4.2 CAN Coordinator Architecture

As shown in Fig. 2, the coordinator in composed of microcontroller and CAN transceiver. Actually, since the coordinator is a logical unit, H/W components as the same as general CAN devices. For CAN ID, we employed CAN 2.0A 11bit ID, which is widely used in most vehicles, as well as compatible with both of CAN 2.0A 11 bits and CAN 2.0B 29 bits. In addition, the data rates are based on 100 kbps which is dominantly used in body electric system in a vehicle.

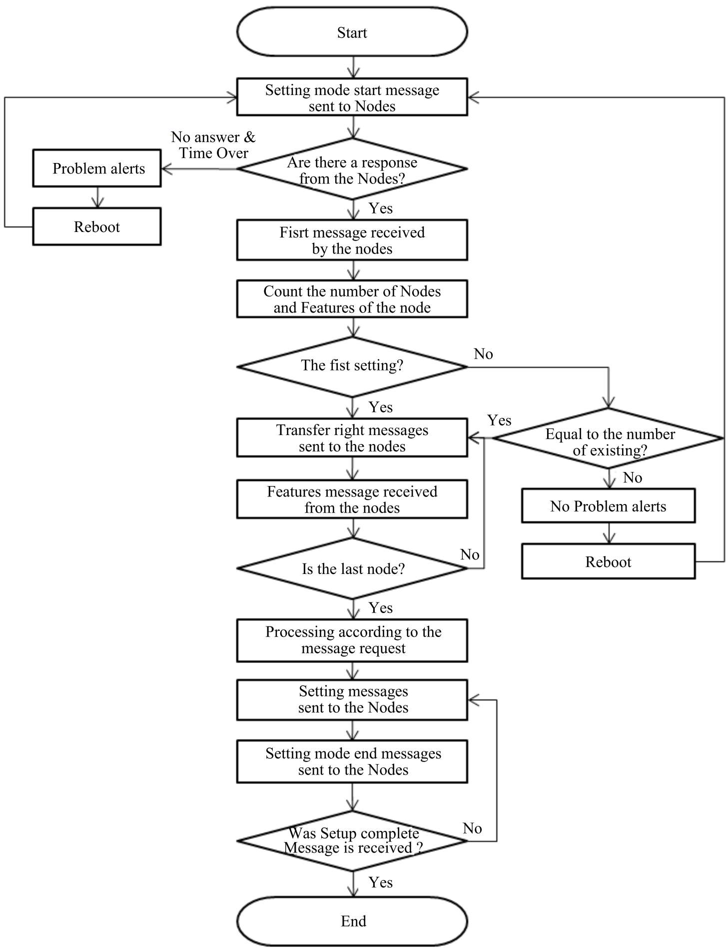


Fig. 3. Controller Area Network coordinator operation flow.

4.3 Device Type Definition

4.3.1 Coordinator

Fig. 3 illustrates a CAN coordinator operation flow. After power-on, the coordinator goes to the registration mode, and sends initiation message on the CAN Bus. During the time, it waits for responses from devices connected on the Bus for a given duration, and monitors data messages having ID between 701 and 7 FF. If there is no response during the time, CAN coordinator terminates the registration mode, and resends no response messages on the Bus for each device to check the system. If each node replies with the initiation message of coordinator, the CAN coordinator counts the number of nodes and the number of messages of each node. If there is no error in the messages, the coordinator sends a second message, which is to notify nodes transmission sequence assigned, and then waits for reply. The CAN coordinator, which receives the responses from devices stores several information about device role, operation, location, urgent message, and ID deployment request. Based on the ID deployment request, CAN coordinator assigns ID according to either automatic deployment or manual deployment. In the case that conventional ID deployment is used, it also check redeployment request for performance enhancement or revisions, and if so, the coordinator reassigns ID according the requests. On the completion of ID deployment, the coordinator checks urgent messages. It is important to note that the urgent message can have dual IDs with respect to a single task.

After finishing ID deployment for role and operation of messages, the coordinator sends a set-up message in which operation information, and data ID in normal mode are defined. Once receipt of set-up message, each device sends back registration termination message on the Bus. It is important to notice that the CAN coordinator does not mediate normal mode CAN operation after registration mode until next power on or reset. Each node is now ready to communicate with each other according to data ID, role, and operation method which are configured by the coordinator.

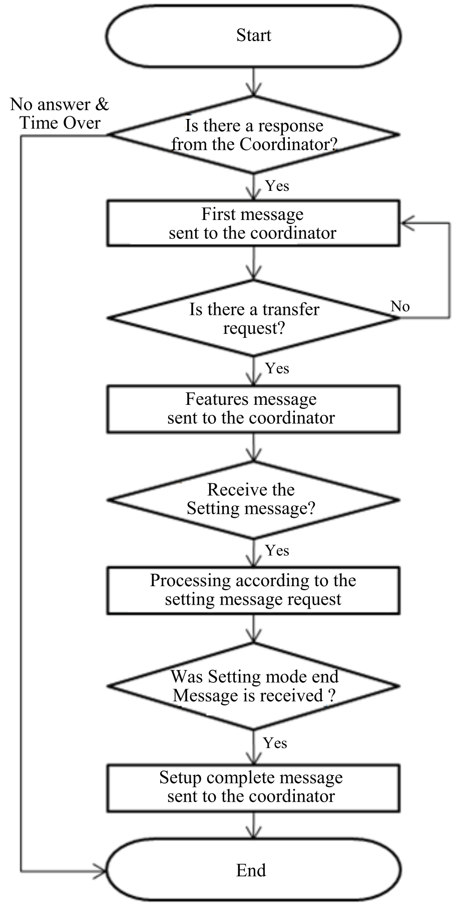


Fig. 4. Device operation flow.

4.3.2 CAN device

Fig. 4 illustrates the operation flow of a CAN device. After power-on, each node goes to registration mode, and waits for registration initiation message from coordinator. If a device cannot receive the initiation message from a coordinator, in order to avoid system faults, it changes its mode to normal mode. Otherwise, after initiation response transmission, each node waits for message sequence having several functional descriptors. The nodes that receive the sequence message from coordinator, transmits second message according to the given sequence. After transmitting configuration and registration completion message, each node changes its mode to normal mode, and operates as in conventional CAN system.

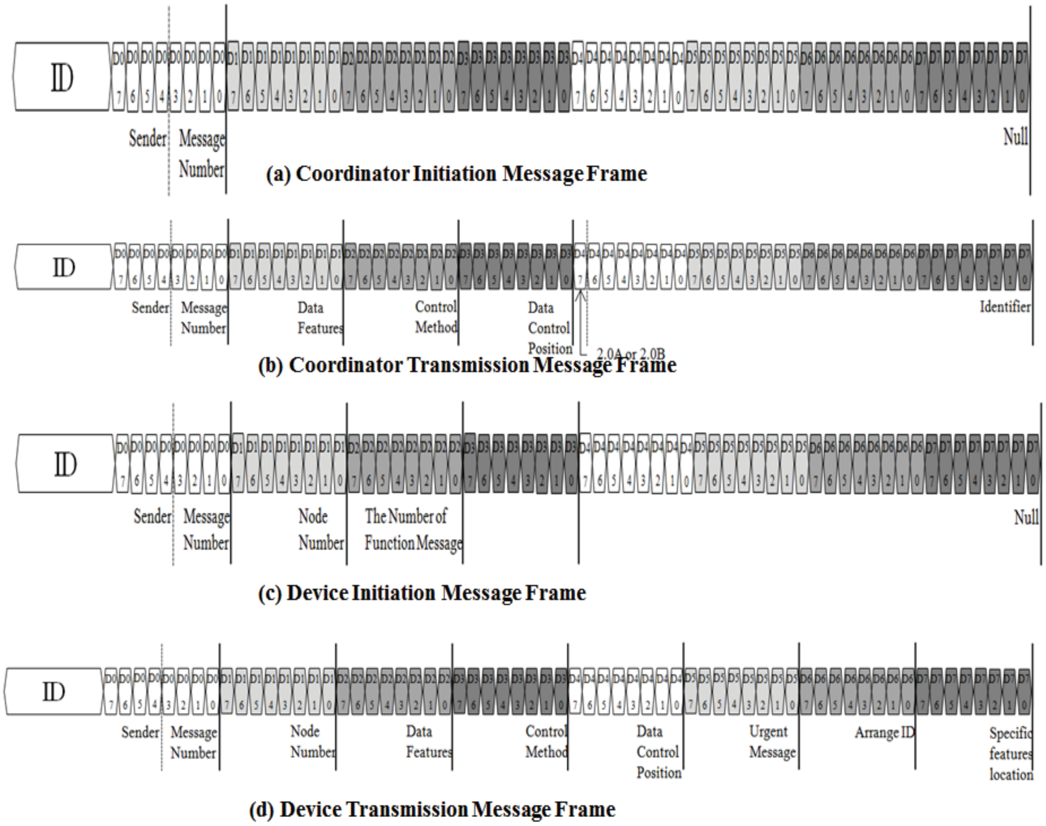


Fig. 5. Examples of frame structures according to different functions.

4.4 Frame Structure

In order to accommodate the functionalities newly defined in this paper, we redefine the frame structure based on the new ID assignment criteria as shown in Fig. 5. As already mentioned in the previous subsection, since we utilize CAN 2.0A 11 bits, the range of ID is 000 to 7 FF as shown in Table 1. Therefore, the section is used for urgent message deployment, normal message deployment, registration messages during registration process.

Table 1. ID range according to the message functions

ID	Message description
000–0 FF	Urgent message section
100–6 FF	Normal message section
700–7 FF	Registration message section

* Urgent message: The messages have the highest priority over other messages. Furthermore, it also has least transmission delay due to high priority. For example, AIR BAG system.
* Normal message: The messages are used in normal CAN operation ranging from 100 to 6 FF.
* Registration message: In the proposed framework, after power-on, all the nodes enter registration mode, the messages for the registration mode are assigned ranging from 701 to 7 FF.

5. Experimental Results

5.1 Experimental Environments

In order to test the feasibility and performance of the proposed CAN framework, we developed new CAN systems including CAN coordinator and devices. As shown in Fig. 6, hardware architecture of CAN device and coordinator is the same but they operate differently according to the software. Hardware prototype is composed of Microcontroller [12] integrated with CAN transceiver, and simple IO (Buzzer and push switch).

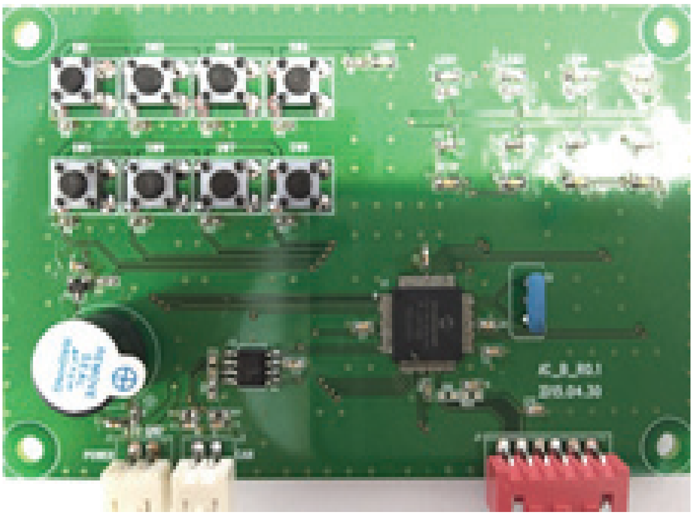
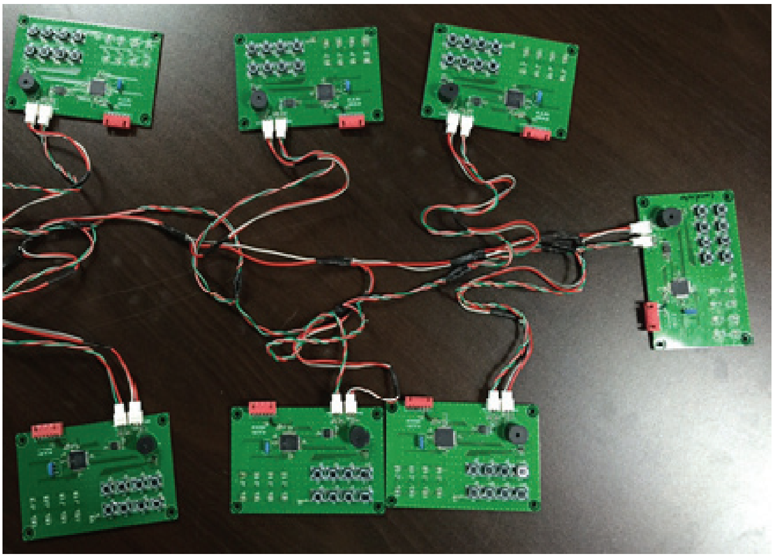
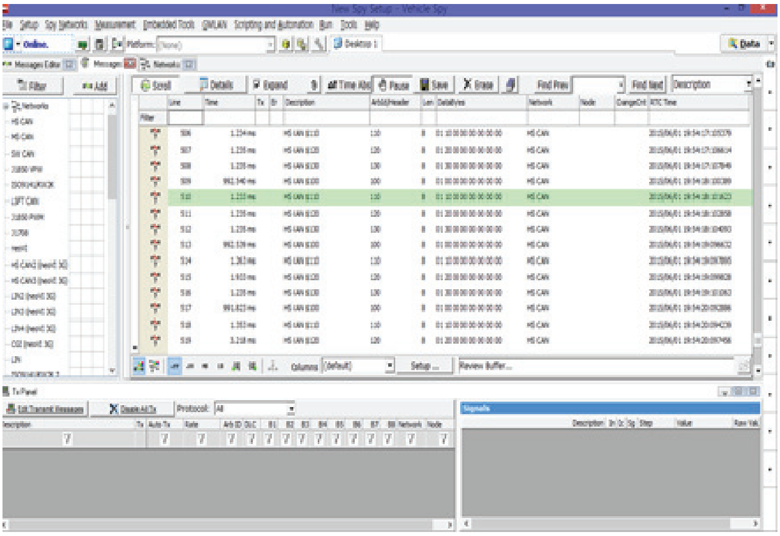


Fig. 6. Controller Area Network test node.

To construct a test-bed, we used 1 coordinator and 6 devices. In addition, each device regardless of device type, each device can generates messages based on on-off signal, and to verify and monitor CAN messages on the Bus in real-time, we used Vehicle Spy [13] with professional instrument, neoVI (Intrepid Control System Inc., Madison Heights, MI, USA) as shown in Fig. 7.



(a)



(b)

Fig. 7. Test-bed (a) and monitoring software (b).

5.2 Performance Evaluation

First we observed transmission delay of the proposed system. As mentioned previously, we used CAN 2.0A ID system, and allocate ID ranging 100–6 FF by separating ID region according to data functions. Each device transmits a message at the data rates of 100 kbps every 1000 ms at the same time. Therefore, several messages are collided with each other, and due to collision arbitration mechanism, some messages that have lower priorities are delayed by 1.2–1.7 ms. As shown in Fig. 8, when 6 collisions occur in the network, the delay is about 6 ms. However, when our algorithm is applied, the delay is negligible compared to normal CAN system. That is because to cope well with urgent messages, we allocated urgent message region ranging from 000 to 0 FF. Therefore, an urgent message can get dynamically high priority, so that urgent messages will experience less transmission delay. On the other hand, transmission delay in the normal CAN system is increasing proportional to the number of TX data.

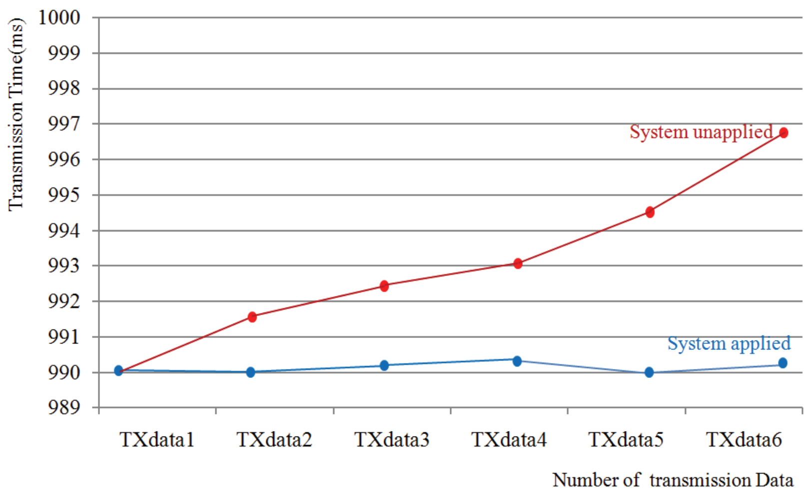


Fig. 8. Transmission delay (proposed CAN framework vs. conventional system).

In addition, we also tested the feasibility and compatibility of the proposed framework with conventional CAN systems. To verify our framework, we experimented with Plug and Play functions, in which 1 coordinator, 4 nodes. They are already installed and operate, and during runtime, a new node is attached to the network. Fig. 9 shows the summary of plug and play operation verified by the CAN monitor instrument.

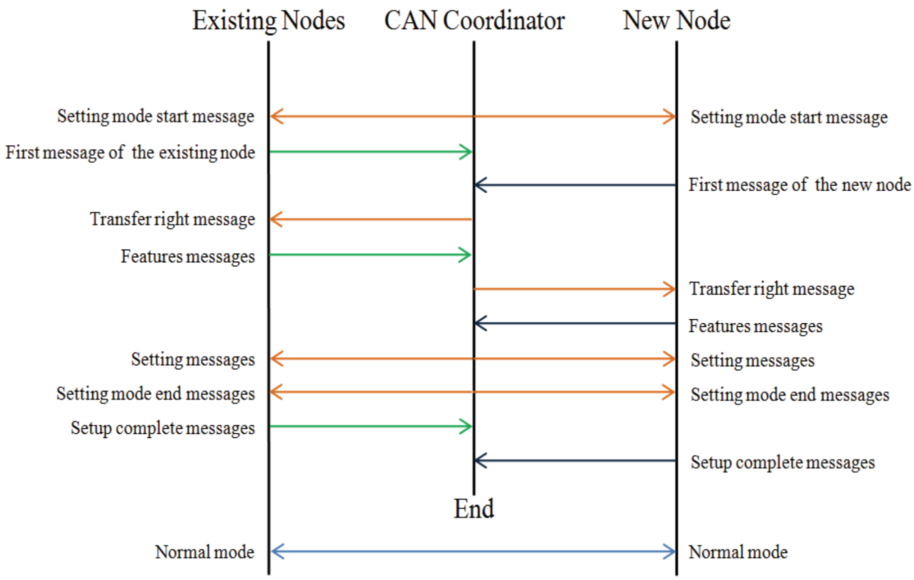


Fig. 9. Plug and Play function verification.

6. Concluding Remarks

In this paper we proposed a new system component, called CAN coordinator, and design a new CAN framework capable of supporting plug and play functionality. In the proposed framework, we tried to solve two inherent problems of CAN systems. First, our framework enhanced unpredictable transmission delay, in particular, with respect to real-time messages that should be transmitted within deadline, and which enables to get higher priorities by newly defining urgent messages using the new ID region. The experiments showed that in the case that a number of nodes transmit messages at the same time, while urgent messages can send earlier, transmission delay of normal messages are relatively increased. However, in real system environment, since the time gap between CAN messages exists, the possibility that normal messages are seriously delayed is very low.

Next goal of our work is to support Plug and Play function. To achieve this, we defined a new device, a CAN coordinator, which can manage the information, such as message function, operation and location, and reallocate ID. In addition, we also redefine a new mode, registration mode for a coordinator to dynamically recognize and manage a new node. Our experimental results also proved that a new node can enter the existing network regardless of operation of the other nodes on the Bus.

Therefore, we expect that the proposed framework will be used as a concrete guideline to enhance CAN system and network, and we are also trying to design flexible CAN system to be used in more various application fields beyond vehicle areas.

Acknowledgement

This work was supported by the Incheon National University Research Grant in 2016.

References

[1] D. Marsh, "Network protocols compete for highway supremacy," 2003 [Online]. Available: http://www.edn.com/design/communications-networking/4331809/Network-protocols-compete-for-highway-supremacy.

[2] S. Lee, M. H. Kim, and K. C. Lee, "Survey on in-vehicle network system researches," *Journal of the Korean Society of Precision Engineering*, vol. 23, no. 9, pp. 7-14, 2006.

[3] Bosch, *CAN Specification Version 2.0*. Stuttgart: Bosch GmbH, 1991.

[4] K. N. Ha, M. H. Kim, K. C. Lee, and S. Lee, "performance evaluation of network protocol for automated transfer crane system," *Journal of Control, Automation, and Systems Engineering*, vol. 11, no. 8, pp. 709-716, 2005.

[5] W. E. Seitz, "Controller area network in embedded machine control," 2004 [Online]. Available: http://www.techonline.com/electrical-engineers/education-training/tech-papers/4125494/Controller-Area-Network-in-mbedded-Machine-Control.

[6] J. P. Lehozky and L. Sha, "Performance of real-time bus scheduling algorithm," *ACM SIGMETRICS Performance Evaluation Review*, vol. 14, no. 1, pp. 44-53, 1986.

[7] G. C. Buttazzo, "Rate monotonic vs. EDF: judgment day," *Real-Time Systems*, vol. 29, no. 1, pp. 5-26, 2005.

[8] M. H. Kim, K. N. Ha, K. C. Lee, and S. Lee, "Traffic prediction of CAN network system with dual communication channels," in *Proceedings of International Conference on Control, Automation and Systems*, Seoul, Korea, 2008, pp. 397-400.

[9] M. H. Kim, J. G. Lee, S. Lee, and K. C. Lee, "A study on distributed message allocation method of CAN system with dual communication channels," *Journal of Institute of Control, Robotics and Systems*, vol. 16, no. 10, pp. 1018-1023, 2010.

[10] S. H. Kim, "Design and implementation of CAN communication system capable of supporting real-time plug and play," M.S. thesis, Incheon National University, Incheon, 2015.

[11] S. H. Kim and K. Hwang, "Design and implementation of CAN communication system capable of supporting real-time plug and play," in *Proceedings of the Korea Information Processing Society (KIPS) Fall Conference*, Jeju, Korea, 2015.

[12] Microchip PIC18F66K80 datasheet [Online]. Available: http://ww1.microchip.com/downloads/en/DeviceDoc/39977f.pdf.

[13] Vehicle Spy [Online]. Available: http://www.intrepidcs.com/products/software/vehicle-spy-professional/.

Sungheo Kim

He received B.S. degree in the National Institute for Lifelong Education (NILE), Korea, 2012, and M.S. degree in Incheon National University, Korea, 2015. He is currently working in Garin System Corp. Korea.

Kwang-il Hwang http://orcid.org/0000-0002-4196-4725

He is an associate professor, in the Department of Embedded Systems Engineering, Incheon National University, Korea. He has received M.S. and Ph.D. in Electronics and Computer Engineering from Korea University, Seoul, Korea, 2004 and 2007, respectively. He is a number of international conferences hosted by KIPS CSWRG. He is also a member of IEEE, KICS, KMMS, KIPS, and KISS.