

초소형전기차 공유서비스를 위한 차량용 통신모듈 개발

Development of vehicle communication module for Micro-EV sharing service

권 오 훈*, 이 수 성*, 최 연 주†, 윤 장 규†, 석 수 영†
†경북IT융합산업기술원

(Oh-Hun Kwon, Soo-Sung Lee, Sung-ho Kim, Yeon-Ju Choi, Jang-Kyu Yun, Soo-Young Suk)
(†Gyeongbuk Industry Institute of IT Convergence Industry Technology)

Abstract : As the car-sharing market expands around the world, various companies are entering the market and expanding their business. Sharing services for micro electric vehicles are also receiving a lot of attention, and various research and development are being carried out accordingly. This paper deals with the development of communication modules for shared services of micro electric vehicles. The target vehicle is Maiv's Maiv M1 vehicle, which is a representative micro electric vehicle in Korea. The communication module collects location information (latitude, longitude) from the GPS sensor and provides it to the server. It also collects various vehicle data and transmits it to the server. Vehicle data includes vehicle ID, latitude, longitude, charger connection status, battery charge status, battery level, gear status, motor RPM, motor temperature, inverter temperature, inverter phase current, pedal value, driving mode, vehicle door lock status, door open status, vehicle speed, odometer, battery pack voltage, battery pack current, and battery pack error status. Each function was developed on an embedded basis using the board developed by GITC. Verification of each function was confirmed through actual vehicle-based tests.

Keywords : micro-EV(Electric Vehicle), car sharing, communication module, embedded

1. 서 론

개인의 유희자원인 주택, 자동차 등을 대여하여 수익을 얻는 새로운 형태의 비즈니스모델을 사업화한 공유경제기업이 급속도로 성장하고 있다. 소유가 필요하지 않고 접속과 공유가 경제활동의 근간이 되는 시대가 도래하고 있으며, 소비방식의 변화가

공유경제의 성장을 통해서 현실화되고 있다. 이러한 움직임은 자동차 산업에도 적용되고 있으며, 국내 기업중에는 쏘카, 그린카가 카셰어링 산업을 주도하고 있다. 카셰어링 산업은 국내뿐만 아니라 전세계적으로 지속해서 성장할 것으로 전망되고 있다.

공유형차량은 일반차량에 통신모듈이 추가 적용되어 공유서비스 관계시스템과 서로 통신하여 차량의 상태를 실시간으로 관리하고, 고객에게 차량을 대여할 수 있도록 하는 것이다. 따라서 공유차량의 통신모듈은 아래 기능을 반드시 포함하여야 한다. 첫번째는 차량과 관계 서버간의 통신 기능이다. 두번째는 차량의 위치 정보를 수집하는 기능이다. 세번째는 차량의 데이터를 수집하는 기능이며, 마지막 네 번째는 차량의 문잠금장치 및 시동을 제어하는 기능이다. 본 논문에서는 차량과 관계서버간의 통신 기능과 위치정보 수집기능에 대한 개발내용을 주로 다룬다.

*Corresponding Author (ohkwon@gitc.or.kr, powerdig@naver.com)

권오훈, 이수성, 최연주, 윤장규, 석수영: 경북IT융합산업기술원

※ 본 연구는 산업통상자원부와 한국산업기술진흥원의 “국가혁신융합단지지원사업(R&D) 성과완성형(P0015266) 서비스 e-모빌리티를 위한 초소형전기차 부품개발 및 실차 평가기술 개발”의 지원을 받아 수행된 연구결과임

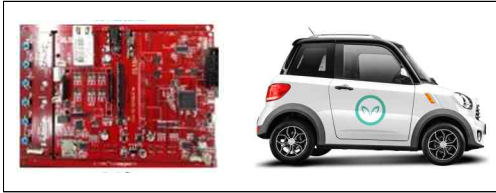


그림 1. 통신모듈 보드 및 대상차량

Fig. 1. Communication module board and target vehicle

공유서비스 통신모듈의 베이스보드는 경북IT융합산업기술원에서 자체 개발한 보드를 활용하였다. 공유서비스 통신모듈의 적용차량은 국내 초소형 전기차 제조사인 마이브사의 마이브_M1으로 선정하였다. 그리고 공유서비스 서버는 플라잉사에서 개발한 서버시스템을 적용하였다. 서버간 통신은 무선통신방식(LTE)을 적용하였고, 위치정보는 GPS센서를 통해서 수집하였다. 그리고 각 기능에 대해 실차기반의 테스트를 통해서 기능검증을 완료하였다.

II. 공유서비스 통신모듈 기능개발

1. 시스템 구성

공유서비스 통신모듈의 베이스보드는 경북IT융합산업기술원에서 자체 개발한 보드를 적용하였다. 개발보드의 MCU는 ARM Cortex-M4 STM32F4가 탑재되어 있으며, 인터페이스는 CAN, UART, GPIO로 구성되어 있다. CAN은 차량의 데이터수집 및 차량제어를 위한 인터페이스며, UART는 LTE모뎀 및 GPS 센서의 데이터수집 및 제어를 위한 인터페이스이다. 그리고 GPIO는 보드에 내장된 LED 및 센서전원을 제어하기 위해 사용된다. 아래 그림1은 공유서비스 시스템을 도식화한 그림이다.

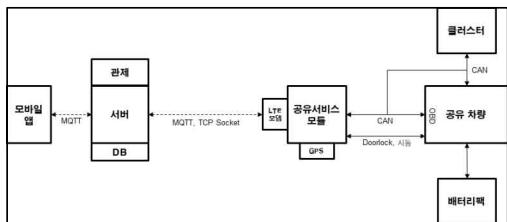


그림 2. 시스템 구성도

Fig. 2. System Diagram

2. 무선통신 기능개발

공유서비스 통신모듈의 모뎀은 일반적인 LTE모뎀은 적용이 불가능하기 때문에 MQTT 프로토콜을 지원하는 임베디드 타입의 LTE모뎀을 선정하였다. 선정된 모뎀은 Murata사의 Type1SC를 탑재한 CodeZoo사의 CAT.M1 이다. CAT.M1 모뎀은 UART 인터페이스를 기반으로 AT Command로 제어 된다. 따라서 무선통신을 구현하기 위한 모뎀 제어 절차 및 Command를 분석을 통해 펌웨어 개발을 수행하였다. 모뎀제어 로직은 “모뎀부팅 → 기본설정값 입력 → 네트워크 연결 → 노드정의 → 프로토콜 설정 → MQTT 서버접속 → 데이터모드 → 연결해제” 순서로 정의된다. 개발보드의 펌웨어는 모뎀의 정의된 로직 순서로 동작이 되도록 개발하였다. 통신모듈과 서버간인 통신 프로토콜은 MQTT(Message Queuing Telemetry Transport) 프로토콜을 적용하였다. 통신모듈에서 서버로 송신하는 데이터 항목은 총 20가지 항목으로 상세 정보는 아래 그림2와 같다.

표 1. 차량 데이터 리스트

Table 1. Vehicle Data list

order	data name	data 설명	data type	unit	example	explanation
1	car_id	차량 ID	string		MAX_3679	차량명, 차량번호
2	latitude	GPS 위도	number		37.000000	value
3	altitude	GPS 고도	number		127.000000	value
4	charge_connection	충전기 연결상태	number		0 or 1	0 : Disconnect 1 : Connect
5	charging_status	배터리 충전상태	number		0 - 3	0 : No charge 1 : Charging failure 2 : Charging 3 : Charging is complete
6	battery_soc	배터리 잔량	number	%	961	soc = 96.1%
7	gear_status	기어상태	number	Neutral Drive Reverse	0 or 1 or 2	0 : N 1 : D 2 : R
8	motor_rpm	모터회전수	number	rpm	3000	value
9	motor_temp	모터온도	number	°C	85	value - 50 temp = 35
10	mis_temp	인버터 온도	number	°C	75	value - 50 temp = 25
11	Phase current	상전류	number	A	50	value
12	acc_off_pedal	과달린	number	%	30	value
13	drive_mode	주행모드	number	ECC, Sports	0 or 1	0 : Sports 1 : ECC
14	doorlock_status	차량문 잠금상태	number		0 or 1	0 : lock 1 : unlock
15	door_status	차량 문 열림/닫힘	number		0 or 1	0 : door close 1 : door open
16	speed	차량속도	number	km/h	70	value
17	odometer	총 주행거리	number	km	3156128	value / 10 315612.8 km
18	battery_voltage	배터리전압	number	V	706	value / 10 70.6 V
19	battery_current	배터리전류	number	A	879	value / 100 = 50 37.9 A
20	battery_failure	배터리팩 오류상태	number		0 or 1 or 2	0 : No fault 1 : First level failure 2 : Second level failure

MQTT서버의 기본 데이터 송수신 형식은 Json 타입으로 정의되어 있기 때문에, 통신모듈에서 차량 데이터를 서버로 송신할 때 데이터를 Json타입으로 변환하여 송신하여야 한다. 아래 그림3은 차량데이터들을 Json타입으로 변환한 펌웨어 코드이다.

```
// flying server data send format
sprintf(typeData_send_buf, "{\"car_id\": \"%s\", \"date_time\": \"%s\",
\"doorlock_status\": %d, \"charge_connection\": %d, \"charging_status\": %d,
\"battery_soc\": %d, \"latitude\": %d, \"longitude\": %d, \"gear_status\": %d,
\"motor_rpm\": %d, \"motor_temp\": %d, \"mcu_temp\": %d, \"phase_current\": %d,
\"accel_pedal\": %d, \"drive_mode\": %d, \"door_status\": %d, \"speed\": %d,
\"odometer\": %d, \"battery_voltage\": %d, \"battery_current\": %d,
\"battery_failure\": %d}\"\\n\\n");
Car_ID: "GIT Test 0218", OBD_PeepData.doorlock_state, OBD_BmsData.charger_connection,
OBD_BmsData.charging_status, OBD_BmsData.battery_soc, gps_data.latitude,
gps_data.longitude, OBD_McuData.gear_status, OBD_McuData.motor_rpm, OBD_McuData.motor_temp,
OBD_McuData.mcu_temp, OBD_McuData.mcu_current, OBD_McuData.accel_pedal, OBD_McuData.drive_mode,
OBD_IcuData.door_state, OBD_IcuData.speed, OBD_IcuData.odometer, OBD_BmsData.battery_voltage,
OBD_BmsData.battery_current, OBD_BmsData.battery_failure );
```

그림 3. 데이터 타입 변환(Json)
Fig. 3. Data type transformation(Json)

통신모듈에서 서버로 차량데이터를 송신하는 주기는 5초로 정의하였으며, 서버 데이터베이스에 저장된 시간을 통해서 5초 주기로 데이터가 저장된 것을 확인하였다. 아래 그림4는 서버 데이터베이스에 저장된 차량데이터를 확인한 내용이다.

key	Value	Type
(554) Object["624e9b054664f5b35236af"]	(22 Fields)	Object
(555) Object["624e9b054664f5b35236b0"]	(22 Fields)	Object
(556) Object["624e9b054664f5b35236b1"]	(22 Fields)	Object
(557) Object["624e9b054664f5b35236b2"]	(22 Fields)	Object
(558) Object["624e9b054664f5b35236b3"]	(22 Fields)	Object
(559) Object["624e9b054664f5b35236b4"]	(22 Fields)	Object
(560) Object["624e9b054664f5b35236b5"]	(22 Fields)	Object
id	Object["624e9b054664f5b35236b5"]	Object
car_id	MAV_9177	String
date_time	070422 080808	String
doorlock_status	0	Int32
charge_connection	0	Int32
charging_status	0	Int32
battery_soc	72	Int32
latitude	3551.72977	Double
longitude	12850.11482	Double
gear_status	1	Int32
motor_rpm	0	Int32
motor_temp	69	Int32
mcu_temp	70	Int32
phase_current	0	Int32
accel_pedal	0	Int32
drive_mode	0	Int32
door_status	1	Int32
speed	0	Int32
odometer	1529	Int32
battery_voltage	787	Int32
battery_current	530	Int32
battery_failure	0	Int32

그림 4. 서버 데이터베이스에 저장된 데이터
Fig. 4. Data stored in the server database

3. 위치정보 수집기능 개발

차량의 위치정보는 GPS센서를 통신모듈에 적용하여 위도, 경도 데이터를 수집하였다. GPS센서는 정확도가 높은 센서일수록 비용이 크게 증가되기 때문에 위치정확도를 고려하여 적합한 센서를 선정하였다. 선정된 센서는 ASCEN사의 GPS610으로 낮은 전력소비와 NMEA (National Marine Electronics Association) 프로토콜을 지원한다. 데이터송수신 인터페이스는 UART로 적용하였다. 아래 그림5는 GPS센서의 수집데이터에 대한 그림이다.

```
$GPGSV,3,1,12,10.51,208.46,12.44,055.30,22.02,325,,23.25,177.34+73
$GPGSV,3,2,12,24.15,066.26,25.71,112.28,26.04,211.38,29.08,137.21+78
$GPGSV,3,3,12,31.33,266.42,32.61,328.38,39,,36.50,48,183.36+41
$GPRMC,3551.71992,N,12850.08854,E,0.01225,0.0,A,D=65
$GPRMC,041226.00,A,3551.72001,N,12850.08851,E,0.039,,221221,,D=70
$GPVTG,T,,M,0.039,N,0.073,K,D=28
$GPGGA,041226.00,3551.72001,N,12850.08851,E,2.07,1.13,122.9,M,23.4,M,0.0000+59
$GPGSA,A,3,10,32,12,23,24,31,25,,,,,1.91,1.13,1.54+09
$GPGSV,3,1,12,10.51,208.47,12.44,055.30,22.02,325,,23.25,177.34+72
$GPGSV,3,2,12,24.15,066.26,25.71,112.27,26.04,211.37,29.08,137+78
$GPGSV,3,3,12,31.33,266.42,32.61,328.38,39,,36.50,48,183.36+41
$GPRMC,3551.72001,N,12850.08851,E,0.01226,0.0,A,D=63
$GPRMC,041227.00,A,3551.72013,N,12850.08849,E,0.041,,221221,,D=74
$GPVTG,T,,M,0.041,N,0.076,K,D=22
$GPGGA,041227.00,3551.72013,N,12850.08849,E,2.07,1.13,123.6,M,23.4,M,0.0000+5C
$GPGSA,A,3,10,32,12,23,24,31,25,,,,,1.91,1.13,1.54+09
$GPGSV,3,1,12,10.51,208.47,12.44,055.30,22.02,325,,23.25,177.34+72
$GPGSV,3,2,12,24.15,066.26,25.71,112.27,26.04,211.37,29.08,137+78
$GPGSV,3,3,12,31.33,266.42,32.61,328.38,39,,36.50,48,183.36+41
$GPRMC,3551.72013,N,12850.08849,E,0.01227,0.0,A,D=68
$GPRMC,041228.00,A,3551.72022,N,12850.08844,E,0.023,,221221,,D=70
$GPVTG,T,,M,0.023,N,0.042,K,D=21
$GPGGA,041228.00,3551.72022,N,12850.08844,E,2.07,1.13,124.3,M,23.4,M,0.0000+5E
$GPGSA,A,3,10,32,12,23,24,31,25,,,,,1.91,1.13,1.54+09
```

그림 5. GPS 센서 로우데이터
Fig. 5. GPS Sensor low-data

GPS센서의 데이터중 위도, 경도 데이터를 추출하기 위해서 GPRMC(Global Positioning Recommended Minimum data) 데이터를 활용하였다. GPRMC는 위도, 경도, 고도, 속도, 방향, 시간정보를 포함하고 있다. 통신모듈에서는 GPRMC 메시지를 수집하여 필요한 정보를 파싱하여 데이터를 확보한다.

```
ui.c: 56 4354:000 [!] Showing_IDLE_S(202) : UI_UNIT_SEC_F
[GPS] GPRMC,055540.00,A,3551.74809,N,12850.12398,E,0.007,,140721,,A,V+1C
$,70
[GPS time] 055540.00
[GPS latitude] 3551.74809
[GPS longitude] 12850.12398
[GPS date] 140721
```

그림 6. GPS 센서 데이터 파싱
Fig. 6. GPS Sensor Data Parsing

III. 공유서비스 통신모듈 테스트

1. 통신모듈 - 서버간 통신 테스트

테스트를 위해 공유서비스 통신모듈을 실차에 연결하여 실차데이터가 서버데이터베이스에 일정주기로 저장되는지를 확인하였다.



그림 7. 통신모듈 차량연결
Fig. 7. Connecting communication module

테스트결과 차량 데이터가 5초주기로 서버 데이터베이스에 저장되는 것을 확인하였다. 또한 저장된 데이터가 실제 차량상태와 동일한지를 검증하였다.

id	car_id	data_time	doorlock_status	charge_connect	charging_status	battery_soc	latitude	longitude
S83	MAV_9177	070422 081003	(x) 0	(x) 0	(x) 0	(x) 72	3551.7287	12850.11359
S84	MAV_9177	070422 081008	(x) 0	(x) 0	(x) 0	(x) 72	3551.72875	12850.11414
S85	MAV_9177	070422 081013	(x) 0	(x) 0	(x) 0	(x) 72	3551.7287	12850.11471
S86	MAV_9177	070422 081018	(x) 0	(x) 0	(x) 0	(x) 72	3551.72873	12850.11532
S87	MAV_9177	070422 081023	(x) 0	(x) 0	(x) 0	(x) 72	3551.72873	12850.11561
S88	MAV_9177	070422 081028	(x) 0	(x) 0	(x) 0	(x) 72	3551.7289	12850.11606
S89	MAV_9177	070422 081033	(x) 0	(x) 0	(x) 0	(x) 72	3551.7301	12850.11652
S90	MAV_9177	070422 081038	(x) 0	(x) 0	(x) 0	(x) 72	3551.73023	12850.11688
S91	MAV_9177	070422 081043	(x) 0	(x) 0	(x) 0	(x) 72	3551.73026	12850.11727
S92	MAV_9177	070422 081048	(x) 0	(x) 0	(x) 0	(x) 72	3551.73013	12850.11748
S93	MAV_9177	070422 081053	(x) 0	(x) 0	(x) 0	(x) 72	3551.72993	12850.11749
S94	MAV_9177	070422 081058	(x) 0	(x) 0	(x) 0	(x) 72	3551.72982	12850.11719
S95	MAV_9177	070422 081103	(x) 0	(x) 0	(x) 0	(x) 72	3551.72877	12850.11683

그림 8. 서버 데이터베이스
Fig. 8. Server database

2. 위치정보 정밀도 테스트

위치정밀도는 테스트는 기준 위치점을 정하고 해당 위치점에서 초정밀 GPS장비를 활용하여 위도, 경도에 대한 기준데이터를 확보한다. 이후 테스트할 통신모듈 및 GPS센서를 기준점에 위치시키고 데이터를 수집후 기준데이터와 비교하여 위치 정밀도를 계산한다.



그림 9. GPS센서 정밀도 테스트
Fig. 9. GPS sensor precision test

초정밀 GPS장비(AKT980R)의 정밀도는 약 2cm이하로 매우 정확하기 때문에 기준점에 대한 데이터는 신뢰할수 있다고 판단하였다. 테스트는 5 회 수행하였고 결과는 아래 표2와 같다.

표 2. GPS 센서 정밀도 테스트 결과
Table 2. GPS sensor precision test result

테스트 순번	거리 차이 결과 (단위: m)
Test 1	1.1274
Test 2	1.2746
Test 3	1.3621
Test 4	1.3825
Test 5	1.4186
AVG	1.3130

테스트 결과 약 1.313m의 정밀도를 나타내었으며, 목표치인 5m이하의 정확도 보다 훨씬 더 정확한 결과를 나타내었다.

IV. 결론

전 세계적으로 차량공유 시장이 확대됨에 따라 기존의 차량공유 업체 및 완성차 업체, ICT 기업이 앞다투어 이 시장에 뛰어들고 관련 사업을 확장하고 있다. 또한, 자율주행기술 등 신기술의 발전과 더불어 글로벌 차량공유 시장은 더 확대될 것으로 전망된다. 그에따라 공유서비스를 위한 모빌리티용 통신모듈의 기술도 다양하게 개발되고 있으며, 앞으로 다양한 분야에 활용할 수 있을 것으로 여겨진다. 예를들어 모빌리티의 정보를 실시간으로 수집하는 기능은 차량관제시스템에 적용할 수 있으며, 축적된 데이터를 바탕으로 차량고장진단 및 예측시스템에 활용할 수도 있다. 그리고 통신기술을 바탕으로 원격제어기능도 적용할 수 있을 것이다. 이러한 기능들을 적용하기 위해서는 실제 차량의 액츄에이터를 제어하기 위한 기술개발 지속적으로 연구되어야 할 것이다. 본 연구에서 수행한 결과물이 향후에도 다양한 분야에 적용하여 활용될 수 있을 것으로 사료된다.

참고 문헌

- [1] 이수성, 권오훈, 석수영, “초소형전기차용 저가형 클러스터 요구사항 분석 및 설계”, 2020 대한임베디드공학회 추계학술대회, Interactive Poster Session A, No.24, 2020
- [2] 이수성, 권오훈, 서효덕, 김성호, 윤장규, 석수영, “초소형 e-모빌리티 공유서비스 정밀위치 데이

터 수집 시스템 개발 및 분석”, 2021 대한임베디드공학회 ISET 2021, Interactive Session B, No.18, 2020

- [3] 정차근, “CAN 통신 프로토콜에 의한 자동차 신호 및 센서 제어 시스템의 개발”, 신호처리시스템학회 논문지, 제3권 3호, 2002
- [4] 이호원, 기석철, “초소형 전기차를 이용한 개방형 자율주행 플랫폼 개발”, 한국자동차공학회논문집 제26권 제4호, 2018