Figure 2 shows the J2ME architecture which comprises three software layers[11].

The first layer is the configuration layer that includes the Java Virtual Machine (JVM), which directly interacts with the native operating system. The configuration layer also handles interactions between the profile and the JVM. The second layer is the profile layer, which consists of the minimum set of Application Programming Interfaces (APIs) for the small computing device. The third layer is the Mobile Information Device Profile (MIDP). The MIDP layer contains Java APIs for user network connections, persistence storage

and the user interface. It also has access to CLDC libraries and MIDP libraries.

A MIDlet is a J2ME application designed to operate on an MIDP small computing device. A MIDlet is defined with at least a single class that is derived from the javax.microedition.midlet.MIDlet abstract class.

MATERIALS AND METHODS

Bluetooth API-JSR 82: The Bluetooth API used in this study is Java Specification Request 82 (JSR 82). The JSR 82 is a specification that defines the optional package for Bluetooth wireless technology for J2ME. The goal of the specification is to define the architecture and the associated APIs required enabling an open, third party Bluetooth application development environment. The API is designed to operate on top of the Connected, Limited Device Configuration (CLDC)[12].

There are 2 packages defined in JSR82 API and they are (1) javax.bluetooth and (2) javax.obex[12]. The first package includes the basic Bluetooth connection and management. The second package is the Object Exchange protocol which is used to transfer files[11]. The JSR 82 API is not applicable to all mobile phone, but it supported by 373 mobile phones from 12 vendors in the market.

Security: Security is the degree of protection against danger, loss and criminals. Individuals or actions that encroach upon the condition of protection are responsible for a "breach of security"[13]. The security has already become an important part in our live. We need to secure our belonging, assets, data and privacy. There are 3 mechanisms available in Bluetooth technologies. There are authentication, encryption and authorization.

Authentication is the mechanism of verifying that a user is who they say they are by validating the user's credentials. Authorization is the function of specifying access rights to resources, which is related to information security and computer security in general and to access control in particular[14]. Encryption is the process of changing data into a form that can be read only by the intended receiver. To decipher the message, the receiver of the encrypted data must have the proper decryption key (password)[15].

Hardware connection (Receiver): Bluetooth module is powered with 3.3V. Thus, higher voltage will definitely damage the Bluetooth module while lower voltage will caused unstable operation. MAX 809 can be used to monitor the operating voltage and prevent

operation in low voltage. The connection with the module is shown in the Fig. 3.

The microcontroller is operating at 5V while Bluetooth module operates at 3.3V. The board should have voltage regulator for both device. Furthermore, voltage from microcontroller must not overload Bluetooth module. To prevent this, a voltage divider is required for UART connection as shown in Fig. 4.

Microcontroller firmware (Receiver): Figure 5 shows the firmware flow chart designed to receive command from mobile phone using Bluetooth module.

Part (a) is the main function where the program start and it always in the looping. As the firmware stated, the firmware initialize all the hardware and resources. During initialization, the firmware will read Bluetooth MAC address and passkey from EEPROM of microcontroller. The data used to configure the Bluetooth module to allow only secure connection. After initializing, the state machine enters the command mode state. During this state, the microcontroller set to receive AT command from Bluetooth module.

After the AT command received, the microcontroller process the AT command and decide which state to be.

After the connection had been established, the state machine changes from the command mode to bypass mode which is shown part (b) in Fig. 5. In the Bypass mode, the microcontroller receives the control command from Bluetooth. As the control command received, the microcontroller interprets the command and generates a few number of control signal to the actuator. The actuator can be motor, fan, LED and air-absorber. The control algorithm is done by another partner in this study.

Part (c) is the interrupt functions, where it will be invokes once there is one character store in UART receive buffer. The function will copy the received ASCII character into an AT Command array. After end of command, the Line feed ("enter") character received, it will set the AT command received flag to true if the Bluetooth not yet established a connection.
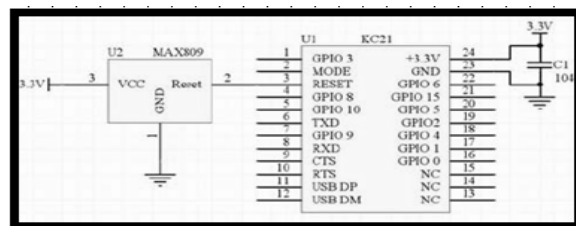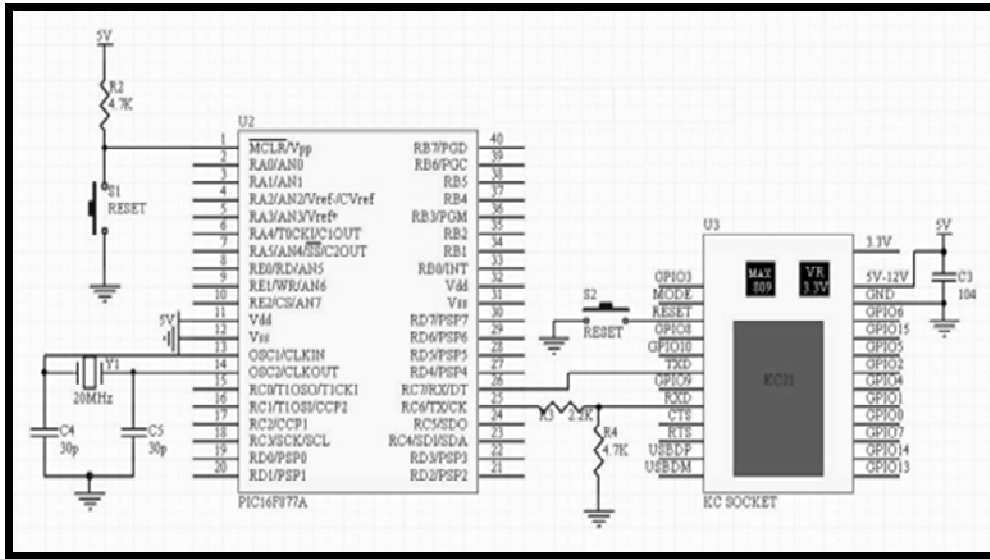

Fig. 3: Max 809 and KC-21
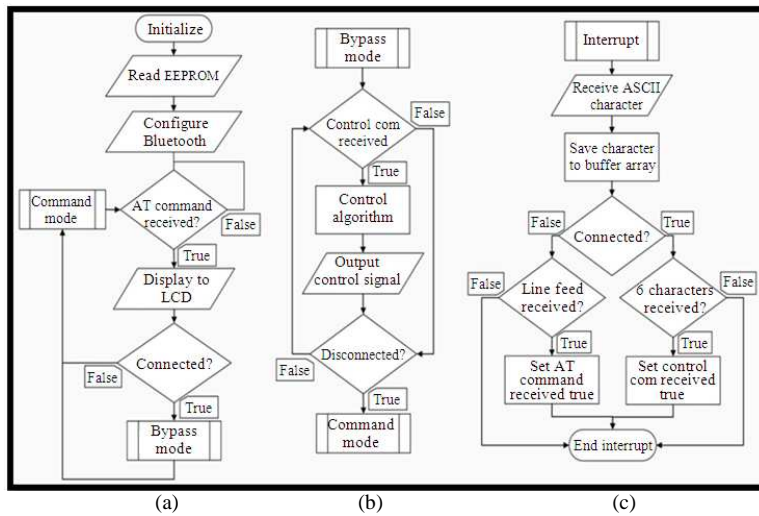
Fig. 4: Schematic of KC-21 with PIC16F877A

Fig. 5: Flow chart of the Bluetooth enabled receiver firmware design. (a): Command mode state (b): Bypass mode state (c): Interrupt function

The command mode state immediate starts its function to analyse the command. If the Bluetooth module already connected with the mobile phone, it will set the control command received flag to true after 6 characters have been received. The control command is then process by the control algorithm to generate control signal.

Java Mobile Application (Transmitter): When a controller or transmitter (client) needs to connect with the microcontroller (server), transmitter needs to start

inquiry, once the inquiry completed, the user can select which device he wants to establish the connection. As the device selected, the code will continue to search for services available and get the connection string which is used to establish the connection.

After the connection had been established, MIDlet program will guide user selects the command wanted to be send. When the user chooses to disconnect, the MIDlet will close the connection and back to main menu. The flow chart in Fig. 6 shows the main program flow.
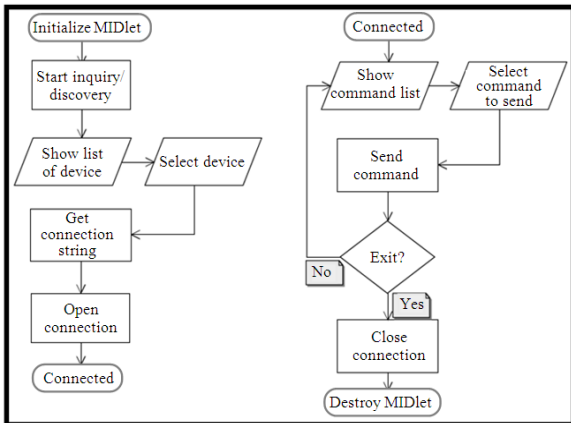
Fig. 6: Java MIDlet controller flow chart

```
String url=
serviceRecord.getConnectionURL
(ServiceRecord.AUTHENTICATE_NOE
NCRYPT,false);
```

Fig. 7: Connection string

The Java APIs for Bluetooth supports secure client and server connections. There are 2 types of securities provided for client application there are authentication and encryption. The client need to connect to a secured server, the connection string get must set to secured type of the server. For this study, authentication is used at receiver, so the connection string get should use the code as shown in Fig. 7.

By using this code, the connection to the server should perform the authentication with no encryption.

RESULTS

Microcontroller (Receiver): The receiver will display any command received from Bluetooth link and display connected and disconnected message at display LCD. Once the mobile application successful connected to Bluetooth module at the receiver, the LCD will display a connected message. On the other hand, the disconnected message on the LCD screen, the message will be display if the mobile phone is out of range or disconnected from receiver by user.

When the connection is up, the user is able to send the control command to the receiver. Once the command is received, the LCD screen will display the appropriate actions. All the control command received at receiver.

During the testing stage, the components and devices are connected and testing on a bread board for easier modification and debugging. The bread board allow us to plug in and plug out the wire easily.
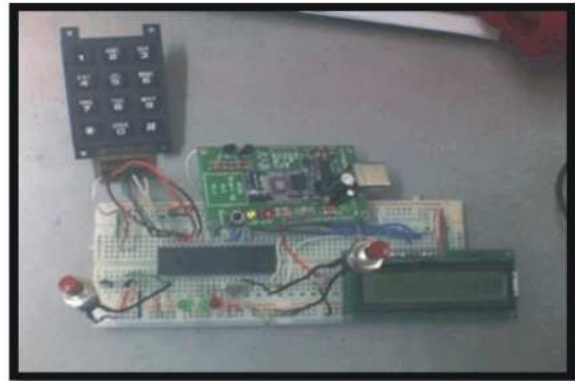

Fig. 8: Test circuits on breadboard

Fig. 9: Prototype

Once the design is working fine, the circuit is then transfer to a donut board. For the donut board, all the components had soldered on the board. Figure 8 and 9 shows the testing circuit and completed prototype.

Java Mobile Application (Transmitter): Once the mobile phone starts the program, the main menu is shown and user is able to select the action listed. There are 2 actions available in the main menu which is shown in Fig. 10. When the start is selected, the system will start to search Bluetooth device available around. On the other hand, the properties action will show the properties of the Bluetooth embedded in mobile phone

At the time Bluetooth is being connected, the user is prompted to key in passkey for secure link connection between mobile phone and microcontroller and this process is known as bonding the passkey authentication is only able to activate in real mobile after the bonding successful, the control command list is display on the screen. The control command list is shown in Fig. 10. The user is able to select the command and send to microcontroller.

Fig. 10: Commands list

DISCUSSION

Firmware design and Bluetooth module: The Bluetooth module selected is the main problem during the design the Bluetooth link. The Bluetooth module available in the market which is able to connect with microcontroller is very less. The Bluetooth module bought with higher price compare to other components in this study. If this study brings to commercialize, the cost for the Bluetooth module should able to reduce.

At first, the character receive function is done inside program but there is some character missing randomly. This is because the structural program flow is time consuming, so sometimes the newly incoming character is replacing the old character in the buffer. So forth, the interrupt function must invoke, when a character receive in UART buffer of microcontroller. This is important to make sure all the character received and processed.

Besides, the PIC 16F877A had defines 2 of the I/O port as the UART transmit and receive port with hardware support. There are PIN_C6 as transmit and PIC_C7 as receive. This knowledge is important, because without the hardware support, the UART interrupt function is unable to invoke and cannot working with highest baud rate (115200Baud).

The string process in C is limited; the library "string.h" included in CCS C has a lot of limitation during process the array of character received. The processing of string required large amount of RAM of the microcontroller. Hence, the algorithm of store the string and compare them with the received command in limited memory PIC 16F877A is impossible for implementation. The failure of the algorithm is then replace by a new algorithm which copy each command to temporary variable for compare with the command received. The memory used for the temporary variable will be release upon finish compared, so there will be sufficient memory by changing to the new algorithm.

Java Programming and JSR 82: The Java application development is designed and simulated using Java Wireless Toolkit, the entire program working fine, but a lot of bugs found during implement into a real mobile phone. The sending message function is working well during the simulation, but the receiver is unable to receive the command sent. It is because in the real implementation on hardware there is some delay exist due to hardware delay which is non significant during simulation. Hence, the mobile application was design to halt for a while before proceed to next instruction just after sending command.

Different mobile phone has different processing speed and has compatibility problems. The Java application was initially tested using Motorola L7. Since it is working fine, hence the application is further improves to have some image icon for a better appearance. The program is then not working well afterward, but the same program has working well after implemented on Sony Ericson Z610i.

Programming in Bluetooth basically needs the fundamental knowledge of the technology itself. This is shown in the implementation of different languages and APIs involved in developing the software. At the initial stage the software was planned to make all Bluetooth devices compatible but it is almost impossible due to different manufacturers have their own driver defined for their hardware.

The JSR 82 Bluetooth library provided by Sun Microsystems is the generic library that support by most of the mobile phone, but there is some of the mobile phone is not compatible with the library. The complete list of JSR 82 compliant mobile phone is attached as appendix B. Since, it is free and support most of the mobile phone, so it is chosen as the design and development library. The mobile phone that does not compliant with JSR 82, because there vendors have their own library which is compatible with their Bluetooth stack.

Even though the mobile phone is not compatible with the JSR 82, the java application developed in this study is able to convert it to compatible library because the flow of program will be the same.

CONCLUSION

This study involves design a new system to decrease the hot temperature inside a car that affecting the health of the car driver and the car driver is able to

control some of the car accessories by using mobile phone. Once the car is equipped with the Bluetooth module and control system, the car accessories is able to connect with microcontroller and control by the mobile application.

By using this system, car driver is able to use their existing mobile device to control car accessories such as power window instead buy a new controller to control the system. The Java MIDlet program can install in most of the mobile phone or Personal Digital Assistant even laptop. Hence, the mobile device can used to control the car accessories rather than buying different standalone controller to control different accessories.

The controller is a flexible controller, where the controllable accessories are easy to manage. The controllable accessories are set depending on users demand because the mobile application is not hard wired and able to change.

ACKNOWLEDGEMENT

REFERENCES

1. en.wikipedia.org., 2009. Smart car-Wikipedia, the free encyclopedia. http://en.wikipedia.org/wiki/Smart_car
2. en.wikipedia.org., 2009. Intelligent transportation system-Wikipedia, the free encyclopedia. http://en.wikipedia.org/wiki/Intelligent_Transportation_System
3. EE Times, 2009. How Bluetooth got its name? http://www.eetimes.eu/scandinavia/206902019?cid=RSSfeed_eetimesEU_scandinavia
4. Bluetooth.com, 2008. Bluetooth.com|Basics. http://www.bluetooth.com/Bluetooth/Technology/Basics.htm
5. Siew, L.S., 2007. Laptop Data Protection Using Bluetooth Enabled Mobile. UPM, Malaysia, pp: 13-15.
6. Bluetooth.com, 2008. Bluetooth.com | Core System Architecture. http://www.bluetooth.com/Bluetooth/Technology/Works/Core_System_Architecture.htm
7. sandbox.cz, 2008. Bluetooth technology. http://sandbox.cz/~covex/bluetooth/Bluetooth.pdf
8. Naveenan, V., 2001. The Bluenurse wireless link. http://innovexpo.itee.uq.edu.au/2001/papers/s369535/thesis.pdf
9. Bhagwat, P., 2001. Bluetooth: Technology for short-range wireless apps. IEEE. Internet Comput., 5: 96-103. DOI: 10.1109/4236.935183
10. KCWireFree, 2008. KCWireFree. kcSerial User Guide. www.kcwirefree.com/docs/kcSerial_UserGuide.pdf
11. Keogh, J., 2003. J2ME: The Complete Reference. McGraw-Hill/Osborne, USA., ISBN: 0072227109, pp: 745.
12. Motorola. Java APIs for Bluetooth Wireless Technology (JSR 82) Specification Version 1.1.1 Java 2 Platform, Micro Edition. 2008. https://prof.hti.bfh.ch/knr1/FH/FBI/MobileComputing/Connectivity/Bluetooth/Documentation/JSR82-spec_1.1.1.pdf
13. palowireless.com, 2008. Serial Port Profile. http://www.palowireless.com/INFOTOOTH/tutorial/k5_spp.asp#Profile%20Overview
14. What is authentication?-A definition from Whatis.com. searchsecurity.techtarget.com. 2009. http://searchsecurity.techtarget.com/sDefinition/0,,sid14_gci211621,00.html#
15. SearchSoftwareQuality.com, 2001. Authorization: Searchsoftwarequality.techtarget.com Definitions. http://searchsoftwarequality.techtarget.com/loginMembersOnly/1,289498,sid92_gci211622,00.html?NextURL=http%3A//searchsoftwarequality.techtarget.com/sDefinition/0%2C%2Csid92_gci211622%2C00.html