An App Visualization design based on IoT Self-diagnosis Micro Control Unit for car accident prevention

YiNa Jeong[1], EunHee Jeong[2] and ByungKwan Lee[1]
[1]Department of Computer Engineering, Catholic Kwandong University
Gangneung-si, Gangwon-do, 25601 South Korea
[e-mail: bklee@cku.ac.kr, lupinus07@nate.com]
[2] Department of Regional Economics, Kangwon National University
Samcheok-si Gangwon-do, 25913 South Korea
[e-mail: jeongeh@kangwon.ac.kr]
*Corresponding author: ByungKwan Lee

Received September 7, 2016; revised December 5, 2016; accepted December 20, 2016; published February 28, 2017

Abstract

This paper proposes an App Visualization (AppV) based on IoT Self-diagnosis Micro Control Unit (ISMCU) for accident prevention. It collects a current status of a vehicle through a sensor, visualizes it on a smart phone and prevents vehicles from accident. The AppV consists of 5 components. First, a Sensor Layer (SL) judges noxious gas from a current vehicle and a driver's driving habit by collecting data from various sensors such as an Accelerator Position Sensor, an O2 sensor, an Oil Pressure Sensor, etc. and computing the concentration of the CO collected by a semiconductor gas sensor. Second, a Wireless Sensor Communication Layer (WSCL) supports Zigbee, Wi-Fi, and Bluetooth protocol so that it may transfer the sensor data collected in the SL to ISMCU and the data in the ISMCU to a Mobile. Third, an ISMCU integrates the transferred sensor information and transfers the integrated result to a Mobile. Fourth, a Mobile App Block Programming Tool (MABPT) is an independent App generation tool that changes to visual data just the vehicle information which drivers want from a smart phone. Fifth, an Embedded Module (EM) records the data collected through a Smart Phone real time in a Cloud Server. Therefore, because the AppV checks a vehicle' fault and bad driving habits that are not known from sensors and performs self-diagnosis through a mobile, it can reduce time and cost spending on accidents caused by a vehicle's fault and noxious gas emitted to the outside.

Keywords: Self-diagnosis, sensor modeling, APP design, Eco-drive, sensor network

A preliminary version of this paper was presented at APIC-IST 2016, and was selected as an outstanding paper.
This work was supported by research fund of Catholic Kwandong University (CKURF-201601180001).

https://doi.org/10.3837/tiis.2017.02.020 **ISSN : 1976-7277**

1. Introduction

A today's society is becoming the IoT-oriented world because of the development of network technology. Various application services are being developed because of IoT technology. An Intelligent Transport System (ITS) and a U-car System are good examples of typical application on IoT. The IoT service that is applied to these vehicles provides drivers with such convenient information as car accidents, failure detection sensing, traffic situation, and course measurement in real time. An IoT service that an existing vehicle owns provides just a limited service by using PC, a b Server, and a built-in device, but nowadays services was developed to communicate the systems within a vehicle and a handheld terminal like a smart phone and PDA anytime and anywhere.

IoT Service of vehicles is in spotlight as a Eco-driving system not only because it provides drivers with convenient information but also because it prevents the damages caused by vehicles. An Eco-driving system has a target that it prevents air pollution caused by noxious gas that is emitted from vehicles and reduces fuel consumption and cost.

This paper proposes an App Visualization (AppV) based on IoT Self-diagnosis Micro Control Unit for accident prevention that collects a current status of a vehicle through a sensor, visualizes it on a smart phone and prevents vehicles from accident. Therefore, because the AppV checks a vehicle' fault and bad driving habits that are not known from sensors, and performs self-diagnosis through a mobile. it can reduce the time and cost spending on accidents caused by a vehicle's fault and the noxious gas emitted to the outside.

The remainder of this paper is organized as follows. Section 2 discusses the related works. Section 3 proposes an App Visulalization design based on IoT Self-diagnosis Micro Control Unit for car accident prevention. Section 4 analyzes and estimates its performance. In the Section 5, our conclusion is described.

2. Related Work

2.1 A Vehicle Sensor

Sensors have been utilized in several technologies and one of them is vehicles. In [1], they used sensors in the vehicle to sense another vehicle in front of the vehicle by using a single monochrome camera. While [2] discuss about UWB (ultra-wideband) in wireless sensor network for vehicles. It proved UWB improved communication between each of vehicle's components with stable signal and high exchange data rate.

In [3] proposed utilizing GPS and vehicle sensors for land vehicle navigation. GPS provided the position of the vehicle, while vehicle sensors provided several data such as: velocity, attitude information, etc. While in [4] utilized sonar sensor and image processing for vehicle detection. Sonar sensor made detecting other vehicles possible despite the lack of lightning of surrounding vehicle. They proposed to sense the light condition before deciding which method that can be used to detect other vehicles.

In [5], a fuzzy method to improve the performance in the mobile clients of the real-time system was proposed. The Vitruvius is used as the real system platform. The Vitruvius consisted of a wide range of sensors for vehicle, such as: speed, engine coolant temperature, air pressure, and etc.

In [6], a method for vehicle tracking on roadways using magnetometers and accelerometers sensors are proposed. The proposed method used multi-rate particle filtering which fuses measurements from the two different sensors. The measurements are handled using Rao-Blackwellization. In [7], an accurate vehicular positioning system which can achieve lane-level performance in urban canyons was proposed. The system used Global Navigation Satellite System (GNSS) receivers, onboard cameras and intertial sensors. The system used a novel GNSS positioning technique with 3D building map which reduced multipath effect and Non-Line-Of-Sight (NLOS) effects.

2.2 Vehicle Diagnosis System

In [8] developed a fuzzy logic based diagnosis system and applied it for a fault detection and isolation. There are some faults that found, such as: sensor disconnection, actuator breakdown, local loop problems and biological process problem. While in [9] develop a low-cost vehicle monitoring system. They retrieved the information to a mobile device from sensor nodes using Bluetooth, the collected data can be used in engine control unit (ECU) and vehicle's built in on-board diagnostics system (OBD). The information can be viewed by the user locally.

In [10] implemented a diagnosis system for the air-path of a turbo-charged diesel engine with EGR in a real car and real road environment. The diagnosis system purpose is to reduce the probability of incorrect isolation. While in [11] evaluated the advantage of vehicle dynamics modeling and implemented it to a steer-by-wire diagnostic system design. The vehicle dynamics modeling reduce the noise input to the steer-by-wire diagnostic system. It also improve the speed and accuracy of the diagnoses.

In [12] the fault diagnosis for vehicle lateral dynamics are investigated. It also proposed a fault diagnosis system which consist of two sub-systems: fault diagnosis observer sub-system and robust threshold subsystem. The proposed system is applied to the accelerometer and gyrometer to diagnose the problem of vehicle lateral dynamics.

In [13] the performance of the Consensus Self-Organizing Models (COSMO) approach for automatic detection of air pressure realted faults on a fleet of city buses is evaluated. The COSMO is a genetic unsupervised deviation detection method. The method used in the evaluation: generic and robust.

3. An App Visualization design

An App Visualization (AppV) based on the ISMCU in this paper consists of a Sensor Layer (SL), a Wireless Sensor Communication Layer (WSCL), an IoT Self-diagnosis Micro Control Unit (ISMCU), a Mobile App Block Programming Tool (MABPT) and an Embedded Module (EM) as shown in **Fig. 1**.

The SL collects the sensor data and CO concentration data measured from the sensors of vehicles that says a current status of vehicles. The WSCL supports a Zigbee, Wi-fi, Bluetooth protocol to connect sensors, smart phones, and cloud server. The ISMCU integrates the received sensor information and transfers the result to a Mobile. The MABPT is a independent App generation tool that changes to visual data just the vehicle information which drivers wants from a smart phone. The EM records the data collected through a Smart Phone real time in a Cloud Server.

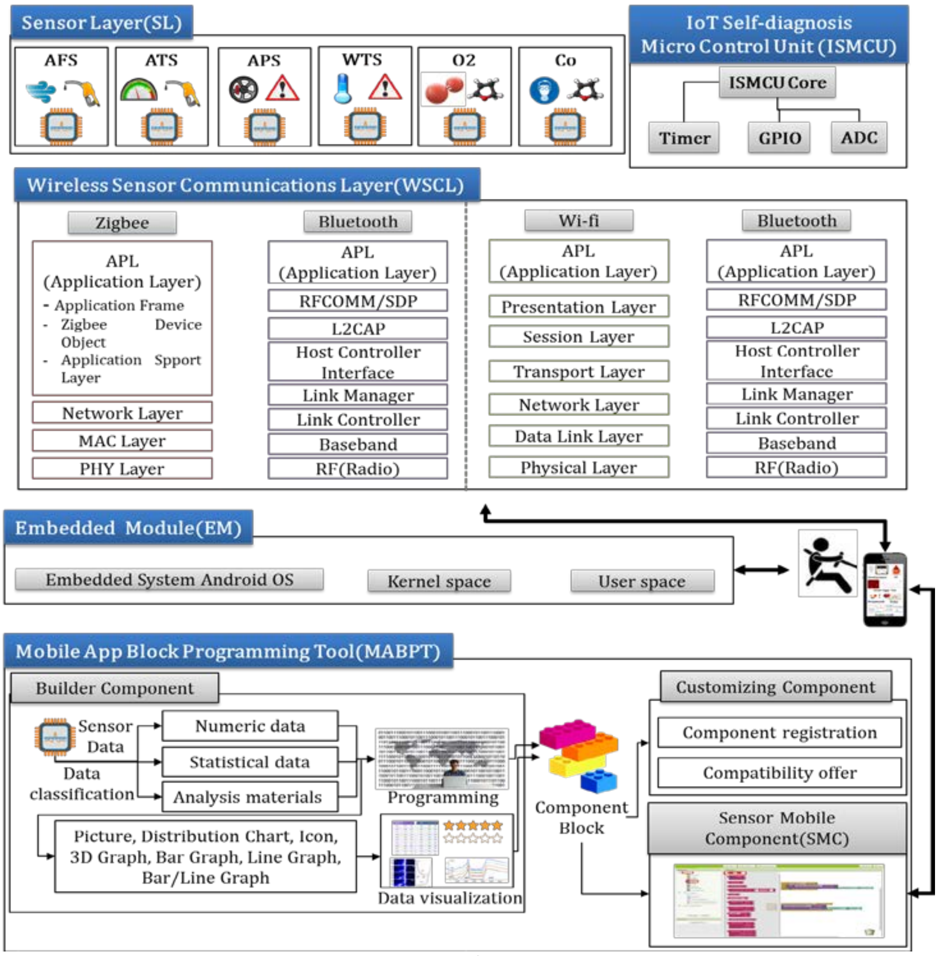

Fig. 1. The Structure of AppV based on ISMCU

3.1 A design of a Sensor Layer (SL)

The SL collects the current status information of vehicles by using various sensors. The sensors that the SL needs can always be changed according to the types of vehicles and the kinds of the collected data. This Paper transfers to the ISMCU the data collected by the Air Pressure Sensor(APS) and Air Temperature Sensor(ATS) that decides basic fuel injection quantity, the Accel Position Sensor(APS) which senses a failure with the precision of pedal action, the Water Temperature Sensor(WTS) measuring collant temperature, the Oxygen Sensor(O2) delivering sudden changed output voltage by measuring the quantity of oxygen contained in car exhaust, the Oil Pressure Sensor(OPS) sensing whether or not engine oil is normal and the Sensor Layer Gas Sensor(SLGS) computing the concentration of CO contained in car exhaust.

In particular, the Sensor Layer Gas Sensor(SLGS) proposed in the SL is based on Micro Electro Mechanical System (MEMS). The SLGS improves CO sensitivity by adding a V2O5 catalyst to SnO2. Because the SLGS reacts to high temperature, pressure and humidity, it has double structure protection cap so as to reduce an error rate. Recently, such material as SnO2, TiO2 and, ZnO has been used as gas-sensing material. Just With these sensing-material, it is difficult to extract the information of special material.

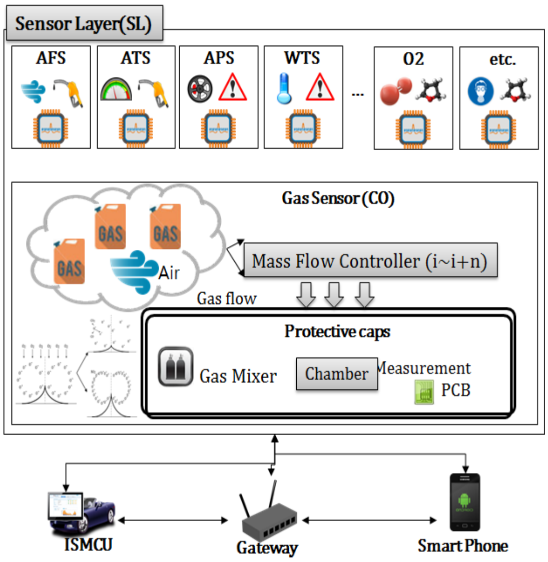

Fig. 2. The Structure of SL

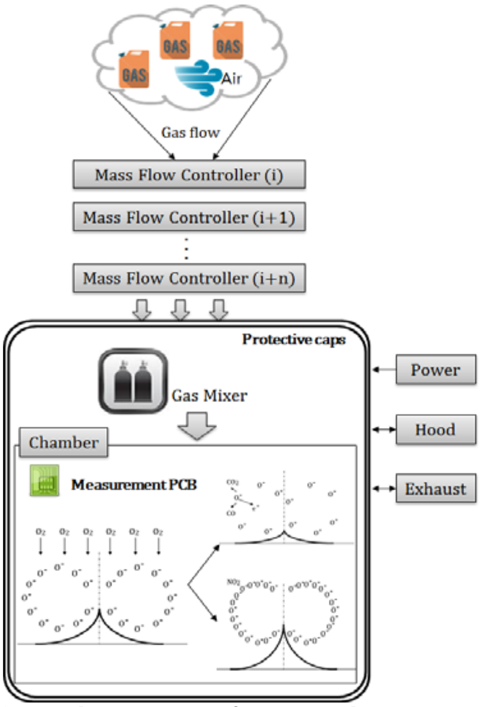

Fig. 3. The Structure of SLGS in the SL

The structure of the SLGS is shown in **Fig. 3**. It adds V2O5 catalyst to help the sensitive reaction and sensing strength of CO. The SLGS consists of a power supply, Data AQquisition (DAQ) Board, Chamber, a Metering valve, a Bomb about CO gas, a Mass Flow Controller(MFC), and PCB. A vehicle's internal temperature that the SLGS knows is decided by an internal temperature sensor and is protected by an external dual cap to minimize an influence on external temperature. The MFC adjusts the quantity to measure gas concentration that flows into a Chamber and the PCB measures gas resistance by voltage of both ends of a sensor.

The micro sensor based on MEMS used in the SLGS has an advantage that it is subminiature and ultra-light and its production cost is low. In addition, the SLGS based on MEMS has a protective cap in the outside of the sensor. The protective cap is less influenced by external temperature and pressure and computes the CO concentration accurately.

3.2 A design of a Wireless Sensor Communication Layer (WSCL)

The WSCL supports the interworking between Sensors and ISMCU and the interworking between ISMUC and a smart Phone. The network among the SL, the ISMCU, and smart Phones and the flowchart of WSCL are shown in **Fig. 4**. All the connected sensors in the SL supports Zigbee or Bluetooth protocol basically. The WSCL consists of the devices as shown in **Fig. 5** and transfers sensor data to the ISMCU according to the flowchart in **Fig. 4**.

The WSCL supports the internetworking between a Smart phone and ISMCU by changing the data of Zigbee and Bluetooth protocol supported by ISMCU to the data of Wi-Fi and Bluetooth protocol supported by a smart phone.

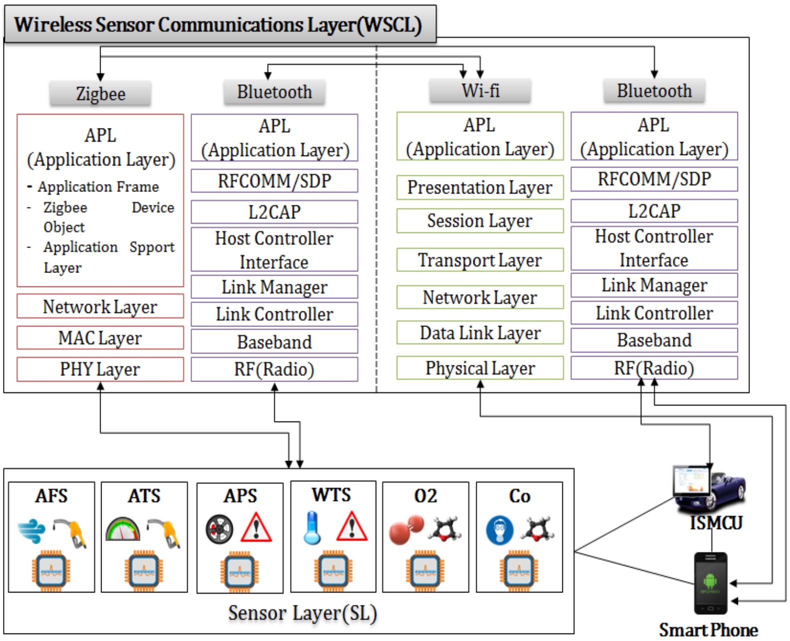

Fig. 4. The Flowchart of a Wireless Sensor Communications Layer(WSCL)

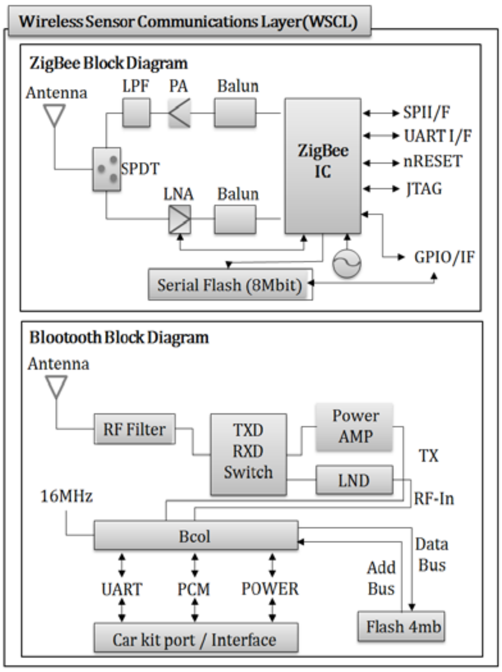

Fig. 5. The structure of Zigbee and Bluetooth in the WSCL

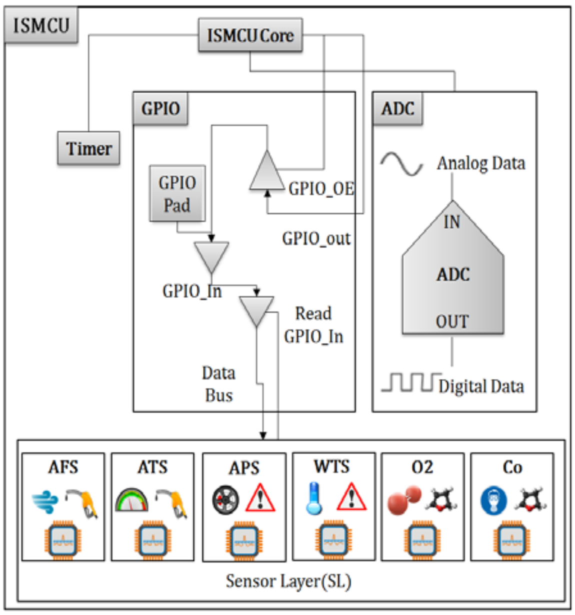

Fig. 6. The structure of the ISMCU

3.3 A design of IoT Self-diagnosis Micro Control Unit (ISMCU)

The structure of the ISMCU is shown in **Fig. 6**. The ISMCU has the following functions.

First, the ISMCU receives the sensor data collected by the SL by using General Purpose Inout/Output(GPIO) port. The GPIO supports the unification to exchange data with various sensors.

Second, the ISMCU changes continuous analogue sensor data like temperature and voltage to digital signal by using Analog to Digital Conversion(ADC). The digital signal is used for a processor as data.

Third, the ISMCU has a Sensing Data Timer(SDT) and generates integrated sensing messages about the simultaneously transferred data based on the SDT.

The sensing messages generated bt the ISMCU are transferred to drivers' smart phone through WSCL.

3.4 Mobile App Block Programming Tool (MABPT)

The MABPT proposed in this paper is a independent App generation tool that changes to visual data just the vehicle information which drivers wants from a smart phone. It consists of a Builder Component, a Customizing Component and a Sensor Mobile Component as shown in **Fig. 7**.

1) A Builder Component

The Builder Component is made up of Design mode and Block mode. The block mode is a phase that changes various sensor data to types of visual information and generates components with the program blocks implemented beforehand. The components can easily be assembled by App by using drag and drop. The block mode is processed as follows.

First, the Builder Component receives data from sensors and classifies it into numeric data, the statistical data about an fixed time, and the analysis data compared with normal data.

Second, the Builder Component generates each program slice about the classified sensor data. The program slice becomes a program block with the combination of the visual data of design mode.

Third, all program blocks becomes components each. and an APP is generated by selecting and assembling components according to App generator.

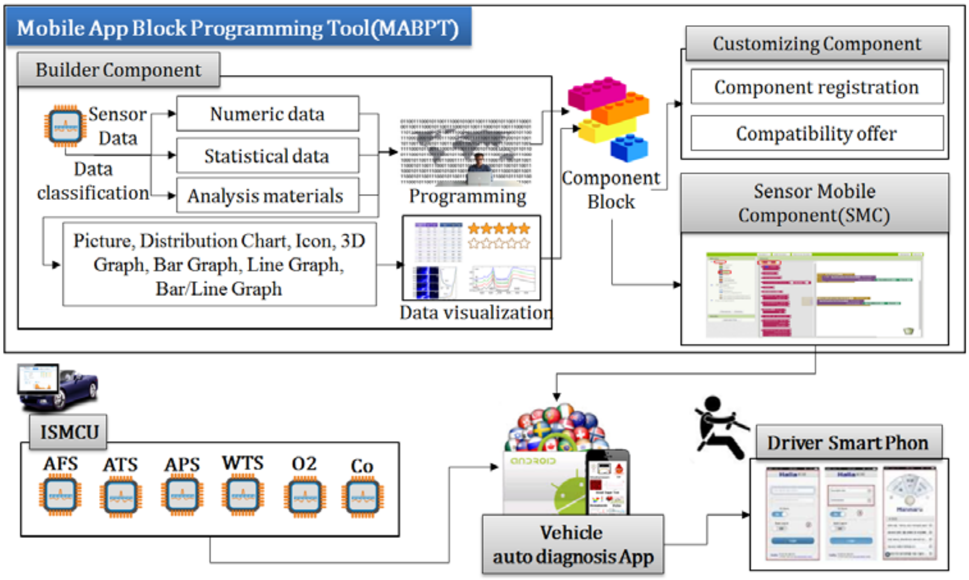

Fig. 7. Mobile App Block Programming Tool(MABPT)

The design mode provides the visual data about sensor data and decides where it will locate the visual data within an APP. The design mode is processed as follows.

First, the design mode generates a component block translating sensor data into chart, graph, image, character, numeric, and graphic.

Second, the visualization by an App in the design mode is classified into Picture, Distribution Chart, Icon, 3D Graph, Bar Graph, Line Graph, and Bar&Line Graph.

Third, an App generator selects information with an App and generates vehicle IoT Self-diagnosis App.

2) A Customizing Component

A Customizing Component is a component register guaranteeing the compatibility with external components and processing a specific area. It is used to register a newly generated component, not a component generated in the MABPT. Each component registered in the MABPT is made based on a development standards and the variable components developed according to this standards must be registered newly in the MABPT. If a new component is generated, it is registered and updated in the MABPT newly after validating its compatibility. The updated component also consists of a block and it is registered in the Builder Component.

3) A Sensor Mobile Component

A Sensor Mobile Component does its role to transfer actual data to drivers. It transfers to vehicle IoT Self-diagnosis APP generated by the Builder Component and the Customizing Component the real time integrated sensor data which was delivered from the ISMCU and provides drivers with the interface about current state of vehicles.

At this time, to transfer to the IoT Self-diagnosis APP the data which was delivered from the ISMCU, it is stored in the DB inside a Smart Phone temporarily. Because it is not stored in the DB of a Smart Phone forever, it will most likely lose former data when drivers compares real time information to former information.

To solve this problem, the Sensor Mobile Component transfers the real time data written in the DB temporarily to a Cloud server according to the network standards of the WSCL.

3.5 A design of Embedded Module (EM)

A vehicle IoT Self-diagnosis system proposed in this paper not only collects the current status data of vehicles but also provides users with an interface about state data through App. At this time, in order to monitor the sensor data collected by sensors, it constructs a server to collect data based on the OS inside a system.

The EM uses a buffer as a temporary server of a smart phone and constructs an Android Server for vehicles to write to a Cloud Server the data collected by a smart phone. The EM processes Linux and Android porting as shown in **Fig. 8**.

The Linux and Android Porting process of the EM is done as follows.

First, The EM embedded in a vehicle uses a Web Page based Client Pull method to transfer the sensor data collected in real time to a Smart Phone and a Client Server repeatedly.

Second, the EM uses Java language to provide compatability between sensors and an Android OS based App Platform.

Third, because the data that the EM transferred to a Smart Phone disappears after a fixed time or in case of the overflow of its buffer, all the data that the EM collected are transferred to the Cloud Server at regular intervals.

Fourth, the data of the Cloud Server uses a Cross Compiler so that it is executed on an Android Platform.

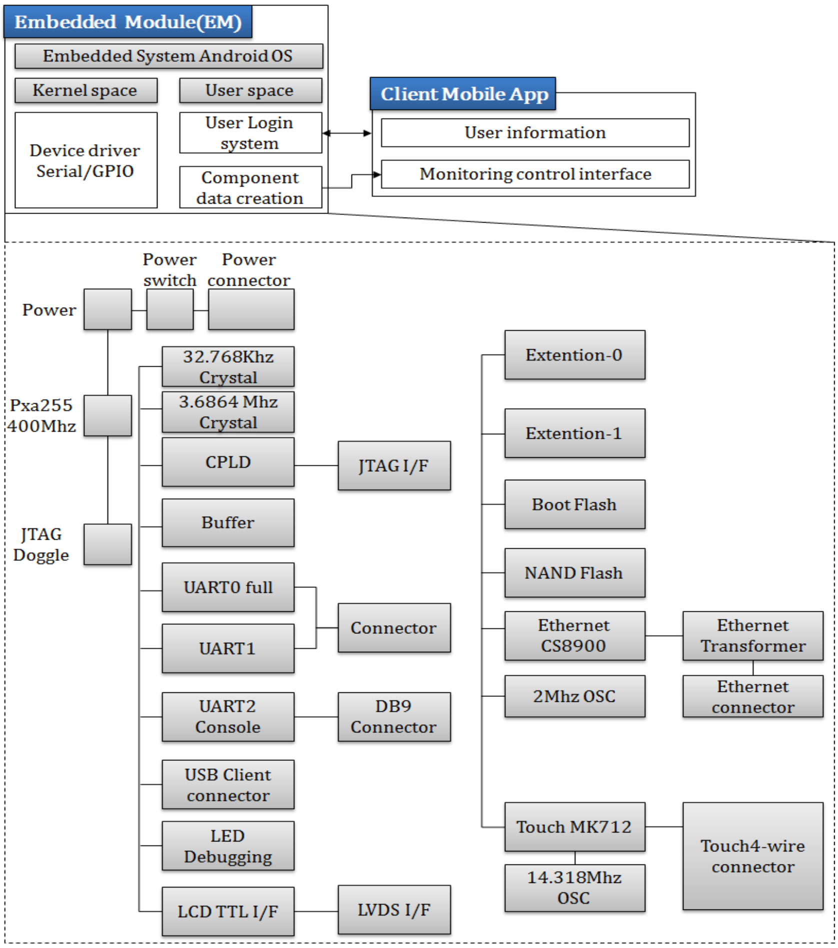

Fig. 8. The structure of the Embedded Module(EM)

4. A Performance Analysis

4.1 The performance analysis of the SL

The output value of a CO sensor using just pure SnO_2 and that of a CO sensor using both pure SnO_2 and catalyst V_2O_5 are compared in this section. The CO concentration used in this experiment is 130ppm. The temperature used in sensitivity measurement is between 200℃ and 500℃and is compared by every 50 interval. The resistance and sensitivity value based on the catalyst V_2O_5 are measured on the basis of 300℃. The output through LabView is shown as follows.

Fig. 9 shows the experimental result about CO sensitivity when the CO concentration is fixed as 130ppm and the temperature is variable. Here, the x-axis means temperature, and the first y-axis and the second y-axis mean Co sensitivity. The comparison between pure SnO_2 and $V_2O_5+SnO_2$ shows that $V_2O_5+SnO_2$ has higher CO sensitivity than pure SnO_2 between

200℃ and 600℃ under the same concentration environment. Pure SnO2 has the highest sensitivity of 45 between 400 ℃and 450℃ and V2O5+SnO2 has the highest sensitivity of 63 in 500℃. In addition, the difference of sensitivity between V2O5+SnO2 and pure SnO2 is minimum 3 and maximum 23. Because vehicles have higher temperature, a Gas Sensor using V2O5+SnO2 is more profitable than that using pure SnO2.

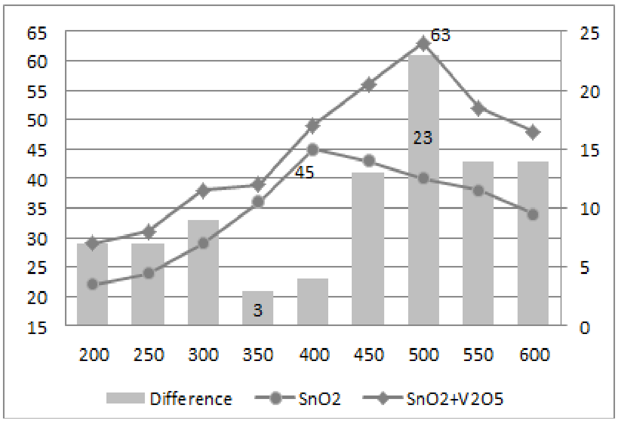

Fig. 9. the difference of CO sensitivity between pure SnO2 and V2O5+SnO2 according to variable temperature

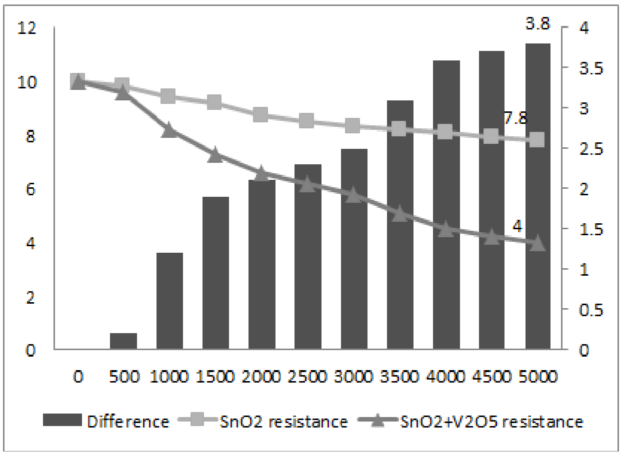

Fig. 10. The comparison of resistance value according to the change of CO gas concentration

Fig. 10 shows that the resistance value between pure SnO2 and V2O5+SnO2 was compared when the CO concentration is between 500 and 5000 and the temperature is 300℃. Here, the x-axis means Co concentration, and the first y-axis and the second y-axis mean the resistance value. According to the experimental result, the resistance value of pure SnO2 shows 7.8 through 9.83 and that of V2O+SnO2 shows 4,0~ 9.6. the difference of the resistance value between pure SnO2 and V2O5+SnO2 is 0.23~ 3.8.

Therefore, pure SnO2 has the average resistance value of 8.718, V2O5+SnO2 has the average resistance value of 6.5. V2O5+SnO2 has a decreasing effect on resistance value by 1.34 times, compared with pure SnO2.

4.2 The performance analysis of ISMCU

The throughput is compared when the existing MCU and the proposed ISMCU receives data from sensors by using different communication environments. The following conditions are used in this experiment.

First, data is transferred through 5 Zigbee sensors and 5 Bluetooth sensors.

Second, the information generation of each sensor is done in the interval of 0.05 second and the delay about information generation is not considered.

Third, the experiment is done 3 times in the interval of 300sec and the data of the previous experiment is not initialized when comparing the throughput of the generated data.

Fig. 11 shows the generated information from Zigbee and Bluetooth sensors used for this experiment and the throughput without information being integrated.

Here, the x-axis means the information generated and throughput through a Zigbee and a Bluetooth sensor, and the y-axis means the number of data during the 1st, 2nd, and 3rd experimental process.

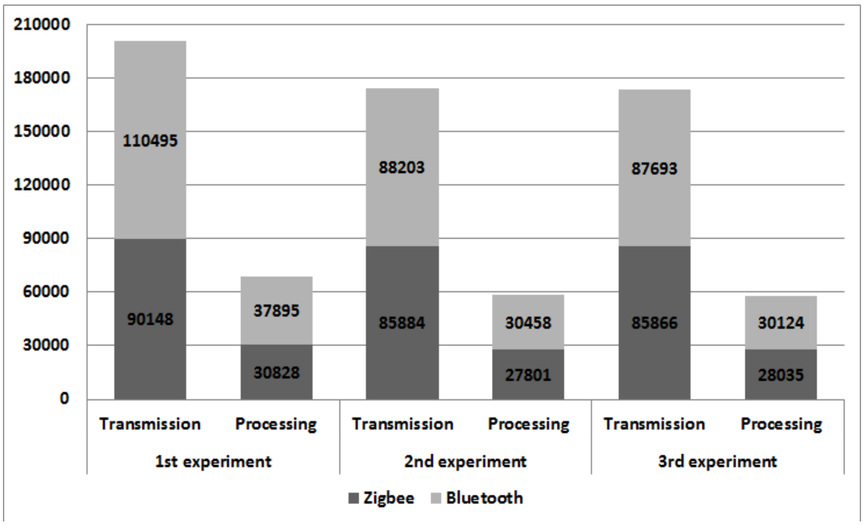

Fig. 11. The information generation and throughput of Zigbee and Bluetooth sensors used for this experiment

When the existing MCU receives data from different communication environments, it receives data through the input and output processors suitable for each communication environment and integrates it again. On the other hand, because the proposed ISMCU reads data by using GPIO and WSCL after integrating the data format of different communication protocols. it reduces the accumulative job amount and reads more sensor information within the same hours. **Fig. 12** shows the experimental result of the MCU and ISMCU. The experiment is done 3 times in the interval of 300sec.

1)The 1st experiment

The MCU generated 200,643 sensor data and the 66,828 sensor data among them(That is, its throughput is 0.333%) are processed. On the other hand, the ISMCU generated 198,949 sensor data and the 63,768 sensor data among them(that is, its throughput is 0.320%) are processed.

If the 1st experiment is finished, the experimental environment of the 2nd experiment is changed as follows because of the 1st result. Let's consider the 2nd experiment. First, the experiment is proceeded between 300sec and 600sec. Second, the data that was not processed in the 1st experiment must be included in the 2nd experiment.

Therefore, the MCU must process the data including 133,815(200,643–66,828) data that was not processed in the 1st experiment, and the ISMCU must process the data including 135,181(198,949–63,768) data.

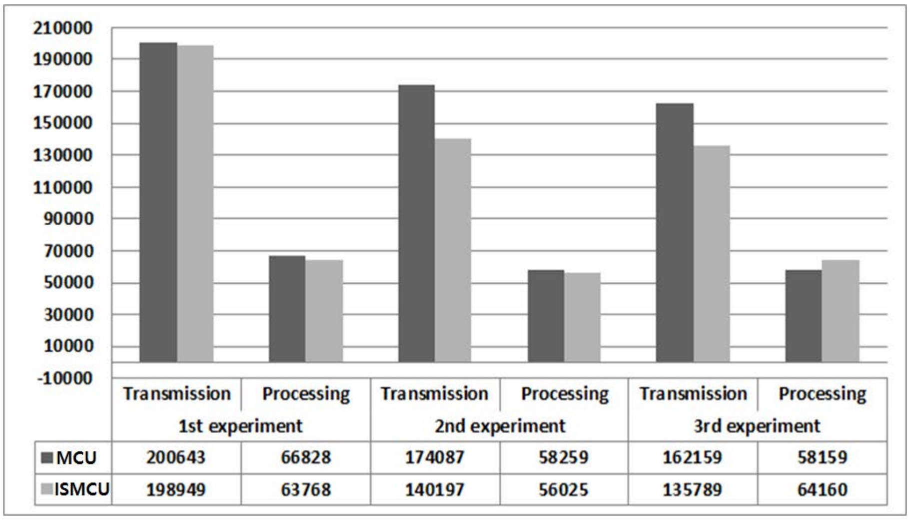

Fig. 12. The comparison on information generation and data throughput of the MCU and the ISMCU.

2) The 2nd experiment

The MCU must process the unprocessed 133,815 and the newly generated 174,087 data. That is, the MCU has 0.334% throughput because it processes just the 58,259 data among accumulated 307,902 (133,815+174,087) data.

The ISMCU must process the unprocessed 135,181 and the newly generated 140,197 data. That is, the ISMCU has 0.36% throughput because it processes just the 56,025 data among accumulated 275,378 (135,181+140,197) data.

If the 2nd experiment is finished, the experimental environment of the 3rd experiment is changed as follows because of the 2nd result. Let's consider the 3rd experiment. First, the 3rd experiment is proceeded between 600sec and 900sec. Second, the data that was not processed in the 2nd experiment must be included in the 3rd experiment.

Therefore, the MCU must process the data including 249,643(307,902-58,259) data that was not processed in the 2nd experiment, and the ISMCU must process the data including 219,353(135,181-140,197) data.

3) The 3rd experiment

The MCU must process the unprocessed 249,643 and the newly generated 162,159 data. That is, the MCU has 0.342% throughput because it processes just the 58,159 data among accumulated 411,802(249,643+162,159) data.

The ISMCU must process the unprocessed 219,353 and the newly generated 135,789 data.

That is, the ISMCU has 0.396% throughput because it processes just the 64,160 data among accumulated 355,142(219,353+135,789) data.

Therefore, when the existing MCU and the ISMCU are compared, this paper shows that the accumulated throughput of real time sensor data was increased by about 0.054%.

5. Conclusion

This paper proposes an App Visualization (AppV) based on IoT Self-diagnosis Micro Control Unit (ISMCU) for accident prevention. An App Visualization (AppV) based on the ISMCU consists of a Sensor Layer (SL), a Wireless Sensor Communication Layer (WSCL), an IoT Self-diagnosis Micro Control Unit (ISMCU), a Mobile App Block Programming Tool (MABPT) and an Embedded Module (EM). Its efficiency is as follows.

First, sensors realizes that a vehicle has its own fault or a driver's habit is bad and judge the noxious gas emitted outside with the noxious gas information collected a gas sensor. Second, the sudden failure and accident can be prevented beforehand. Third, it can reduce the time and cost caused by a vehicle's own fault and check service. Fourth, noxious gas emitted to outside can be reduced.

References

[1] D. Ponsa, A. López, F. Lumbreras, J. Serrat, and T. Graf, "3D vehicle sensor based on monocular vision," *Intelligent Transportation Systems, Proceedings*, pp. 1096-1101, 2005. Article (CrossRef Link).

[2] J. Li and T. Talty, "Channel characterization for ultra-wideband intra-vehicle sensor networks," in *Proc. of Military Communications Conference, MILCOM 2006, IEEE*, pp.1-5, 2006. Article (CrossRef Link).

[3] T. Li, T, M.G. Petovello, G. Lachapelle, and C. Basnayake, "Ultra-tightly Coupled GPS/Vehicle Sensor Integration for Land Vehicle Navigation," *Navigation*, Vol.57, No.4, pp.263-274, 2010. Article (CrossRef Link).

[4] S. Kim, S.Y. Oh, J. Kang, Y. Ryu, K. Kim, S.C. Park, and K. Park, "Front and rear vehicle detection and tracking in the day and night times using vision and sonar sensor fusion," in *Proc. of Intelligent Robots and Systems, 2005(IROS 2005), 2005 IEEE/ RSJ International Conference*, pp. 2173-2178, 2005. Article (CrossRef Link).

[5] G. Cueva-Fernandez, J. P. Espada, V. García-Díaz, and R. Gonzalez-Crespo, "Fuzzy decision method to improve the information exchange in a vehicle sensor tracking system," *Applied Soft Computing*, Vol.35, pp.708-716, 2015. Article (CrossRef Link).

[6] R. Hostettler, and P.M. Djurić, "Vehicle tracking based on fusion of magnetometer and accelerometer sensor measurements with particle filtering," *IEEE Transactions on Vehicular Technology*, Vol.64, No.11, pp.4917-4928, 2015. Article (CrossRef Link).

[7] Y. Gu, L.T. Hsu and S. Kamijo, "Passive sensor integration for vehicle self-localization in urban traffic environment," *Sensors*, Vol.15, No.12, pp.30199-30220, 2015. Article (CrossRef Link).

[8] A. Genovesi, J. Harmand, and J.P. Steyer, "A fuzzy logic based diagnosis system for the on-line supervision of an anaerobic digestor pilot-plant," *Biochemical engineering journal*, Vol.3, No.3, pp.171-183, 1999. Article (CrossRef Link).

[9] B.A. Dattatraya, P.S. Ganapati, R.A. Kadu, and G.S. Popatrao, "Android Based Universal Vehicle Diagnostics and Tracking System," *International Journal of Modern Engineering and Management Research*, Vol.2, No.1, pp.35-41, 2014. Article (CrossRef Link).

[10] M. Nyberg and T. Stutte, "Model based diagnosis of the air path of an automotive diesel engine," *Control Engineering Practice*, Vol.12, No.5, pp.513-525, 2004. Article (CrossRef Link).

[11] C.D. Gadda, P. Yih, and J.C. Gerdes, "Incorporating a model of vehicle dynamics in a diagnostic system for steer-by-wire vehicles," in *Proc. of AVEC*, Vol. 4, pp.779-784, 2004. Article (CrossRef Link).

[12] J. Su, and W.H. Chen, "Fault diagnosis for vehicle lateral dynamics with robust threshold," in *Proc. of 2016 IEEE International Conference on Industrial Technology (ICIT)*, pp.1777-1782, 14-17 March 2016. Article (CrossRef Link).

[13] Y. Fan, S. Nowaczyk, and T. Rögnvaldsson, "Evaluation of self-organized approach for predicting compressor faults in a city bus fleet," *Procedia Computer Science*, Vol.53, pp.447-456, 2015. Article (CrossRef Link).

YiNa Jeong received her B.Eng. degree from Kwandong University in 2011, the M.Eng. degree in Computer Science from Kwan-Dong University in 2012. She is currently working towards a doctorate in Computer Science from Catholic Kwandong University. Her current research interests are Sensor Network, IT security, and Network security.

EunHee Jeong received her B.S. degree from Kangneung National University in 1991, the M.S. degree in Computer Science from Kwandong University in 1998 and the Ph.D. degree in Computer Science from Kwandong University in 2003 in Korea. She has been a professor of department of Regional Economics at Kangwon National University in Korea since 2003, Sept. She is a regular member of the KSII. Her current research interests are Sensor Network, IT security, web programming, and e-commerce security.

ByungKwan Lee received his B.S. degree from Pusan National University in 1979, the M.S. degree in Computer Science from Chung-Ang University in 1986 and the ph.D. degree in Computer Science from Chung-Ang University in 1990 in Korea. He has been a professor of Computer Science at Catholic Kwandong University in Korea since 1988. He was a visiting professor at Saginaw Valley State university, Michigan, USA during 2000~ 2001. He is a permanent member of the KISS and KIPS. His current research interests are network security, Wireless Sensor Network Security, Internet of Things and Big Data.