

# 다양한 차종을 고려한 데이터베이스 기반 차량 데이터 수집 시스템의 설계 및 구현

김 범 준\*, 김 찬 우\*, 백 윤 주<sup>o</sup>

## Design and Implementation of a Database-Based Vehicle Data Acquisition System Considering Various Vehicle Models

Beomjun Kim\*, Chanwoo Kim\*, Yunju Baek<sup>o</sup>

### 요 약

차량이 첨단화됨에 따라 차량의 내부에는 많은 수의 전자 제어 장치가 내장되고 있다. 또한 차량의 진단을 목적으로 차량에 존재하는 OBD-II 포트를 통해 차량의 정보를 획득하여 활용하는 기술에 대한 관심이 높아지고 있다. 하지만, 차량의 주요 정보 중에서 표준을 통해 획득 방법이 정의되지 않은 일부 차량 정보들은 차량 제조사나 차종에 따라 데이터 수집 방법이 다른 문제가 있다. 본 논문에서는 다양한 차종을 고려하여 데이터베이스에서 차종에 따른 데이터 수집 방법을 찾아 적용하는 차량 데이터 수집 시스템을 제안하고 구현하였다. 이 시스템은 차종에 따라 여섯 가지 차량 정보에 대한 정보 수집을 수행한다. 시스템의 구현을 위하여 OBD-II 장치와 스마트폰 애플리케이션을 개발하였다. 이 과정에서 OBD-II 장치는 차량 진단을 위한 3가지 프로토콜을 활용할 수 있도록 구현하였다. 소형 하드웨어의 플래시 메모리에 다양한 차종에 대한 데이터베이스를 구축하기 위하여 데이터를 인코딩하는 기법을 적용하였다. 또한 시스템의 동작을 확인하기 위하여 32대의 실제 차량을 이용하여 데이터 수집 과정을 검증하였다. 실험 결과 23대의 차량은 데이터베이스에 저장된 방법을 통해 여섯 개의 정보를 수집할 수 있었고, 8대의 차량은 데이터베이스에 저장된 다섯 개의 정보를 수집할 수 있음을 확인하였다.

**Key Words** : On-Board Diagnostics, database, embedded system, Vehicular IT, CAN

### ABSTRACT

As vehicle manufacturing technology develops, a large number of electronic control units (ECUs) are embedded inside the vehicle. In addition, there is increasing interest in technology for querying the status of the vehicle through the OBD-II port, which is mandatory for the vehicle for diagnostic purposes. However, some vehicle information whose acquisition method is not defined through the OBD-II standard has a problem in that the data collection method is different depending on the vehicle manufacturer or vehicle type. In this paper, a vehicle data collection system is proposed and implemented to find and apply data collection methods according to vehicle models in a database considering various vehicle models. The system collects information on six types of vehicle information depending on the vehicle type. For the implementation of the system, OBD-II devices and smartphone applications were developed. In order to build a database for various models in a flash memory of small hardware, a technique for encoding data was applied. In order to verify the operation of the system, the data collection process was verified using 32 actual vehicles.

※ 이 논문은 부산대학교 기본연구지원사업(2년)에 의하여 연구되었음

• First Author : Pusan National University Dept. of Electrical and Computer Engineering, beomjun.kim@eslab.re.kr, 학생회원

◦ Corresponding Author : Pusan National University Dept. of Electrical and Computer Engineering, yunju@pusan.ac.kr, 종신회원

\* Pusan National University Dept. of Electrical and Computer Engineering, chanwoo.kim@eslab.re.kr, 학생회원

논문번호 : 202007-179-D-RE, Received July 31, 2020; Revised September 27, 2020; Accepted October 8, 2020

## I. 서 론

차량 제조 및 제어 기술이 첨단화됨에 따라 차량의 전자화를 위하여 내부에 다양한 전자 제어 장치를 내장하게 되었다. 더불어 차량 내부의 전자 제어 장치 간 통신에 대한 관심도 함께 높아지고 있다. 관련 기술이 지속적으로 발전하면서 차량 내부의 네트워크에 대한 관심은 기술자를 넘어 일반 운전자에게까지 확대되었다.

특히 차량의 상태를 진단하기 위한 목적으로 표준화된 포트에 상용 제품을 장착하면 누구나 쉽게 자신의 차량을 모니터링할 수 있게 되었다. 하지만 많은 상용 제품들은 일반적인 진단 목적에 활용되는 정보만을 추출할 수 있는 제한점을 가지고 있으며, 스마트 차량 기술의 연구나 주행 기록에서 관심을 가지는 주요 정보를 추출하기 위한 기능은 정비를 목적으로 하는 고가의 장치에서만 지원하고 있다. 일부 상용 제품은 ADAS 기능을 지원하기 위하여 차량의 네트워크로부터 정보를 수집하고 있지만 그 정보를 공개하지 않는다.

차량 내부의 정보를 활용하여 운전자를 위한 기술을 개발하는 다양한 연구들이 존재한다. 예를 들어, 운전자의 안전을 위하여 피로 및 졸음운전을 인식하는 연구<sup>1-3)</sup>나 운전자의 운전 습관을 분석하는 연구<sup>4,5)</sup>가 있다. 또한 운전자의 의도를 분석하기 위하여 차선 변경이나 회전을 인식<sup>6,9)</sup>하기도 하고, 도난 방지를 목적으로 운전자를 구분하는 연구<sup>10, 11)</sup>도 존재한다. 이와 같은 다양한 차량 연구에서 활용하는 정보들의 일부는 표준에 정의되어 있지 않아, 데이터 수집에 어려움을 겪는 경우가 많다.

대표적인 상용 제품들은 ELM Electronics 사의 ELM327<sup>12)</sup>을 기반으로 OBD-II PID를 활용한 표준 정보를 스마트폰으로 모니터링을 하는 것을 목적으로 한다. 따라서 OBD-II PID에 정의되지 않은 정보를 획득하기 어려운 한계점을 가지고 있다. 실제 차량 기술 연구에 활용되는 다양한 차량 조작 정보를 수집하기 위해서는 고가의 정보 수집 솔루션을 구매해야 하는 문제점이 있다.

이러한 문제를 해결하기 위하여 표준으로 정의되지 않은 차량 정보들을 획득하는 방법에 대한 기존 연구들<sup>13-19)</sup>이 존재한다. CAN 정보 분석을 위해 정보를 여러 클래스로 구분하고 데이터간의 경계를 분석하는 연구들<sup>13-15)</sup>은 CAN 정보를 변하지 않는 고정 값과 지속적으로 증가하는 counter 값, CRC, 그리고 센서 정보로 나누었다. 데이터의 종류와 경계는 시간 순서대

로 수집한 데이터의 변화를 관찰하면서 수행된다. 또한, 표준으로 획득 가능한 OBD-II 정보를 참고하거나 IMU에서 얻은 정보를 활용하여 회귀분석이나 필터링을 기반으로 CAN 정보를 분석한 연구<sup>16)</sup>도 있다. 유사한 CAN 정보를 클러스터링하여 분석한 연구<sup>17)</sup>는 이미 알고 있는 CAN 정보와 유사한 다른 정보들을 분석하는 방법을 제시하였다. 또한 기존 연구들이 고려하지 못한 CAN 정보의 바이트 순서, 부호의 유무를 적극적으로 활용하여 정보를 해석하는 연구<sup>18)</sup>와 비트의 변화를 지속적으로 감시하는 기법으로 5대의 실제 차량을 활용하여 CAN 정보의 분석을 시도한 연구<sup>19)</sup>가 있다.

하지만, 차량 제조사와 차종에 따라서 정보를 획득하는 방법이 달라지기 때문에 다양한 차종의 정보 획득이 어렵다는 문제가 있다.

본 논문에서는 다양한 차종에 대한 차량 데이터 수집 과정을 단순화할 수 있는 시스템에 관하여 연구하였다. 이를 위해 차대번호를 획득하여 차종을 분석할 수 있고 차종에 따라 센서 정보의 획득 방법을 데이터베이스화하는 기법을 고안하였다. 이 과정에서 센서 정보를 획득하기 위한 진단 프로토콜 3종을 지원할 수 있도록 설계하였다. 또한 낮은 비용으로 제작할 수 있는 소형 하드웨어를 기반으로 데이터베이스를 저장하고 활용하기 위한 소프트웨어를 구현하였다. 시스템은 크게 소형 OBD-II 장치와 안드로이드 애플리케이션으로 나누어 구현하였으며, 장치를 실제 차량들에 장착한 후 데이터 수집 실험을 수행하여 제안하는 시스템의 동작을 검증하였다.

## II. 관련 연구

### 2.1 CAN 프로토콜

CAN(Controller Area Network)이란 차량 내에 존재하는 전자 제어 장치(Electronic Control Unit)의 통신을 목적으로 설계된 국제 표준 규격으로 물리 계층과 데이터 링크 계층이 ISO 11898로 제정되어 있다. CAN은 버스 구조로 되어 있어 여러 ECU가 동시에 연결될 수 있다. 또한, CAN은 메시지 기반 프로토콜로 물리적인 주소를 통해 수신 대상을 설정하지 않는다. 각 ECU는 우선순위에 따라 11개 또는 29개의 비트로 구성된 식별자(Identifier, ID)를 가지고 있다. 낮은 ID 값을 가지는 ECU가 CAN 버스에서 더 높은 우선순위를 가지게 되며, 우선순위가 낮은 ECU는 버스가 비활성 상태가 되면 재전송을 시도하게 된다. CAN은 버스 구조를 통해 배선 길이가 짧아진다는 장점

로 인해 현재에도 차량 ECU 간 통신 프로토콜로 널리 사용되고 있다.

### 2.2 차량 진단 기술

온보드 진단기(On-Board Diagnostics)는 차량의 상태를 진단하여 고장을 빠르게 파악할 수 있게 도와주는 장치이다. 온보드 진단기가 사용하는 대표적인 인터페이스인 OBD-II는 커넥터가 SAE J1962로 표준화되어 있으며, 한국에서도 의무화됨에 따라 모든 차량이 OBD-II 포트를 가지고 있다. 표준화된 OBD-II 커넥터에서 7번 핀은 K-line으로 할당되어 있으며, 6번과 14번 핀은 CAN 통신을 위해 할당되어 있다. OBD-II 표준을 따르는 차량들은 진단을 통해 발견한 고장을 고장 진단 코드(Diagnostic Trouble Codes, DTC)로 나타낸다.

또한 SAE J1979 표준에서는 OBD-II Parameter ID(OBD-II PID)를 기반으로 차량의 정보를 요청하는 방법을 정의하고 있다. 이 표준에서 정의한 질의에는 차량의 엔진 RPM, 속도, 차대번호 등 다양한 정보를 요청할 수 있으나 차량의 조작과 관련된 일부 정보들의 요청은 정의되어 있지 않아 요청이 불가능한 단점이 있다.

진단을 위해 K-line을 활용하는 Keyword Protocol 2000(KWP2000)은 ISO 14230으로 표준화되어 있으며, 물리 계층은 ISO 9141을 확장하였다. 이를 지원하는 차종에서는 K-line을 기반으로 차량의 정보를 질의 메시지로 요청하여 응답 데이터를 획득할 수 있다.

Unified Diagnostic Services(UDS)는 ISO 14230과 ISO 15765와 같은 표준들을 기반으로 ECU와 통신할 수 있도록 만들어진 진단용 통신 프로토콜이다. UDS에서 정의된 서비스 중 ID를 통해 데이터를 읽어오는 질의를 사용하면 특정 ECU의 데이터를 확인할 수 있다.

또한, 많은 차량들에서 별도로 요청하지 않은 메시지가 OBD-II의 CAN 버스에서 지속적으로 제공되는 경우도 있다. 각 메시지의 데이터가 의미하는 뜻이 공개되어 있지 않지만 정보의 활용성이 높기 때문에, 이를 분석할 필요성이 있다. 대표적인 유용한 정보는 기어의 위치, 스티어링 휠 각도, 브레이크 페달 압력, 오도미터, 바퀴의 속도 등이 있다.

## III. 차량 데이터 수집 시스템

### 3.1 시스템의 구조

본 논문에서 제안하는 시스템은 크게 차량으로부터 데이터를 수집하여 전달하는 OBD-II 장치와 데이터를 수신하여 저장하는 스마트폰으로 나누어진다. 제안하는 시스템의 구성요소 및 상호작용은 그림 1과 같다.

OBD-II 장치는 차량에 존재하는 OBD-II 포트에 연결되어 차량 내부의 정보를 수집하는 장치이다. 이 장치에는 CAN 통신을 지원하는 SoM(Systems on Module)과 CAN 트랜시버가 내장되어 있어 CAN에 메시지를 전송하거나 수신할 수 있다. 또한 ISO 9141 인터페이스를 활용하여 K-line을 통한 데이터 송수신이 가능하다. 또한, 차량별로 K-line에 사용되는 핀이 달라질 수 있다는 점을 고려하여 하드웨어가 설계되었다. 다양한 차종의 동작을 고려하여 OBD-II 장치의 메모리에는 차량별로 상이한 센서 데이터 요청 DB가 저장되어 있다. DB는 차량의 정보 획득을 위해 필요한 다양한 테이블로 구성되어 있다. 또한, SoM에 포함된 통신 기능 중 하나인 Bluetooth Low Energy(BLE)를 활용하여 데이터를 무선으로 전송할 수 있다.

스마트폰은 OBD-II 장치의 동작을 설정하여 데이터를 요청할 수 있으며, 요청한 데이터를 BLE를 통해 수신할 수 있다. 또한 필요에 따라 스마트폰에 내장되어 있는 IMU나 GPS, 카메라 등의 추가적인 센서 정보를 함께 저장할 수 있다. 스마트폰이 OBD-II 장치에 요청할 수 있는 센서 데이터의 종류는 스티어링 휠 각도(SWA), 방향 지시등(TSS), 브레이크 페달 압력(BPP), 오도미터(ODO), 운전석 안전벨트 스위치(BELT), 기어 포지션(GEAR)의 여섯 종류이다.

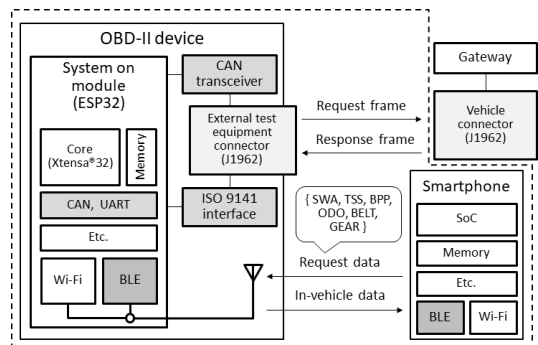


그림 1. 데이터 수집 시스템의 구조  
Fig. 1. Structure of data acquisition system

### 3.2 시스템의 동작

스마트폰의 요청으로 OBD-II 장치가 센서 데이터를 수집하기 위해서는 데이터베이스에 접근하여 관련 정보를 확인해야 한다. 여섯 종류의 데이터를 획득하는 방법은 제조사 및 차량 모델에 따라 달라질 수 있기 때문에 해당 정보를 확인하는 과정이 필요하다. 제안하는 시스템에서 센서 관련 정보를 획득하는 과정은 그림 2와 같이 나타난다. 먼저 OBD-II 질의를 통해 차대번호(Vehicular identification number)를 획득한다. OBD-II 표준에서는 9번째 서비스 중 PID 02에 대하여 질의하면 17자의 차대번호 응답을 돌려주도록 정의되어 있어 차량의 종류를 확인할 수 있다. 일부 차량의 경우 OBD-II 표준에 따라 질의를 하여도 응답하지 않는 경우가 존재하며, 시스템 설계 과정에서 해당 예외상황에 대한 정보 획득 과정은 고려하지 않았다.

OBD-II 장치는 메모리에 저장된 모델 테이블을 활용하여 차량으로부터 얻은 차대번호를 모델명으로 변환할 수 있다. 차량 모델에 대한 테이블은 국내에서 유통되는 차량의 세부 차종과 엔진 종류를 고려하여, 979개의 차대번호에 대한 모델 정보를 저장하고 있다. 해당 테이블이 취급하는 차량 제조사는 현대, 기아, 쌍용, 쉐보레, 르노의 다섯 종류이다.

차대번호를 이용하여 데이터베이스에서 획득한 모델 정보를 이용하면 시스템에서 고려하고 있는 여섯 종류의 센서 데이터에 대한 관련 정보를 획득할 수 있다. 6개의 테이블은 세부 차종과 엔진 종류를 모두 고려한 경우와는 다르게 각각 347종의 모델에 대한 데이터를 저장하고 있다. 센서 관련 정보는 차량에 질의할 때 사용되는 진단용 프로토콜 종류와 질의에 필요한 ID, 응답에 대한 해석 방법 등을 포함한다.

이후, OBD-II 장치는 획득한 센서 관련 정보를 기반으로 차량에 여섯 종류의 센서 정보를 질의하고 응답을 분석하여 스마트폰으로 전송하게 된다.

의 3가지 차량 프로토콜을 기반으로 차량의 정보를 획득하는 기법을 채택하였다. 본 시스템에서 차량의 센서 정보 데이터베이스를 구성하고 정보를 획득하는 과정에서 3개의 프로토콜은 고정된 우선순위를 가진다.

제안하는 차량 정보 획득 기법은 가장 먼저 UDS를 기반으로 데이터 획득을 시도한다. 일부 최신 차량들은 비요청 CAN이 존재하지 않으며 UDS로 대부분의 정보를 획득 가능한 경우가 존재한다. 또한 비요청 CAN이 존재함과 동시에 UDS를 지원하는 차종이 많기 때문에 UDS가 비요청 CAN에 비해 높은 우선순위를 가진다. 만약 UDS를 통해 차량 정보를 획득하지 못할 경우, 비요청 CAN을 통해 정보 획득을 시도하며, 마지막으로 K-line을 통한 정보 획득 과정이 수행된다. K-line 기반의 정보 획득 과정은 관심 있는 정보들을 제공하는 경우가 적어 가장 낮은 우선순위를 가진다.

비요청 CAN 기반의 데이터베이스는 UDS로 획득할 수 없는 차량 정보들을 대상으로 기존 연구[20]를 적용하여 추출하여 구성하였다. UDS 및 K-line 기반의 데이터베이스는 별도의 추출 기법을 활용하지 않고, 상용 진단 장치에 활용되는 데이터베이스를 기반으로 구성하였다.

### 3.4 데이터베이스 구조

제안하는 시스템에는 모델에 대한 테이블과 여섯 종류의 센서 정보 테이블, 차량의 추가적인 정보를 포함하는 INFO 테이블로 구성된다. 데이터베이스의 상세한 구조는 그림 3과 같이 나타난다. 차량에 대한 추가적인 정보는 차대번호를 조회한 직후 스마트폰에 전송하여, 차량에 대한 정보를 제공하게 된다. 이 정보들이 저장된 INFO 테이블은 연식, 모델, 세부 모델, 배기량, 연료 탱크, 연료 종류 등의 정보가 있다. 또한, 차량에 따라 OBD-II 포트에서 K-line으로 사용하는 핀이 표준과 다른 경우가 있기 때문에 해당하는 핀 정보를 추가적으로 포함한다.

또한 센서 정보 중에서 기어 포지션은 GEAR, 방향 지시등은 TSS, 안전벨트 착용 여부는 BELT, 브레이크 페달 압력은 BPP, 오도미터는 ODO, 스티어링 휠 각도는 SWA 테이블에 저장된다. 여섯 테이블은 공통적으로 차대번호로 조회 가능한 모델명 정보 model을 가지고 있다. 여섯 종류의 센서 관련 테이블은 센서 데이터를 분석하기 위해 최대 2개의 플래그 값을 가진다. 첫 번째 플래그(flag1)에는 데이터의 바이트 순서(Byte order)가 빅 엔디언(Big-endian)인지 리틀 엔디언(Little-endian)인지를 나타내는 비트, 데이터의

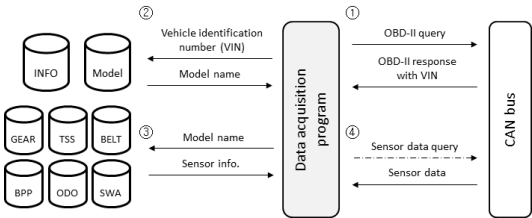


그림 2. 차대번호를 통한 센서 관련 정보 획득 과정  
Fig. 2. Obtaining sensor-related information through VIN

### 3.3 차량 정보 획득 기법

제안하는 시스템에서는 UDS, 비요청 CAN, K-line

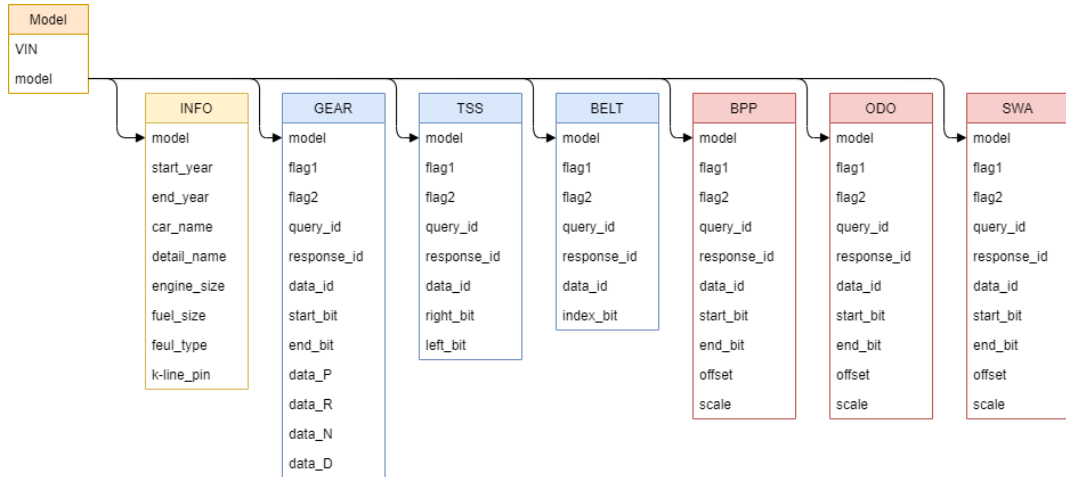


그림 3. 펌웨어에 내장된 데이터베이스의 구조와 각 테이블의 구성요소  
 Fig. 3. Structure of the database built into the firmware and the componets of each table

부호가 존재하는 경우(signed)와 부호가 존재하지 않는 경우(unsigned)를 구분하는 비트, 해당 정보를 획득하기 위한 정보가 UDS와 K-line 또는 요청하지 않은 CAN 메시지 내에 존재하는지에 대한 여부를 표현하는 비트로 구분된다. 두 번째 플래그(flag2)는 데이터의 단위 종류를 나타낸다.

query\_id와 data\_id는 질의 과정에서 필요한 ID 정보를 저장하고 있으며, response\_id는 응답 과정에서 필요한 ID 정보를 저장하고 있다. UDS로 데이터를 수집할 경우에는 질의와 응답에 필요한 ID가 다르기 때문에 둘 다 저장하고 있으며, K-line의 경우에는 질의와 응답에 동일한 ID를 사용하기 때문에 요청 ID만 존재한다. 요청하지 않은 CAN 메시지를 분석하는 경우에는 질의 과정이 없으므로 응답 ID만 존재한다.

시작 비트(start\_bit)와 종료 비트(end\_bit)는 응답 메시지를 분석하기 위해 사용하는 비트 위치를 나타낸 것이다. 시작 비트부터 종료 비트까지의 비트들을 모두 비교에 활용한다.

추가적으로, 기어 포지션의 분석은 기어가 P, R, N, D의 위치에 있을 때, 어떤 값을 가지는지에 대한 데이터(data\_P, data\_R, data\_N, data\_D)를 저장하고 있다. 방향 지시등은 왼쪽과 오른쪽에 대한 비트 위치(left\_bit right\_bit)들을 저장하고 있으며, 운전석 안전벨트 스위치는 해당하는 데이터 비트 위치(index\_bit)를 저장한다.

오프셋(offset)과 스케일(scale)은 수식 (1)과 같은 형태로 계산에 활용된다. 오프셋과 스케일은 부동소수점의 형태로 저장되어 있다.

$$value = (data[s:e] + offset) \times scale \quad (1)$$

#### IV. 구현 및 테스트

##### 4.1 하드웨어 구현

시스템에서 제안한 OBD-II 장치는 Wi-Fi와 BLE 통신을 지원하는 SoM인 ESP32-WROOM-32를 기반으로 제작되었다. CAN 통신을 수행하기 위한 트랜시버는 MCP25625를 사용하였고, ISO 9141 인터페이스는 L9637를 사용하였다. 또한, 차량에 존재하는 OBD-II 포트를 활용하기 위하여 J1962 커넥터를 사용하고 있다. 하드웨어의 전원은 OBD-II 포트의 12V를 통해 공급되며, 레귤레이터를 통해 ESP32 모듈에 전원을 인가한다. 마이크로 USB 포트는 차량에 연결하지 않은 상태에서 전원을 인가하는 목적과 USB를 통한 시리얼 통신으로 펌웨어를 업로드하기 위한 목적으로 제작되었다. 제작한 하드웨어의 구조와 사진은 그림 4와 같이 나타난다.

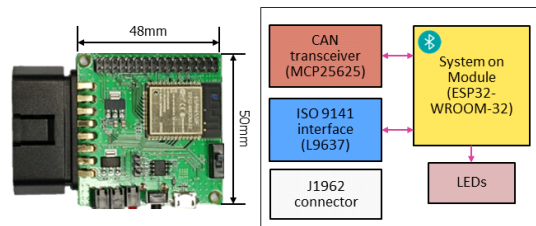


그림 4. OBD-II 장치의 사진과 하드웨어 구조  
 Fig. 4. Hardware and structure of OBD-II device

#### 4.2 소프트웨어 구현

제안하는 시스템의 소프트웨어는 OBD-II 장치에 구현한 펌웨어와 스마트폰의 안드로이드 애플리케이션을 구현하였다. 펌웨어는 스마트폰 애플리케이션은 BLE를 통해 서로 연결하여 동작할 수 있도록 구현하였다.

OBD-II 장치의 펌웨어는 다양한 데이터베이스를 포함하기 위하여 제조사에서 제공하는 Non-volatile storage(NVS) 라이브러리를 활용하였다. NVS 라이브러리는 플래시 메모리 영역에 키(key)와 값(value)을 짝지어 저장할 수 있는 기능을 제공한다. 모델 테이블은 차대번호를 키로 사용하여 간단하게 모델 정보를 제공한다. 하지만, 나머지 7개의 테이블은 여러 개의 값을 가져야하기 때문에 16진수로 인코딩된 값을 사용한다. 테이블에서 정의된 값들은 모두 하나의 고정된 길이를 가진 값으로 인코딩되어 키와 짝지어진다. 최종적으로 데이터베이스는 바이너리(binary) 파일로 변환되어 ESP32의 플래시 메모리에 적재된 후, 펌웨어 프로그램이 호출한 라이브러리 함수를 통해 키에 해당하는 데이터 값을 조회하게 된다.

OBD-II 장치가 차량에 장착되어 전원이 인가되면, 안드로이드 애플리케이션의 연결을 기다린다. 애플리케이션과 연결되면 차량의 차대번호를 조회하여 모델 정보를 획득하고 이를 기반으로 여섯 종류의 센서 정

보를 실시간으로 획득하여 안드로이드 애플리케이션으로 전송하게 된다.

시스템에서 사용한 안드로이드 애플리케이션의 구현 결과는 그림 5와 같이 나타난다. 안드로이드 애플리케이션은 실행 후 BLE 장치 검색을 통해 OBD-II 장치를 찾고 연결을 시도하며, 연결 후 요청을 통해 실시간으로 센서 데이터를 수집하여 저장하게 된다. 또한 추가적으로 IMU, GPS, 카메라 영상 등을 함께 저장할 수 있다.

#### 4.3 실차 테스트

시스템의 동작을 확인하기 위하여 32대의 실제 차량을 통해 동작을 테스트하였다. 실험에 사용한 차종과 센서 정보를 얻기 위해 사용한 진단 프로토콜은 표 1에 나타나있다. 표에서 각 센서 데이터 수집을 위해 사용한 진단 프로토콜은 UDS(U), K-line(K), 요청하지 않은 CAN 메시지(N)로 나누어 표기하였다. 차량 제조사와 차량 모델, 연식을 바꿔가면서 실험하였고 데이터베이스에 저장된 정보를 대부분 정상적으로 조회할 수 있음을 확인하였다. 32대 중 동일한 모델을 제외하면 26종의 차종에 대한 검증을 수행하였다. 32대 중 23대는 여섯 가지 정보를 수집할 수 있었고, 8대의 차량은 다섯 가지 정보만을 수집하는 것을 확인하였다. 모닝의 경우 실험 과정에서 브레이크와 기어 포지션만 추출되는 것을 확인하였다. 즉, 제안한 차량 정보 획득 기법을 통해 스티어링 휠, 방향 지시등, 브레이크 페달, 오도미터, 운전자 안전벨트 스위치, 기어 포지션의 정보 획득 성공률은 각각 96.9%, 78.1%, 100%, 93.8%, 93.8%, 100%이다.

테스트 결과 중 일부 차량들은 특정 센서에 대한 정보 획득에 실패했는데, 정보를 추출하는 방법이 데이터베이스에 존재하지 않은 경우들이다. 시스템의 데이터베이스에 존재하지 않는 정보들은 3가지 프로토콜을 통해 데이터를 수집하는 방법을 아직 알아내지 못한 경우이다. 특히 모닝의 경우는 정보 획득에 성공한 경우보다 실패한 경우가 많았는데, 해당 연식의 차량에 대한 정보 추출이 불가능하거나 아직 방법을 알아내지 못한 경우이다. 기존 연구[20]에서는 2017년에 제조된 모닝에 대한 비요청 CAN 메시지 분석을 통해 스티어링 휠, 방향 지시등, 브레이크 페달, 기어 포지션 등의 정보 획득에 성공한 바 있으나, UDS, K-line에 대한 실험이 이루어지지 않아 결과표에서 제외하였다.

실험을 수행한 결과, 같은 제조사에서 같은 해에 만들어진 차량이더라도 데이터를 수집을 위해 사용하는

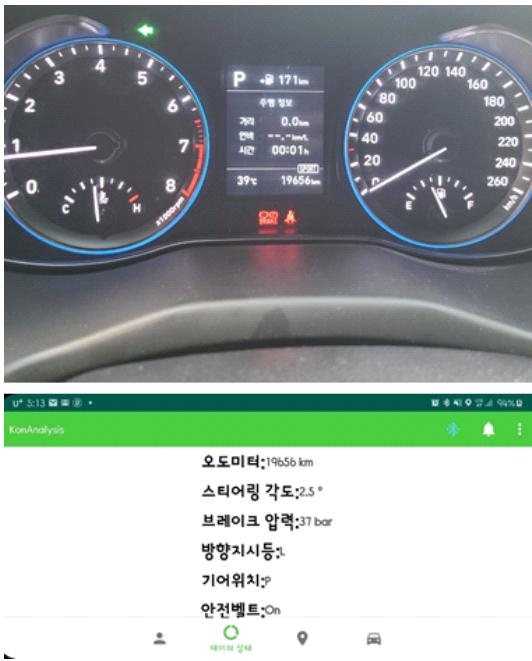


그림 5. 안드로이드 애플리케이션의 구현 결과  
Fig. 5. Android application implementation

표 1. 차량 실험을 통한 차량 정보 획득 결과 비교표  
Table 1. Comparison table of vehicle information acquisition results through vehicle experiment

Manufacturer	Model	Engine	Year	ST	TS	BP	OD	BE	GE	Manufacturer	Model	Engine	Year	ST	TS	BP	OD	BE	GE	
HYUNDAI	ACCENT	G1.6	2012	U	-	U	N	N	U	KIA	MORNING	G1.0	2013	-	-	U	-	-	U	
	ACCENT	G1.6	2013	U	-	U	N	N	U		RAY	G1.0	2014	U	-	U	U	K	U	
	ACCENT	G1.6	2016	U	-	U	N	N	U		K5	L2.0	2011	U	N	U	N	N	U	
	AVANTE	G1.6	2013	U	N	U	N	K	U		K5	L2.0	2015	U	N	U	N	N	U	
	AVANTE	G1.6	2014	U	N	U	N	K	U		K7	L3.0	2015	U	K	U	N	K	U	
	SONATA	L2.0	2013	U	K	U	N	K	U		K9	G3.8	2018	U	U	U	U	U	U	
	GRANDEUR	L3.0	2018	U	U	U	U	U	U		CARNIVAL	D2.2	2016	U	U	U	N	U	U	
	EQUUS	G5.0	2012	U	K	U	-	K	U		MOHAVE	D3.0	2016	U	U	U	N	U	U	
	KONA	G1.6	2018	U	U	U	U	U	U		RENAULT	SM3	G1.6	2016	U	U	U	U	N	U
	MAXCRUZ	D2.2	2018	U	U	U	U	U	U			SM3	G1.6	2018	U	U	U	U	N	U
	GRAND STAREX	D2.5	2014	U	-	U	U	K	U	SM6		L2.0	2017	U	N	U	N	N	U	
	GRAND STAREX	D2.5	2017	U	-	U	U	K	U	SM7		L2.0	2015	U	U	U	U	N	U	
	CHEVROLET	CRUZE	G1.4	2017	U	N	U	N	N	N		QM3	D1.5	2017	U	U	U	U	U	U
IMPALA		G2.5	2015	U	N	U	N	N	N	QM6	D2.0	2016	U	N	U	U	U	U		
SPARK		G1.0	2015	U	N	U	N	N	N	SSANGYONG	TIVOLI	D1.6	2015	U	U	U	N	N	N	
TRAX		D1.6	2016	U	N	U	N	N	N		KORANDO C	D2.2	2017	U	U	U	N	-	N	

ID나 진단 프로토콜이 다른 경우가 다수 발견되었다. 또한 제안하는 시스템은 차량에서 관심 있는 정보를 최대한 추출할 수 있도록 3가지 진단 프로토콜에 모두 대응할 수 있음을 확인하였다.

### V. 결 론

본 논문에서는 여러 차종을 고려한 데이터베이스 기반의 차량 데이터 수집 시스템을 설계하고 구현하였다. 차량 제조사나 차종별로 달라질 수 있는 차량 내부 정보 수집 기능을 위하여 데이터베이스와 차대 번호를 활용한 정보 획득 과정을 설계하고, NVS 라이브러리를 통해 키와 인코딩 된 정보를 플래시 메모리에 함께 저장할 수 있도록 구현하였다. 제안한 시스템은 차량의 종류나 관심 정보를 직접 등록해야 하는 기존 기술과는 다르게 자동으로 차종을 인식하고 정보를 수집할 수 있다는 점에서 의미가 있다.

또한, 제안한 시스템의 동작을 검증하기 위해 32대의 차량을 활용한 실증 테스트를 수행하였다. 차량 제조사는 동일하더라도 차종에 따라 데이터 수집 방법이 달랐으며, 지원하는 진단 프로토콜이 다른 경우도 존재하였다. 데이터 수집 시스템은 3가지 진단 프로토콜에 대응하여 데이터베이스에 정의된 정보를 정상적으로 수집할 수 있음을 확인하였다.

하지만, 제안하는 시스템이 사용하는 3가지 프로토콜 중 UDS와 K-line에 대한 정보 수집이 데이터베이스

스에 의존한다는 한계점이 있다. 따라서, 데이터베이스에 기입되지 않은 차량에 대한 정보 수집을 위해서는 기존의 비요청 CAN 정보의 자동 추출 과정과 같이 UDS와 K-line을 위한 추출 기법에 대한 연구가 필요하다.

본 시스템을 이용할 경우, 센서 정보 수집 방법이 공개되지 않은 다양한 국내 차량에서 내부 정보를 수집하여 연구에 활용할 수 있다. 특히 운전자의 조작 정보들을 기반으로 운전자 행동 분석이나 위험 운전 인식, 운전자 식별 등 스마트 차량 기술 분야에서 활용할 수 있다.

### References

- [1] K. Vasudevan, A. P. Das, B. Sandhya, and P. Subith, "Driver drowsiness monitoring by learning vehicle telemetry data," *10th Int. Conf. HIS 2017*, pp. 270-276, Ulsan, Korea, Jul. 2017.
- [2] S. Moon, M. Min, J. Nam, J. Park, D. Lee, and D. Kim, "Drowsy driving warning system based on GS1 standards with machine learning," *IEEE Int. Congress on Big Data*, pp. 289-296, Honolulu, USA, Jun. 2017.
- [3] Y. Zhang, M. Chen, N. Guizani, D. Wu, and V. C. Leung, "SOVCAN: Safety-oriented

- vehicular controller area network,” *IEEE Commun. Mag.*, vol. 55, no. 8, pp. 94-99, Aug. 2017.
- [4] S. H. Chen, J. S. Pan, and K. Lu, “Driving behavior analysis based on vehicle OBD information and AdaBoost algorithms,” *IMECS 2015*, pp. 18-20, Hongkong, Mar. 2015.
- [5] J. E. Meseguer, C. K. Toh, C. T. Calafate, J. C. Cano, and P. Manzoni, “Drivingstyles: A mobile platform for driving styles and fuel consumption characterization,” *J. Commun. and Netw.*, vol. 19, no. 2, pp. 162-168, Apr. 2017.
- [6] A. Sathyanarayana, S. O. Sadjadi, and J. H. Hansen, “Leveraging sensor information from portable devices towards automatic driving maneuver recognition,” *15th Int. IEEE Conf. Intelli. Transp. Syst.*, pp. 660-665, Anchorage, USA, Sep. 2012.
- [7] Y. Zheng and J. H. Hansen, “Lane-change detection from steering signal using spectral segmentation and learning-based classification,” *IEEE Trans. Intell. Veh.*, vol. 2, no. 1, pp. 14-24, Mar. 2017.
- [8] V. Leonhardt and G. Wanielik, “Recognition of lane change intentions fusing features of driving situation, driver behavior, and vehicle movement by means of neural networks,” *Advanced Microsystems for Automotive Appl.*, pp. 59-69, Aug. 2017.
- [9] V. Leonhardt and G. Wanielik, “Feature evaluation for lane change prediction based on driving situation and driver behavior,” *20th Int. Conf. Inf. Fusion*, pp. 1-7, Xi’an, China, Aug. 2017.
- [10] M. Enev, A. Takakuwa, K. Koscher, and T. Kohno, “Automobile Driver Fingerprinting,” in *Proc. Privacy Enhancing Technol.*, vol. 2016, no. 1, pp. 34-50, Jan. 2016.
- [11] B. I. Kwak, J. Woo, and H. K. Kim, “Know your master: Driver profiling-based anti-theft method,” *14th Annu. Conf. Privacy, Secur. and Trust*, pp. 211-218, Auckland, New Zealand, Dec. 2016.
- [12] ELM Electronics, *ELM327 Datasheet* (2017), Retrieved Jul. 17, 2020, from [www.elmelectronics.com/DSheets/ELM327DS.pdf](http://www.elmelectronics.com/DSheets/ELM327DS.pdf).
- [13] M. Markovitz and A. Wool, “Field classification, modeling and anomaly detection in unknown CAN bus networks,” *Veh. Commun.*, vol. 9, pp. 43-52, Mar. 2017.
- [14] M. Marchetti and D. Stabili, “READ: Reverse engineering of automotive data frames,” *IEEE Trans. Inf. Forensics and Secur.*, vol. 14, no. 4, Apr. 2019.
- [15] M. E. Verma, R. A. Bridges, and S. C. Hollifield, “ACTT: Automotive CAN tokenization and translation,” arXiv preprint arXiv:1811.07897, pp. 1-6, Nov. 2018.
- [16] M. D. Pesé, T. Stacer, C. A. Campos, E. Newberry, D. Chen, and K. G. Shin, “LibreCAN: Automated CAN message translator,” in *Proc. 2019 ACM SIGSAC Conf. Comput. and Commun. Secur.*, pp. 2283-2300, London, UK, Nov. 2019.
- [17] C. Young, J. Svoboda, and J. Zambreno, “Towards reverse engineering controller area network messages using machine learning,” in *Proc. IEEE World Forum on Internet of Things*, Apr. 2020.
- [18] M. E. Verma, R. A. Bridges, J. J. Sosnowski, S. C. Hollifield, and M. D. Iannacone, “CAN-D: A modular four-step pipeline for comprehensively decoding controller area network data,” arXiv preprint arXiv:2006.05993, Jun. 2020.
- [19] M. Zago, S. Longari, A. Tricarico, M. Carminati, M. G. Pérez, G. M. Pérez, and S. Zanero, “ReCAN - Dataset for reverse engineering of Controller Area Networks,” *Data in brief*, vol. 29, Apr. 2020.
- [20] B. Kim, D. Han, and Y. Baek, “Design and implementation of filtering-based automatic extraction system for real-time monitoring of vehicle information,” *J. KICS*, vol. 44, no. 9, pp. 1778-1787, Sep. 2019.



**김 범 준 (Beomjun Kim)**



2012년 2월 : 부산대학교 컴퓨터공학과 공학학사  
2014년 2월 : 부산대학교 컴퓨터공학과 공학석사  
2014년 3월~현재 : 부산대학교 컴퓨터공학과 박사과정

<관심분야> 지능형 자동차, 저전력 기술, 임베디드 시스템, 딥러닝

[ORCID:0000-0002-2945-1054]

**백 윤 주 (Yunju Baek)**



1990년 2월 : 한국과학기술원 전산학과 공학학사  
1992년 2월 : 한국과학기술원 전산학과 공학석사  
1997년 2월 : 한국과학기술원 전산학과 공학박사  
1999년~2002년 : NHN 기술연구소 소장

2003년~현재 : 부산대학교 컴퓨터공학과 정교수

<관심분야> RFID, WSN, RTLS, 임베디드 시스템

[ORCID:0000-0002-3873-2624]

**김 찬 우 (Chanwoo Kim)**



2019년 2월 : 부산외국어대학교 임베디드소프트웨어학과 공학학사

<관심분야> 임베디드 시스템, 하드웨어 설계, CAN 분석

[ORCID:0000-0003-4955-3071]