使用藍芽散射網路實作的智慧型車輛多工診斷系統

The Implementation of Multiplexing Diagnostic System Using Bluetooth Scatternet for Intelligent Vehicles

Cheng-Min Lin [(1)]	Chyi-Ren Dow [(2)]	Chen-Wei Yang [(2)]	Jyh-Horng Lin [(1)]	Hung-Jie Shih [(2)]
lcm@nkc.edu.tw	crdow@fcu.edu.tw	P9525258@fcu.edu.tw	jhlin@nkc.edu.tw	M9506772@fcu.edu.tw

[(1)] Department of Computer and Communication Engineering
Department of Electrical Engineering
Nan Kai Institute of Technology
[(2)] Department of Information Engineering and Computer Science
Feng Chia University

摘要

藍芽是目前智慧型車輛上最受歡迎的無線通訊技術之一。在許多新型的智慧型車輛中，診斷掃瞄器也已使用藍芽通訊取代電纜線進行車輛診斷作業。然而，目前診斷作業對於技師來說，繁雜的操作或程序是不常使用的主要因素之一。本篇文章提出使用藍芽散射性網路架構來設計具有可同時進行多方傳輸功能的診斷系統，此系統可同時覆蓋更廣的範疇和提供更多車輛同時進行診斷的機會，並使用標準的自動化診斷程序，使定期的維修工作變得更容易。在遠端診斷方面，設計高效能閘道器橋接控制區域網路和外部無線網路是一個重要議題。本篇文章提出散射網多工和排隊串接兩種診斷模型。在系統雛型建立方面也探討閘道器、橋接器、及診斷伺服器等實作技術。此外，我們分別制用控制器區域網路分析儀進行實際量測和使用M/M/c/K排隊模型來計算此系統的效能。經由實驗結果顯示出散射網多工診斷模型比排列串接診斷模型來得有效率。

關鍵詞：藍芽散射網、智慧型車輛、控制器區域網路、診斷和閘道器。

ABSTRACT

Bluetooth is a popular wireless technology for various intelligent vehicle applications. In some modern vehicles, diagnostic scanners have employed Bluetooth handset to diagnose ECUs (Electronic Controller Units). However, those complicated operations and procedures are not useful for mechanics. In this paper, a multiplexing diagnostic system is proposed using Bluetooth scatternet in the automobile service department, which covers wider space and more vehicles to make diagnostic automatically and easily. In remote diagnostic system, the gateway is an important issue for the bus inside the vehicle to connect with outside wireless LANs. The architecture connecting to Bluetooth scatternet and CAN (Controller Area Network) bus is emulated. Bluetooth scatternet communication capacity and interference affect the gateway implementation. The experiment results would offer a way to develop effective gateway estimate the performance between Bluetooth scatternet and CAN bus.

KEY WORDS: Bluetooth scatternet, intelligent vehicle, CAN bus, diagnostic and gateway.

1. Introduction

The main objective of the ITS (Intelligent Transport System) is to integrate man, vehicle and information of roadway for providing a safe environment of transportation and solving practical traffic problems. In the researches of intelligent vehicle, it has became an important subject for mobile industries and academic organizations how to apply technologies of electronic, information, machine and sensor in the ITS.

Most modern vehicles today utilize CAN bus [6] as a standard control network for internal communication between sensors, instruments and control units. CAN bus is cheap, reliable, and durable against electrical disturbances, thus it is well suited for vehicle industry. While the acceptance of CAN bus has grown over the years from 2000 to 2007, it has become necessary to standardize services possible to perform using this network. All of services in the CAN bus covers everything from the testing of exhaust levels of fumes, to the reading of sensor values, as well as the downloading of new software. The sets of services that can be performed are referred to as diagnostics. If an error occurs, for instance, a sensor will report an erroneous value, which will be logged as a DTC (Diagnostic Trouble Code) in the ECU. It is now possible to extract and interpret the DTC when fixing the vehicle, which yields the problem, and by this replace the faulty sensor.

When accessing CAN bus, a cable has to be connected to the vehicle, which limits the freedom of vehicle movement. An example would be that when a mechanic performs tests with his/her scan tool (say to check if all the lights on a truck are operational), the mechanic needs a rather long cable to reach around the trailer. In this case, a wireless link would be a preferred alternative. Hence, Bluetooth technology could be used as the link between the diagnostic tool and vehicles CAN bus [7].

In this paper, we propose a multiplexing diagnostic architecture based on Bluetooth scatternet [10] used in the automobile service department. The multiplexing diagnostic system initially connects to a database server via a wireless channel and then constructs a standard diagnostic procedure when Bluetooth devices are distributed around the service department to serve vehicles in this area. This system also reduces the mechanic’s loading in vehicle maintaining and avoiding mistakes of work done by hand.

The rest of this paper is organized as follows. The related research is described in Section 2. Bluetooth and CAN bus on vehicle protocol are presented in Section 3. An automatic diagnostic architecture in the automobile service department is described in Section 4. Implementation details and emulation for the proposed system is described in Section 5. Section 6 illustrates numerical results, including monitored by analyzers and simulated by a queueing model. Finally, Section 7 concludes the study.

2. Related Work

There are many applications of Bluetooth for vehicles [2,3]. For example, Jorge *et al.* [1] uses Bluetooth sensor network for remote diagnostics in vehicles. Data from sensors installed in vehicles can be transmitted wirelessly through a Bluetooth network. The approach is similar to us, but they don’t use CAN bus to diagnose vehicles and can not diagnose vehicles concurrently.

In the article [4], it introduces a two-level in-vehicle application consisting of system level and user device level. At the system level, it could be used to replace the wired connection in the nonsafety-critical control network, and brings convenience to the vehicle's diagnosis, repair and software upgrades. At the user device level, integrating the mobile electronic devices to the vehicle to form a wireless network, it could provide people with a convenient and multifunctional device interface for the business and entertainment objectives on travel. The paper only provides a concept that to use the Bluetooth wireless connection to replace the wired connection in CAN. It does not consider diagnose vehicles concurrently and no further implement on it.

The article [15] discusses the potential of Bluetooth within the automotive industry. Currently it is estimated that more than 80% of all innovations within vehicles are derived from electronic systems. Bluetooth technology is used to reduce wire cost of the electronic systems.

The safe operation of vehicles on roads is very important. There are many diagnostic systems are develop for safety. In [22], a model-based diagnostic system for air brakes is presented. The diagnostic system is based on a nonlinear model for predicting the pressure transients in the brake chamber. Kher *et al*. [17] attempt to design a diagnostic system for detection of faults based on neural network. The system is developed based on a fault table for the engine. Such a diagnostic module is aimed to increase the utility of the system. Shrivastava and Rajamani [21] develop a fault diagnostic system to monitor the health of the lateral motion sensors on an instrumented highway vehicle. The system utilizes observer design with the observer gains chosen so as to ensure that each sensor failure causes estimation errors to grow in an unique direction. Sawant *et al.* [19] present a novel approach to increase the safety of road travel using Bluetooth protocol. Vehicles can exchange data sensed by the on-board sensors form mobile ad-hoc networks. The fusion of these data could give a better understanding of the surrounding traffic conditions.

3. Preliminary

Research backgrounds concerning this work are introduced in this section, including Bluetooth and CAN bus. There are four major components in Bluetooth wireless technology system, as the Core [23] states: radio unit, baseband unit, software stack, and application software. The radio unit dose what its name implies, which is to carry the data packages into the air. The baseband unit/link manager is implemented in hardware using ASIC (Application Specific Integrated Circuit), which provides a fast interface between software and radio unit. The baseband/link manager hardware provides all required functionality to establish and maintain a Bluetooth wireless connection. The software stack is essential driver software, which makes it possible for the application (i.e. user defined program that performs customized tasks) to interact with the baseband unit.

To run an application on a Bluetooth chip, some kind of software is necessary to complete the interface between the application and the hardware, which is called the Bluetooth stack. Stack relates to that the different layers are stacked on top of each other. An incoming message entering from bottom (the radio link) must pass all layers, finally reaching the application. Some of the top modules of the core are optional.

Bluetooth is a low-cost, low-power, and short-range radio WPAN [8,12] (Wireless Personal Area Network) technology, using piconets and scatternets to connect between mobile electronic devices. Bluetooth uses TDD (Time Division Duplex) and FHSS (Frequency Hopping Spread Spectrum) technology to transmit SCO (Synchronous Connection-Oriented) data and ACL (Asynchronous ConnectionLess) data. In ACL data transmission, the data packet formats are divided into DM and DH types. Such as DM1 packet carries 1-18 payload bytes, plus a 16-bit CRC code. The DM1 packet occupies a single time slot and the information plus CRC bits are coded with 2/3 FEC. The DH1 packet is similar to the DM1 packet, except that the information in the payload is not FEC(Forward Error Correcting) encoded. As a result, the DH1 packet has between 1 and 28 information bytes plus a 16-bit CRC code.

A Bluetooth piconet contains one master device, and up to seven active slave devices. A master or a slave that participates in two or more piconets can act as a bridge, which connects two or more piconets to form a scatternet. A bridge node is also called a PMP (Participate Multiple Piconet). A PMP can play a role of Master/Slave or Slave/Slave bridge inter-piconets [18].

As mentioned earlier, CAN is a multicast-shared serial bus; its maximum transfer rate is 1Mbit/s, assuming the cables are not too long and that the environment is fairly shielded from disturbances. Under normal situation, however, networks do not run faster than 250 Kbit/s, or maybe 500 Kbit/s. This has all been standardized in the Bosch CAN specification [26]. There are two signal wires called CANH (CAN High) and CANL (CAN Low), which are used for both receiving and sending messages (contrary to say RS-232 communication where one wire is used for transmission in each direction). CANL has a digital signal output of 0-2.5V, where as CANH has an output of 2.5-5V, and is inverted compared to CANL.

With this setup, it is possible to perform an operation called differential signalling. When CANH is subtracted from CANL, the remaining part is a digital signal with the output of 0-5V. When a disturbance occurs, it is likely that it will take place on both wires simultaneously. Since the signals are subtracted, the disturbance will more or less vanish.

Every ECU is connected to the bus serially, which requires lesser cables. But if breaks at a critical place, it might disconnect several nodes from the chain. The cables used should be of the impedance 120Ω, but must also fairly flexible.

For data link layer, there are two different sizes of CAN messages available, one with an 11-bit identifier called standard CAN or CAN2.0A, while the other one uses a 29-bit identifier and is called extended CAN or CAN2.0B. Extended CAN is replacing standard CAN since it can carry more information in the address field, even though the bandwidth is reduced. CAN2.0A is used in the proposed system.

Today, most modern intelligent vehicles have more than one bus. Usually there is one high speed CAN bus and a low speed CAN bus. On top of this, some vehicles have optical buses, which are used to transfer large amount of data. An ECU usually has sensors that take the measurement on a regular time basis. If a measurement somehow does not correspond to pre-defined allowed values, it is then stored in the ECU as a DTC. Reading DTCs is only one of many services implemented in the nodes of a vehicle. The ISO 14229-1 [29] document specifies how services are structured, how to read the answers and so on. ISO 15765-2 [28] specifies diagnostic network layer, it defines how to transmit multi-frame between sender and receiver.

4. System Architecture

In this section, we will first introduce two proposed models, queued model and scatter model, and then present diagnostic service. In addition, the protocol stack including gateway, bridge, and host, and diagnostic sessions are mentioned.

4.1 Queued Model

In the traditional diagnostic service, an inspection stall is just for one car. Each car must waits until the preceding car is served completely. This service pattern is called queued model, as shown in Figure 1. On the other hand, only one vehicle is served at the same time, or to serve more vehicles use more diagnostic scan tools and mechanics at the same time. If the scan tools or mechanics are busy, other arrival car queued up for diagnostic service. If there are too many cars want to be served, we must have more the inspection stalls to serve the cars. The cost is increased when more the inspection stalls and scan tools need to buy for serving those cars. If increate the capability of the inspection stall, for example, every inspection stall can serve more than one car, the cost for buying inspection stall and to setup diagnostic service will be decrease.

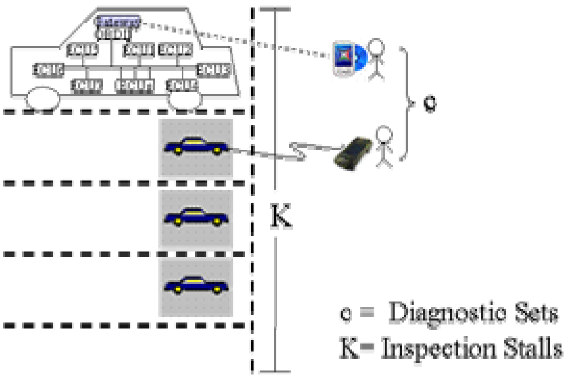

Figure 1 Traditional diagnostic service is modeled.

For the performance evaluation of vehicle diagnostic service for vehicle, a queueing model of *M/M/c/K* [9] is used to model our system. The series of symbols to denote Poisson-input, exponential-service, *c* exponential servers and waiting space of size *K-c* respectively. Denoting the average rate of customers entering the queueing system as λ, the average rate of serving customers as μ and the number of parallel service channels as c, a measure of traffic congestion for c-server system is $\rho \equiv \lambda / c\mu$.

4.2 Scatter Model

In order to solve above problem, a model based on Bluetooth scatternet is proposed in here. We call the model is scatter. In the proposed model, the wireless network is constructed by Bluetooth scatternet. The protocol of diagnosis is used for large DTCs transmission, as well as updating software of ECU on vehicles. In the automobile service department, Bluetooth scatternet can be designed to cover full space and to serve more vehicles at the same time. The Bluetooth scatternet includes a diagnostic host, bridges and gateways constructed using Bluetooth technology.

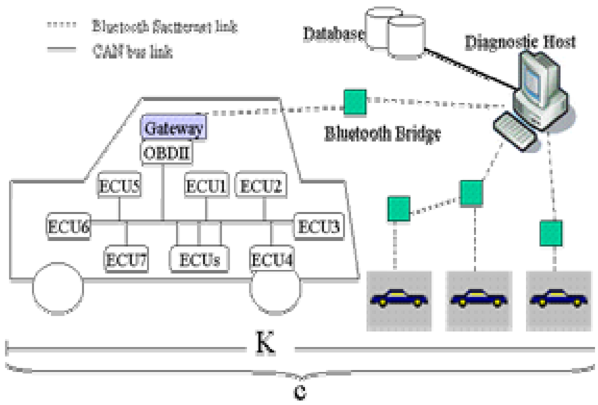

Figure 2 The multiplexing diagnostic system consists of host, Bluetooh bridge and gateway.

Bluetooth scatternet can support point to multipoint adapters [20]. In automobile service departments, Bluetooth devices can be distributed by tree formation [5, 16], and the root node will play as the master role to account for establishing links among bridge nodes. When the vehicle is driven into the service department, the Bluetooth-CAN bus gateway is then plugged into the OBDII (On Board diagnostic) connector to establish a link between CAN bus and Bluetooth scatternet.

The node of the root is called a diagnostic host, which can initiate a standard diagnostic procedure, and connects to a database server to record the history of diagnostic. When a vehicle connects into the Bluetooth scatternet, the diagnostic host will start the diagnostic service procedure and display the errors occurred on the vehicle at the panel of diagnostic host. The bridge nodes play a Master/Slave role between diagnostic host and Bluetooth-CAN bus gateway. The bridge nodes will inquiry routinely until a Bluetooth-CAN bus gateway is discovered. Then the bridge node will page to establish a link and bypass data between the gateway and the diagnostic host. The bus gateway plays a slave role in Bluetooth scatternet and relays messages between Bluetooth and CAN bus interface. In addition, the gateway can plug in OBDII connector to communicate with ECUs on the vehicle.

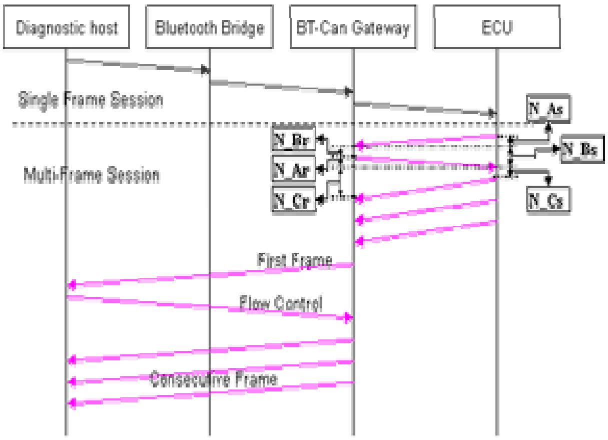

Figure 3 An interaction diagram illustrates single frame session and multi-frame session in the session-based diagnostic service.

4.3 Diagnostic Services

CAN bus is connected by ECUs on a vehicle. Those ECUs interact and manage each other by CAN bus. When the vehicle is in stand-by mode, Bluetooth-CAN bus gateway can be employed to connect Bluetooth scatternet and to diagnose the vehicle. When diagnostic host starts a diagnostic session, the ECU will stop interacting with other ECUs and transmits the service respond to diagnostic host. The sets of services include many sessions, such as security access, read DTC, read/write ECU identifier and input/output sensor control. Those session-based services use single frame or multi-frame protocol to request or respond a service, as shown in Figure 3. In CAN bus protocol, the single frame only carry 0~7 bytes data payload and the multi-frame can carry 8~4095 bytes data payload. In the Bluetooth scatternet, we use DM1, DM3, DM5, DH1, DH3 and DH5 packets payload length to define the single frame protocol payload length, because the single frame will be divided into continuous slots in the session-based protocol. In emulation, we will analyze the performance in each payload length.

4.4 Protocol Stacks

The protocol stack in multiplexing diagnostic system is shown in Figure 4. CAN bus in intelligent vehicles diagnostic protocol stack is used to construct the system protocol. In Bluetooth scatternet, The multiplexing forwarding protocol frame is used to embed a source node BD (Bluetooth Device) address, a remote node BD address, CAN2.0A identify, and data payload, as shown in Figure 5. It supports point to multipoint adapter and forwarding around nodes in a Bluetooth scatternet.

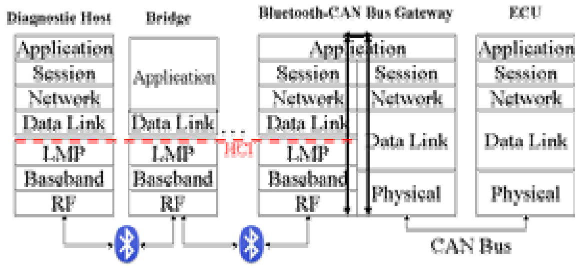

Figure 4 The protocol stacks include diagnostic host, bridge, gateway, and ECU.

Source BD_ADDR (6 Bytes)	Remote BD_ADDR (6 Bytes)	Identify (2 Bytes)	Data Payload

Figure 5 Multiplexing forwarding protocol frame

By the definition of multi-frame scenario in ISO 15765-2, there are three timeout parameter values that should be defined as follows.

1.Network time interval A on the sender side/network time interval A on the receiver side (N_As/N_Ar).

2.Network time interval B on the sender side/network time interval B on the receiver side (N_Bs/N_Br).

3. Network time interval C on the sender side/network time interval C on the receiver side (N_Cs/N_Cr).

The network time intervals A, B, and C are the time to transmit/receive a network protocol data unit (NPDU), receive the next flow control of NPDU, and next consecutive frame of NPDU, respectively. In the protocol of CAN bus, we define N_As/N_Ar=25ms, N_Bs/N_Br= 75ms and N_Cs/N_Cr=150ms as the timeout parameters of window size. Those values of timeout parameter should be extended in Bluetooth scatternet, because the serial data transmission delay between host and Bluetooth module, and the propagating delay time between source and remote nodes will be increased. Hence, the maximum delay time between diagnostic host and gateway in this system will be formulated.

The total transmission data bits are n, the serial port baud rate is S bps, and there are x segments for serial data sending and receiving. The number of bridge nodes is ***BN***, and the maximum propagating delay is T_{Delay}. The total delay time in Bluetooth scatternet is $T_{BTS-Dalay}$, the value of $T_{BTS-Dalay}$ is thus:

$$T_{BTS\_Delay} = \frac{nx}{S} + (T_{Delay} \times (2BN - 1)) + \mathrm{T}_{\mathrm{IR}} \quad (1)$$

T_{IR} is a response time which is the time to execute instructions of processing a received package in the controller computing. The number of bridge nodes ***BN*** is relative to the segments of ***x***. The message sends to a bridge node from a diagnostic host, and bypass to a gateway. If there is one bridge node, the transmission segments through serial port are **4**. Each bridge node will receive a message and forward the message to the other side and send from a diagnostic host to a gateway or from a gateway to a diagnostic host, so the segments ***x*** equals **(2BN+2)**. For control scatternet bandwidth, the sniff mode is used to release the link into the other piconet. The parameters of sniff command include sniff maximum interval, sniff minimum interval, sniff attempt and sniff timeout. The value of T_{Delay} is related to sniff parameters including sniff attempt and sniff timeout. If the sniff attempt parameter is ***A*** slots, and sniff timeout parameter is ***P*** slots. The bridge node will serve the link of sniff mode during ***A*** slots time then switch to the other piconet during ***P*** slots time. The maximum delay time in the link of sniff mode will be ***P*** slots and in the other link the maximum delay time will be ***A*** slots. We can set the value of computing time to zero, because the computing time approaches several microseconds. The formulation of T_{BTS_Delay} is thus:

$$T_{BTS\_Delay} = T_{IR} + \frac{2n(BN+1)}{S} + \frac{(\mathrm{P}+\mathrm{A})(2\mathrm{BN}-1) * 0.625 * 10^{-3}}{2} \sec \quad (2)$$

4.5 Scenario of Automatic Diagnostic Session

In the automobile service department, the link always exists between diagnostic host and bridge. When a Bluetooth-CAN bus gateway is plugged into an OBDII connector, and the link between bridge and gateway is established by inquiry and page routinely, Bluetooth-CAN bus gateway will send Link-Established-Finish packet to the diagnostic host through bridge node. The bridge and the diagnostic host will update the BD address in the forwarding table, and the diagnostic host will initiate the diagnostic session.

4.6 Safety and Security in Diagnostic Session

The safety is the most important issue for vehicle manufactories to design vehicles. The external diagnostic services should be used when the vehicle is in stand-by mode, it mean when the vehicle is in power-up mode or driving on the road, the ECUs will not respond the external diagnostic sessions. The ECUs on road vehicle should only execute its own the specific functions. So the vehicle should be in stand-by mode (such as the key position is in before power-up mode) and the external diagnostic sessions can be used to serve the vehicle. When the vehicle is in stand-by mode, there is no safety issue for the external diagnostic services.

Security is handle by the ISO 14229-1 protocol. Using this protocol it is possible to secure all requests and nodes or just the ones necessary for specific purpose, such as security access session. The parameters of security access session can be defined by the manufactory to use uniform code or dynamic code to access the other services.

5. Implementation

In this section, how to implement multiplexing diagnostic system using the Bluetooth scatternet is described.

5.1 Bluetooth Scatternet

The CSR (Cambridge Silicon Radio) [31] BlueCore 2 External chip is used to develop Bluetooth communication network. This chip includes those devices: radio modulation /demodulation device, MCU, RAM, DSP, I/O interfaces. I/O interfaces include SPI, UART, USB, PIO and PCM communication interface. This chip can be developed a Bluetooth module to add XTAL and 8 MBytes flash ROM in it. The XTAL is a crystal radio.

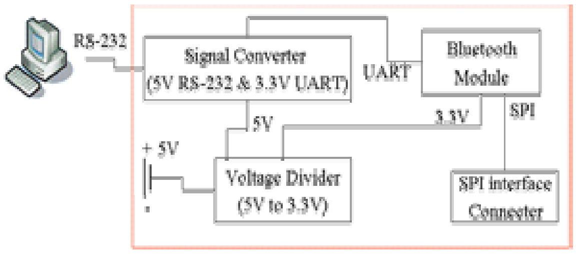

Figure 6 The circuit diagram using RS-232 to communicate with BT module.

The module embeds HCI stack, RFCOMM stack and SDP stack in the flash ROM. To implement the Bluetooth scatternet, the HCI stack should be used, the firmware version should be above HCIStack 1.1v16.4 [27], and this version supports only PMP to play the Master/Slave role. The UART interface is selected to communicate between the host and the Bluetooth module in this project, the circuit diagram is shown in Figure 6. In the PC-based implementation, the UART signal can be converted to RS-232 signal by HIN-232 chip. The Bluetooth module work voltage is 3.3V, so the signal converter should switch signal voltage between 5V RS-232 signal and 3.3V UART signal. To develop Bluetooth scatternet, an important issue is setting the Bluetooth firmware to support scatternet. The SPI interface connecter is used for the PS (Persistent Store) key tool to configure the firmware setting, such as enable 2.5 piconets support, setting communication interface and some usable parameters. The PS tool is developed by CSR and uses a printer port to connect with a Bluetooth module with SPI connector to set Bluetooth chip options.

The 8051 CPU is selected for bridge implementation, because the 8051 CPU is cheap, low power and the process speed is enough to handle to communicate with the Bluetooth module [13]. We use W77E58 CPU designed by Winboard and its crystal speed is set in 11.0592 MHz to implement the bridge node. This CPU supports 2 UART serial ports, the UART0 is selected for Bluetooth module communication and UART1 is reserved for convert to RS-232 to communicate with PC for debug using. For communication of Bluetooth module, the signal voltage in UART0 should be reduced from 5V to 3.3V, because the work voltage of Bluetooth module is 3.3V. The CPU port P0 is connected to the scan type LED and be used for to display the receiving data. The CPU port P1~P4 are reserved for using in future, The Bridge hardware is shown in Figure 7.

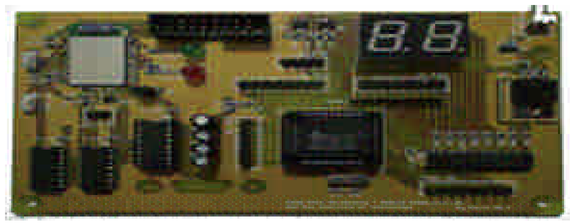

Figure 7 The bridge hardware.

In Bluetooth, the main state of link establishment is from stand-by state to connection state. In this system development, we develop a reusable software library for those devices connection. Bluetooth scatternet can use sniff mode and hold mode to release the link and connect with the other link [24,25]. In our system, the sniff mode is selected to implement. The reusable library is developed following the link establishment state machine, as shown in Figure 8.

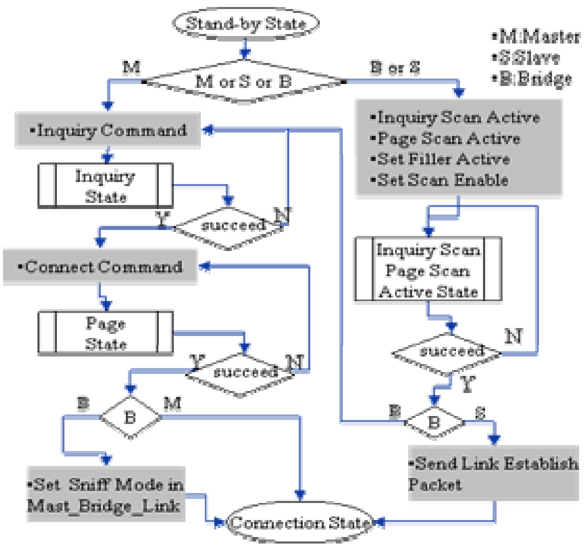

Figure 8 Link establishment state machine.

5.2 CAN Bus

For implementation of our gateway hardware, the prototype development environment uses PC-based to develop. Hence, the PC uses RS-232 to communicate with Bluetooth module and the adapter uses Advantech PCI-1680U CAN bus card [32] to communicate with CAN cable. The ECU also uses PC and PCI CAN bus card to emulate the communication on the CAN bus. This card support two CAN bus channels. The DB-9 connector pin assignments are CANH in pin 7, CANL in pin 2 and the ground in pin 3. This CAN bus card uses two on board jumpers to set the terminator resistor. In the jumper close mode (connect the pins with the clip), the terminator resistor is set to 120Ω.

6. Experiment Results

The experiment uses the prototype of multiplexing diagnostic system to emulate. Vector's CANscope [33] is used to monitor data on CAN bus, as shown in Figure 9. CANscope offers the real time trigger, and the time resolution is millisecond. CANalyser [34] is the software used to monitor and analyze message scenario on CAN bus [14,30]. The communication speed on the CAN bus is 500 Kbit/s, and the overhead data will affect the latency. One CAN frame size is 120 bits, so the total transmission time will be 0.24 ms [11].

In the first experiment, the relation between inquiry time and distance is an important issue to construct the multiplexing diagnostic system. The class 2 Bluetooth modules are used to emulate in this experiment. The module maximum working distance is 20 meters and 0dBm output power. Power consumption for transmitting is 60mA and receiving is 30mA. There are two Bluetooth modules are distributed the distance range between 1~20 meter. For each meter, we try the inquiry time N*1.28 sec from N=1 to N=4. For each N value, we experiment 100 times. The result is shown in Figure 10. The inquiry time should be set more than 3.84 seconds, because the amount of inquiry procedure time is more than 2.56 seconds. When the inquiry time is set to 5.12 seconds, the inquiry success rate will reach to 100%.

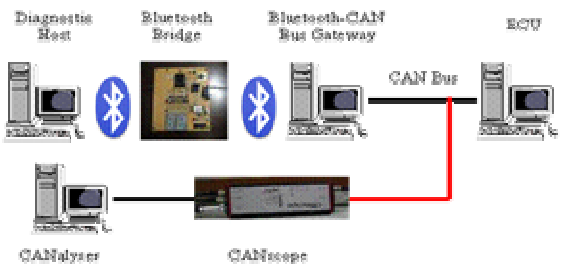

Figure 9 The experiment architecture is monitored by CANalyser and CANscope..

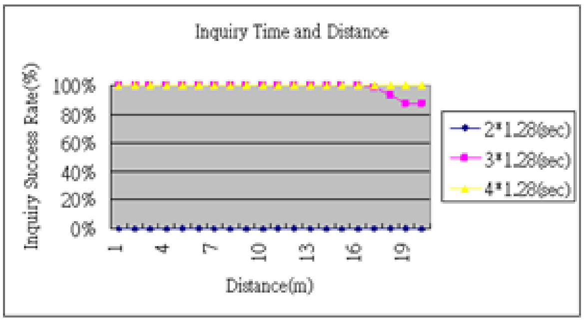

Figure 10 Inquiry time vs distance.

In the connection setup experiment, the master and the slave should do handshake first to switch ID packets and a FHS packet, and then enter the optional security procedures to do authentication and encryption. The average connection setup time is 2.29 seconds. The link establishment time is (5.12+2.29)= 7.41 seconds for the diagnostic host to connect the bridge and the same time for the bridge connect the gateway. So the link establishment time for one bridge node in multiplexing diagnostic system is 14.82 sec.

In Bluetooth, there are six packet types for ACL data transfer. Those packet types are DM1, DM3, DM5, DH1, DH3 and DH5. In Bluetooth scatternet data transfer, an important issue is that DH type packet is interfered. When there are more than two piconets into the same frequency or the environment effect that the error will be occurred. The experiment uses DH1, DH3 and DH5 data packet to transmit 1Kbytes data payload from 1 to 6 piconets. The piconets are overlap in the same area and the distance is 5 meters. We transmit each data packet 100 times for each number of piconets. When the data transmission error occurs the Bluetooth module will respond the error event to the host. The experiment result is shown in Figure 11. The transfer error rate is relative to number of piconets and the type of data packet. We can see that the DH5 packet type transmission error is more than the DH3 packet type, because the DH5 in the sequence transmission slots, the error probability will be higher than DH3 and DH1 when other piconet frequency hop to the same.

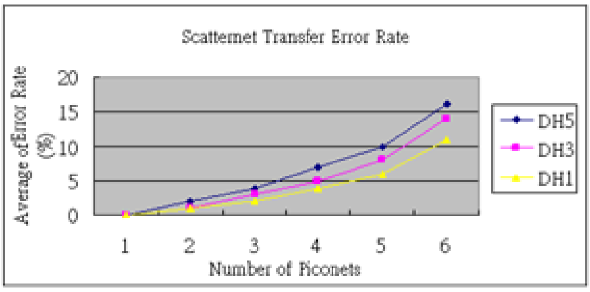

Figure 11 Scatternet data transfer error rate.

For the session-based service transmission, the multi-frame should be defined the value of timeout parameters. In the latency experiment, the host communicates with its Bluetooth module through a UART interface with baud rate 9600. The most part of the latency is transmission delay in serial port and the other is propagating delay in Bluetooth scatternet. In one bridge node of multiplexing diagnostic system, the serial data transmission delay occurs in four segments from diagnostic host to the gateway. Those segments are diagnostic host sending data to Bluetooth module, the bridge node receiving and sending data from Bluetooth module, and the gateway receiving data from Bluetooth module. For each HCI data packet transfer, the packet header will include 4 bytes transmission information. The scatternet propagating delay occurs in two segments. When the sending data has readied by Bluetooth module, the link schedule is into the other piconet. The delay time is the waiting time for the link schedule back to the link of piconet. The maximum delay time is the sniff attempt time to add in sniff timeout time. In the experiment for each type of data packet transmission performance, the 4Kbytes data transmit from the diagnostic host to the gateway. The results are shown in Figure 12. The transfer time is to meet our expectancy. The performance of DH5 data packet transmission is the best.

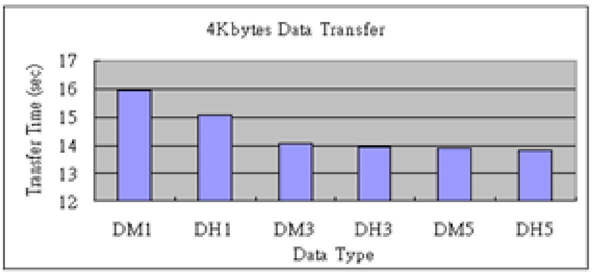

Figure 12 4Kbytes data transmission delay time for each data type.

In the experiment for our multi-frame transmission of multiplexing diagnostic system in network layer, DH1 packet is used to transmit with the length of different data payload. In ISO 14229-2, the length of single frame data payload in CAN bus is 8 bytes. In our Bluetooth scatternet of multiplexing diagnostic system, the single frame includes source BD address, remote BD address, identifier and data payload. The single frame length in our multiplexing diagnostic system should be at least 22 bytes. The ratio between the single frame transmissions with multi-frame transmissions is almost 80% in ISO 14229-2 diagnostic sessions. We use DH1 packet to emulate data transfer time in single frame and multi-frame, as shown in Figure 13 and Figure 14 respectively. When the data length is DH1 data payload length (27 bytes) or a multiple of DH1 data payload length the transfer performance will be better.

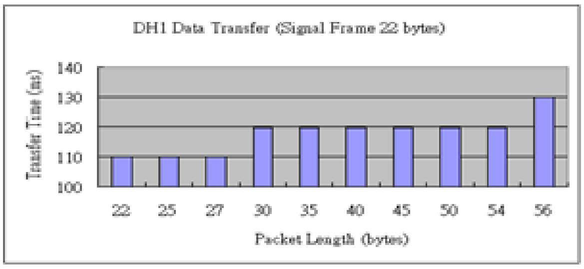

Figure 13 Single frame transfer time vs packet length.

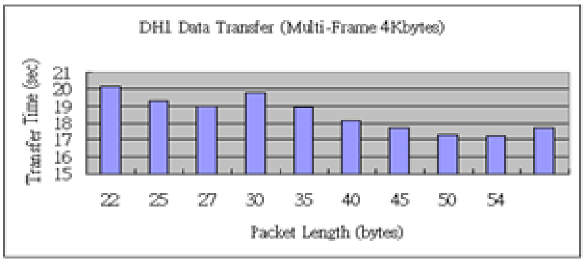

Figure 14 Multi-frame transfer time vs packet length.

For emulation the diagnostic efficiency, we compare with diagnostic methods used the queued and the scattered models, those architectures are shown in Figure 15. For each ECU, we use DH1 packet to transmit 22 bytes single frame 8 times and 80 bytes multi-frame 2 times to emulate the vehicle diagnostic transmission scenario, and compare the transfer time. The result is shown in Figure 16. According to the result, the diagnostic method using scattered model can diagnose vehicles more at the same time and make diagnostic more easily and efficiently.

In those experiment results, the transmission delay time in each type of data packet is shown in Table 1. We can define the transmission timeout parameters to extend from the CAN bus timeout parameters. The values of timeout parameters and the single frame length in each packet type are shown in

Table 2.

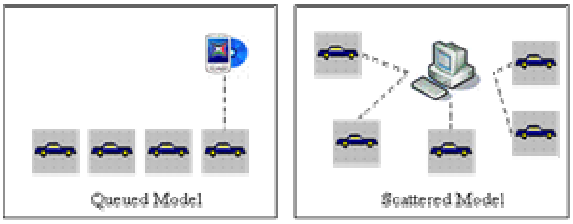

Figure 15 Diagnostic methods.

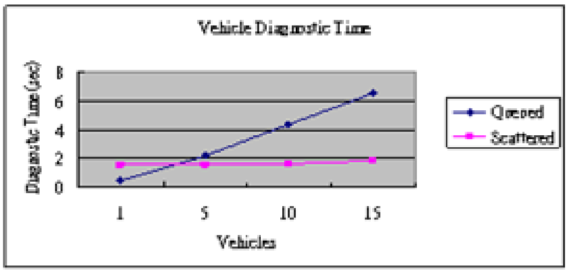

Figure 16 Comparing the diagnostic time of queued model with that of scattered model.

Table 1 The Serial Data Transmission Delay and Scatternet Propagating Delay

Packet Type	Total Bits	Total delay (ms)
DM1	22*8	96.6
DH1	32*8	138.2
DM3	126*8	534.8
DH3	188*8	793.2
DM5	229*8	969
DH5	344*8	1448.2

Table 2 The definition of timeout values and single frame length.

Packet Type	As/Ar	Bs/Br	Cs/Cr	Length
DM1	120ms	170ms	230ms	17
DH1	160ms	210ms	270ms	27
DM3	560ms	610ms	670ms	121
DH3	820ms	870ms	930ms	183
DM5	990ms	1040ms	1100ms	224
DH5	1470ms	1520ms	1580ms	339

We also use the queueing theory to compare with Queued Model and Scattered Model. The Queued Model can be described by *M/M/1/K*. The Scattered Model can be described by *M/M/5/K*. Assume that the number of the packet for the vehicle diagnostic transmission has transmitted 22 bytes single frame 4800 times and 80 bytes multi-frame 1200 times. In Queued Model, we assume an automobile emission inspection station with one inspection stall and the inspection stall is just for one car. The station can accommodate at most K-1 cars waiting (K in the station) at one time. The service for the vehicle diagnostic transmission is exponential with mean 260 seconds. In Scattered Model, we assume an automobile emission inspection station with one inspection stall and the inspection stall is just for five cars. The station can accommodate at most K-5 cars waiting (K in the station) at one time. The service for the vehicle diagnostic transmission is exponential with mean 246 seconds. From equation $L = L_q + r(1 - p_k)$, we can get the average number of cars in the system during peak periods. From equation $W = \frac{L}{\lambda_{eff}} = \frac{L}{\lambda(1 - p_K)}$, we can get the average waiting time (including service) . The two model with different arrival rates λ (λ from 0.1 to 0.2, every 0.01 is unit one) and K=10 to compute the L and W, the result is shown in Figure 17 and Figure 18.

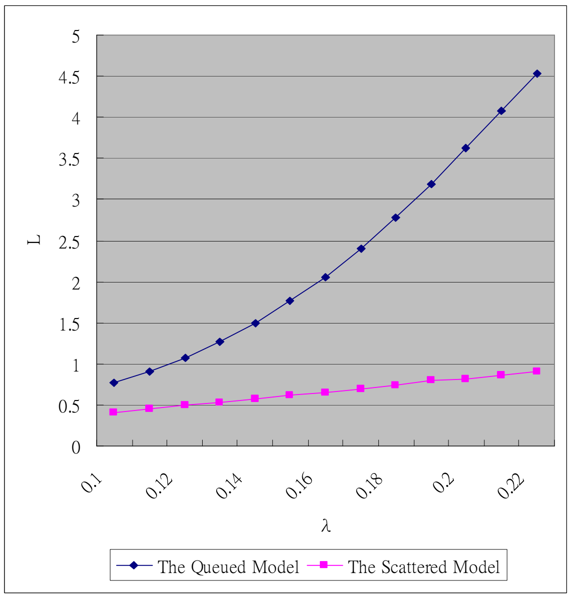

Figure 17 The relation between L and λ for the Queued Model and the Scattered Model is illustrated.

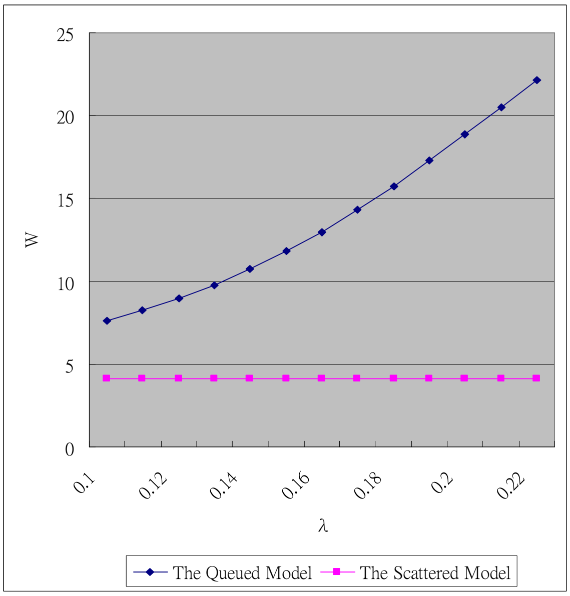

Figure 18 The relation between W and λ for the Queued Model and the Scattered Model.

7. Conclusions and Future Works

In this paper, a multiplexing diagnostic system based on Bluetooth scatternet is proposed to use in the automobile service department. Our prototype of multiplexing diagnostic system is implemented to emulate the transmitting scenario between CAN bus and Bluetooth scatternet. The experiment results offer the time of connection set up, each type of data packet transmission latency and how to define the optimal data frame length. The serial data transmission delay is the major transmission delay in Bluetooth scatternet. Using higher baud rate in a serial port the transmission delay time can be reduced. In the diagnostic sessions, an optimal protocol is selected to increase transmission efficiency in scatternet. The suitable data packet length and data packet type can make the transmission to have high performance. The experiment results are rather smooth according to our requirement. Thus, the proposed architecture is proved to be feasible. Hence, the mechanic can maintain vehicles conveniently and thoroughly by this system. This communication architecture also satisfies the requirement of ITS services in RVC, IVC, and TVC.

As future work, we will develop other applications in Bluetooth scatternet based on this architecture, such as Bluetooth home gateway, and Bluetooth gateway for personal health care. For intelligent vehicle development, the real time diagnostic requirement can use the agent to diagnose in the safety situation and use on road communication technology to offer the real time information for the vehicle.

Acknowledgements

The authors would like to thank the National Science Council of R.O.C. for financially supporting this research under Contract NSC-94-2745-E-252-004-URD.

References

[1]Jorge A.M.POLAR S., Dulciane S. SILVA, Andre L. FORTUNATO, Luiz A.C. ALMEIDA, and Carlos A. DOS REIS Filho., "Bluetooth Sensor Network for Remote Diagnostics in Vehicles", IEEE International Symposium on Industrial Electronics, 2003.

[2]Bilstrup, "The usability of Bluetooth in Vehicles", technical report at CCA Halmstad University, July, 2000.

[3]U. Bilstrup, "Applicability of Bluetooth Applications in Vehicles", speech at Bluetooth Developers Conference, US, San Jose, December, 2000.

[4]Y. Chen and L. Chen, "Using Bluetooth wireless technology in vehicles", IEEE International Conference on Vehicular Electronics and Safety, pp. 344-347, 2005.

[5]F. Cuomo, G. D. Bacco. and T. Melodia, "Optimized Scatternet Topologies for Personal Area Networking in Dynamic Environments", Proc. IEEE International Conf. on Communications, Vol. 6, pp. 3696-3700, June 2004.

[6]K. Etschberger, "Controller Area Network: Basics, Protocols, Chips and Applications", IXXAT Automation GmbH, Weingarten, Germany, 2001.

[7]P. Enstrom and E. Svedback, "Road Vehicle Diagnostics Using Bluetooth", Master Thesis, Department of Computer Science and Electrical Engineering, Lulea University of Technology, 2003.

[8]J. A. Gutierrez, M. Naeve, E. Callaway, M. Bourgeois, V. Mitter and B. Heile, "IEEE 802.15.4: a Developing Standard for Low-Power Low-Cost Wireless Personal Area Networks", Proc. IEEE Network, pp. 12-19, September 2001.

[9]Donald Gross and Carl M. Harris, "Fundamentals of Queueing Theory", Third edition, pp. 74-80, 1998.

[10]L. E. Hodge and R. M Whitaker, "What Are the Characteristics of Optimal Bluetooth Scatternets?", Proc. MOBIQUITOUS of Mobile and Ubiquitous

Systems: Networking and Services, pp. 124-125, August 2004.

[11]S. H. Hong and W. H. Kim, "Bandwidth Allocation Scheme in CAN Protocol", Proceedings of the IEE Control Theory and Applications, pp.37-44, January 2000.

[12]P. Johansson, M. Kazantzidis, R. Kapoor and M. Gerla, "Bluetooth: An Enabler for Personal Area Networking", Proc. IEEE Network, pp. 28-37, October 2001.

[13]D. Johnson, "Hardware and Software implications of Creating Bluetooth Scatternet Devices", Proc. 7th AFRICON Conf., Vol. 1, pp. 211-215, September 2004.

[14]R. Kuchta and R. Vrba, "Measuring and Monitoring System for Electric Cars", Proc. 4th IEEE International Symposium on Diagnostic for Electric Machines, Power Electronics and Drivers, pp. 342-344, August 2003.

[15]G. Leen and D. Heffernan, "Vehicles Without Wires", Computing & Control Engineering Journal, pp. 205-211, Vol. 12, Issue 5, 2001.

[16]C. M. Lin, C. R. Dow, T. W. Ko, C. M. Chiu and K. L. Liao, "BlueTree++: A Novel Scatternet Formation Scheme", Proc. 23rd IASTED International Conf. on Parallel and Distributed Computing and Networks, pp. 59-63, February 2005.

[17]S. mer, P. K. Chande and P. C.Sharma, "Automobile Engine Fault Diagnosis Using Neural Network", in Proceedings of IEEE Intelligent Transportation Systems Conference, pp.492-495, 2001.

[18]J. Misic and V. B. Misic, "Adaptive Management of Bluetooth Master/slave Bridge", Proc. IEEE Internationals Conf. on Performance, Computing, and Communications, pp. 199-206, April 2003.

[19]H. Sawant, J. Tan, Q. Yang, and Q. Wang, "Using Bluetooth and Sensor Networks for Intelligent Transportation Systems", in proceedings of the 7th International IEEE Conference on Intelligent Transportation Systems, pp. 767-772, 2004.

[20]K. P. Shih, S. S. Wang and J. H. Su, "A Bluetooth Group-scatternet Formation Algorithm for Efficient Routing", Proc. International Conf. on Parallel Processing Workshops, pp. 50-56, October 2003.

[21]A. Shrivastava and R. Rajamani, "Fault Diagnostics for GPS-Based Lateral Vehicle Control", in Proceedings of the American Control Conference, Vol.1, pp. 31-36, 2001.

[22]S. C. Subramanian, S. Darbha, and K. R. Rajagopal, "A Diagnostic System for Air Brakes in Commercial Vehicles", IEEE Transactions on Intelligent Transportation System, pp. 360-376, Vol. 7, No. 3,2006.

[23]Bluetooth Special Interest Group, "Specification of the Bluetooth System", Ver. 2.0+EDR, Specification Vol. 0-3, November, 2004.

[24]Bluetooth Special Interest Group, "Scatter Mode, PAN IPS Whitepaper", Ver. 0.25, October 2001.

[25]Bluetooth SIG Radio WG, "Improvement Proposal for Scatternets", BT PAN IPS, Ver. 0. 46, April 2002.

[26]Bosch CAN, "Specification 2.0", Germany, Bosch, September 1991.

[27]CSR, "Scatternet Support", August 2005.

[28]ISO-15765, "Road Vehicles-Diagnostics on CAN", Part. 1-4, November 1999.

[29]ISO-14229, "Road Vehicles-Diagnostic System" Part. 1-3, May 2001.

[30]SAE-J1979, "E/E Diagnostic Test Modes - Equivalent to ISO/DIS 15031-5", April 30, 2002.

[31]http://www.csr.com.

[32]http://www.advantech.com/products/.

[33]http://www.vector-cantech.com/index.html.

[34]http://www.canalyser.com.