Design and Implementation of Distributed Charge Signal Processing Software for Smart Slow and Quick Electric Vehicle Charge

Tae Uk Chang[1], Young Su Ryu[2], Seul Ki Song[2], Ki Won Kwon[2] and Jong Ho Paik[1*]
[1] Department of Multimedia, Seoul Women's University
Seoul, 01797 Republic of Korea
[e-mail: itchangtuk@gmail.com, paikjh@swu.ac.kr]
[2] Korea Electronics Technology Institute
Seongnam-si, 13509 Republic of Korea
[e-mail: {fernwehist, seulkiemma, kwonkw}@keti.re.kr]
*Corresponding author: Jong Ho Paik

Abstract

As environmental pollution and fossil fuel energy problems from fuel vehicle have occurred, the interest of electric vehicle(EV) has increased. EV industry and energy industry have grown dynamically in these days. It is expected that the next generation of primary transportation will be EV, and it is necessary to prepare EV infra and efficient energy management such as EV communication protocol, EV charge station, and smart grid. Those EV and energy industry fields are now on growth. Also, the study and development of them are now in progress. In this paper, distributed charge signal processing software for smart slow and quick EV charge is proposed and designed for dealing with EV charge demand. The software consists of smart slow and quick EV charge schedule engine and EV charge power distribution core. The software is designed to support two charge station types. One is normal EV charge station and the other is bus garage EV charge station. Both two types collect the data from EV charge stations, and then analyze the collected data. The software suggests optimized EV charge schedule and deliveries EV charge power distribution information to power switchboard system, and the designed software is implemented on embedded system. It is expected that the software provides efficient EV charge schedule.


---

1. Introduction

EV is one of environment-friendly transprotations. EV can reduce envrionmental pollution and fossil fuel energy problems caused by fuel vehicle[1]. As global warming has been going on, the interest and demand of EV have increased. Many countries of the world support EV development and study[2]. In addtion, they provide the support fund policy for people who buy EV[3]. The next generation of primary transportaion will be EV, so the construction of EV infra and efficient energy management technology will be necessary. **Fig. 1** and **2** show annual plug-in EV(PEV) sales forecast and sales and revenue forecast of EV chargers respectively[4, 5].
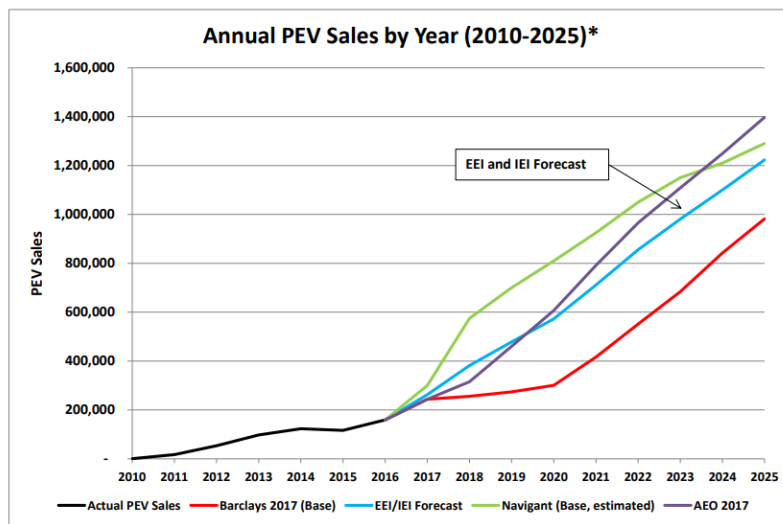
*Includes battery electric vehicles and plug-in hybrid electric vehicles
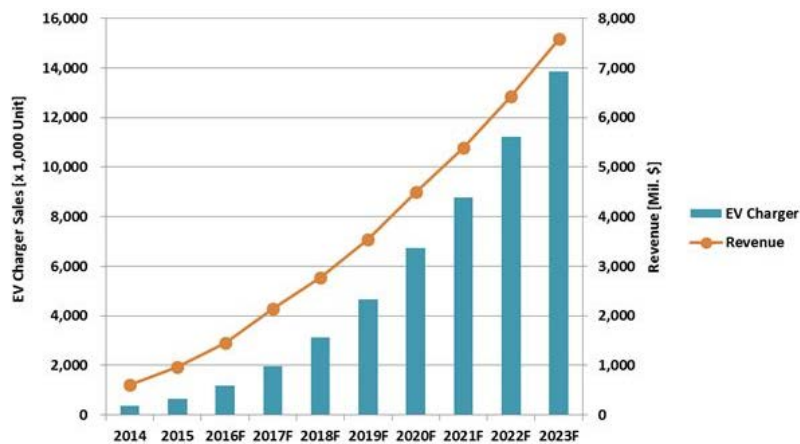
Fig. 1. Annual PEV sales forecast

Fig. 2. Sales and revenue forecast of EV chargers(source: SNE research)

In this paper, distributed charge signal processing software for smart slow and quick EV charge is proposed for dealing with EV charge demand to provide efficient EV charge power. The designed software supports two charge station types. One is normal EV charge station which is used for general EV with slow and quick charge. The other is bus garage EV charge station which is used for EV bus with quick charge. Both two types are designed to collect the data from EV charge stations such as EV charge demand power, charge finish time, charge fee per time, and so on. The designed software analyzes the collected data, and then suggests optimized EV charge schedule that consists of EV charge power distribution and time. After that, the software deliveries the result to power switchboard system which conducts to distribute the EV charge power. The software is implemented on embedded system.

In section 2, the overview of smart slow and quick EV charge system is described. The system consists of EV charge power switchboard system, control system, and smart EV charge schedule engine system. In section 3, smart charge schedule and communication between the designed software and other systems are designed. In section 4, the implementation results are shown, and section 5 presents the conclusions of the proposed software.

2. System Overview

2.1 Smart Slow and Quick EV Charge System

Smart slow and quick EV charge system mainly consists of three parts. Three parts are EV charge power switchboard system, control system, and smart EV charge schedule engine system as illustrated in **Fig. 3**.
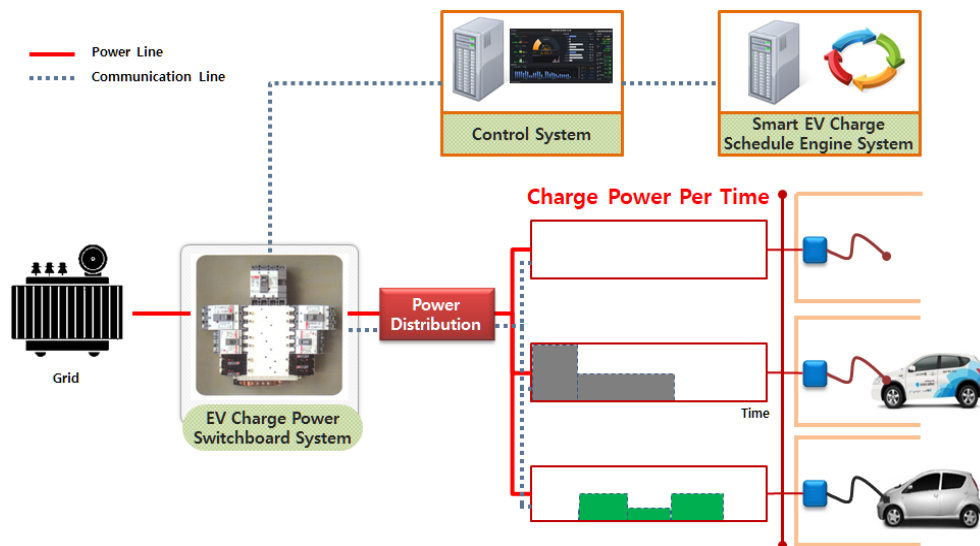

Fig. 3. Structure of smart slow and quick EV charge system

EV charge power switchboard system is designed to conduct to distribute EV charge power to each charge port. For dealing that, the switchboard system uses ISO/IEC 15118 to get EV information from EVs and to communicate with other systems[6]. **Table 1** shows the data set for the switchboard system based on ISO/IEC 15118.

Table 1. Data set for EV charge power switchboard system based on ISO/IEC 15118

Communication	Input Data
EV – Switchboard system	Charge control data
	Contract
	EV power supply ID
	Measurement information
	Charge plan
	Safety monitoring
	Safety monitoring information
EV – DB server	Charge information
	Charge fee information
	Charge payment information
EV – Business system operator	Additional Service

The switchboard system sends data such as EV charge demand power, charge completion time, charge fee per time, and so on to control system, and receives EV charge schedule from control system. After receiving the schedule, the system conducts power distribution signal processing with the schedule by using EV charge power distribution. Control system is located between EV charge power switchboard system and smart EV charge schedule engine system. The control system conducts communication between them and provides the entire monitoring status of EV charge station. Smart EV charge schedule engine analyzes the data received from control system, and calculates optimized EV charge schedule.

2.2 EV Charge Power Switchboard System, Control System, and Smart EV Charge Schedule Engine

Smart EV charge system main parts, EV charge power switchboard system, control system, and smart EV charge schedule engine, use two methods to communicate. Communication between EV charge power switchboard system and control system uses transmission control protocol/internet protocol(TCP/IP). The other communication between control system and smart EV charge schedule engine system uses hypertext transfer protocol(HTTP) with representational state tranfer(REST) application programming interface(API). The communication structure of smart EV charge system main parts is illustrated in **Fig. 4**.

Fig. 4. Communication structure of smart EV charge system main parts

EV charge power switchboard consits of power switchboard core and EV charge power distribution core. Power switchboard core communicates with outside networks such as control system and EV charge ports by using TCP/IP and uses local TCP/IP network to communicate with EV charge power distribution core. When power switchboard core sends

EV charge schedule data from control system to EV charge power distribution core, EV charge power distribution core translates the data and then sends the translated data to power swtichboard core. After power switchboard core receives the translated data, it commands each EV charge port how much and long to charge EV. The structure of EV charge power switchboard is illustrated in **Fig. 5**.
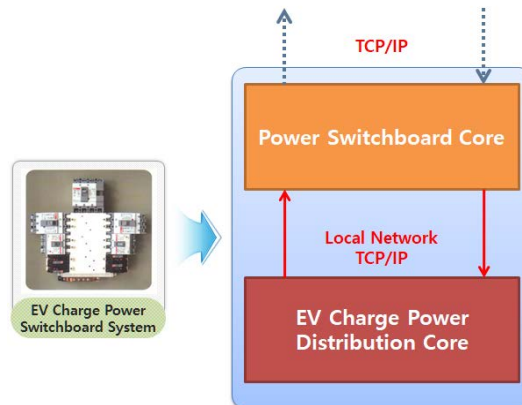
Fig. 5. Structure of EV charge power switchboard

Control system consists of server and client. The server communicates with EV charge power switchboard system by using TCP/IP, and the client communicates with smart EV charge schedule engine system by using HTTP RESTful. The server provides entire EV charge status monitoring. The client sends the data for calculating smart EV charge schedule to smart EV charge schedule engine system by using HTTP request POST method with JSON data structrue. After sending the data, the client gets the response which is filled with calculated smart EV charge schedule by using JSON data structure. The structure of control system is illustrated in **Fig. 6**.
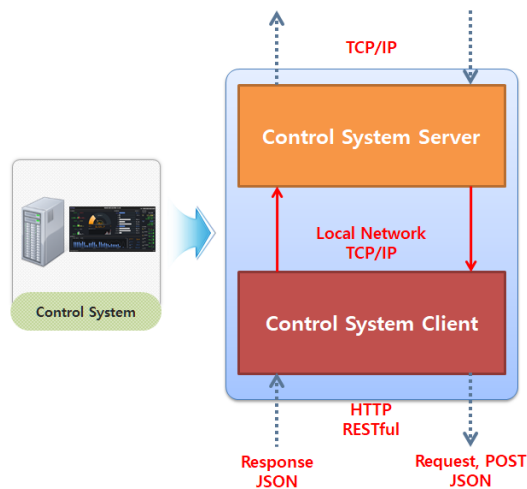
Fig. 6. Structure of control system

Smart EV charge schedule engine system consists of smart EV charge schedule engine, normal EV scheduler, bus garage EV scheduler, and simulator. When smart EV charge schedule engine receives HTTP request message from control system, the schedule engine launches the appropriate scheduler with received data, and then sends the response message filled with the calculated EV charge schedule to the control system. The simulator provides EV charge schedule simulation by user input data. The structure of smart EV charge schedule engine system is illustrated in **Fig. 7**.
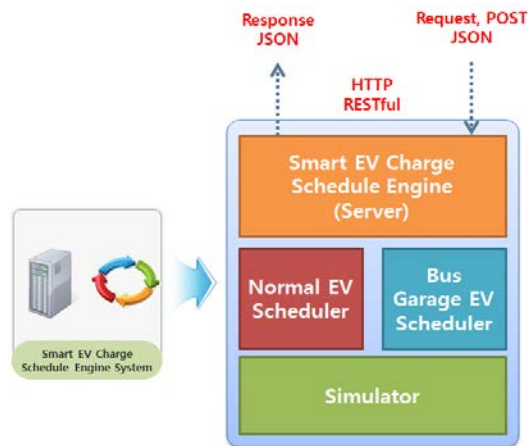

Fig. 7. Structure of smart EV charge schedule engine system

3. Design

3.1 EV Charge Schedule Models

EV charge schedule models are designed as normal EV charge schedule and bus garage EV charge schedule. The difference of two schedule models is the specific of EV charge station basically. However, the fundamental difference is whether the smart EV charge schedule engine system could get much information from EV or not. EV charge communication protocol standard, ISO/IEC 15118, has been published, however, most EVs have not adapted to ISO/IEC 15118 yet. Normal EV charge schedule model is designed for simple input data and the other schedule model is designed for various input data about EV.

Mathematical normal EV charge schedule model gets only the average number of EVs, the average charged power, and EV charge station data as input. Therefore, this mathematical model is designed to minimize EV waiting time for charge by using queueing theory. Queueing theory plays a major role in establishing dynamic traffic model[7]. This model decides the number of charge ports for focused charge power. The focused charge power ports provide maximum charge power capacity. For example, in case of slow EV charge, the ports provide 7kW each. The other ports provide uniformly distributed power of the rest of charge power capacity. Poisson process is used for arrival average EV, and exponential distribution is used for charging time. **Fig. 8** and **Table 2** show the structure of normal EV charge scheduler and input data for normal EV charge scheduler respectively.
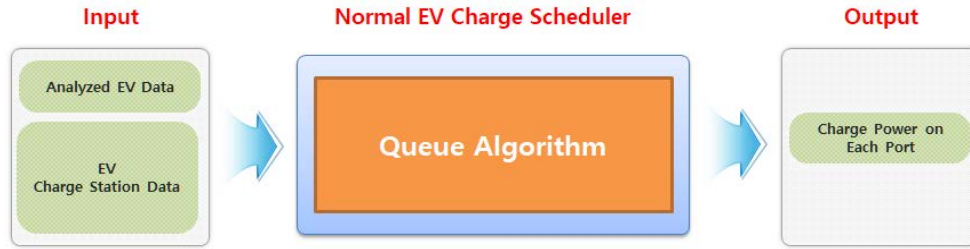
Fig. 8. Structure of normal EV charge scheduler

Table 2. Input data for normal EV charge scheduler

Group	Input Data
Analyzed EV Data	Average Number of Arrival EVs
	Average Charged Power
EV Charge Station Data	Total Number of Charge Ports
	Total Power per Time
	Max Power per Time on Each Charge Port
	Minimum Power per Time on Each Charge Port
	Blocking Probability per Time

Mathematical bus garage EV charge schedule model gets various data about EV such as EV bus schedule and information, EV charge fee, and EV charge station information. This mathematical model is designed to maximize the total EV charge power, and minimize charging time and fee by using mixed integer linear programming(MILP). Several studies with EV charge schedule by using MILP have been conducted[8-10]. **Fig. 9** and **Table 3** show the structure of bus garage EV charge scheduler and input data for bus garage EV charge scheduler respectively.
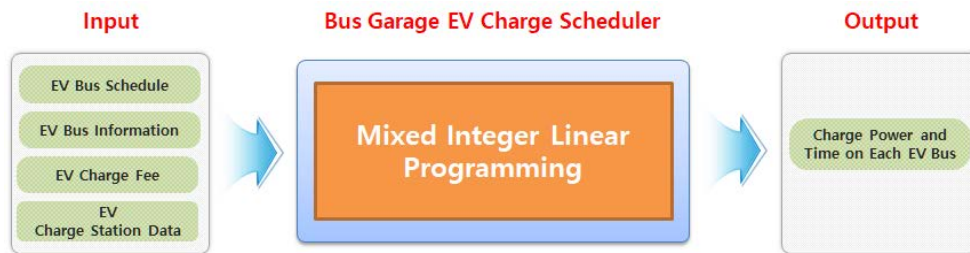

Fig. 9. Structure of bus garage EV charge scheduler

Table 3. Input data for bus garage EV charge scheduler

Group	Input Data
EV Bus Schedule	Arrival Time on Each Bus
EV Bus Information	Max Available Charge Power per Time on Each Bus
	Minimum Charge Power per Time on Each Bus Demand
	Battery Life Time
EV Charge Fee	Charge Fee per Time
EV Charge Station Data	Total Number of Charge Ports
	Total Power per Time
	Max Power per Time on Each Charge Port
	Minimum Power per Time on Each Charge Port

3.2 Smart EV Charge Schedule Engine System

Smart EV charge schedule engine system consists of smart EV charge schedule engine, normal EV charge scheduler, bus garage EV charge scheduler, and simulator. Those are designed in Python language. Flask server is used for smart EV charge schedule engine. Designed APIs for smart EV charge schedule are shown in **Table 4**.

Table 4. APIs for smart EV charge schedule engine

API	Method	Direction	Period	Description
charge_start			Event	EV charge start info.
charge_stop		To smart EV	Event	EV charge stop info.
status_update	POST	charge schedule	Event	EV charge status info.
bus_schedule		engine	Init. and once a day	Bus schedule and charge fee info.

Table 5 and **Table 6** show designed HTTP request and response messages of charge_start API by using JSON

Table 5. HTTP request message to smart EV charge schedule engine of charge_start API

Body Key	Type	Direction
ChargerID	String	ID of charge port, Station ID(8) + Port ID(2)
ReqDate	String	Date of request
ChargeMode	String	Mode of charger, 0:Run, 1:Stop, 2:Error Run(Charger Power On && Charger No Error) Error(Charger Error || Charger Disconnected)
ChargerStatus	String	Status of charger, 0:Idle, 1:Charging, 2:Error
CardNo	String	Card information
ChargeReq	String	Requested amount of charge
ChargeWatt	String	Requested amount of charge power
ChargeBill	String	Requested amount of charge fee
ChargeMethod	String	Method of pay
BattStatus	String	Battery status
BattSOCPercent	String	Battery SOC percentage
BattMAXSOC	String	Battery max SOC
BattSOC	String	Current battery SOC
BattVolt	String	Current battery voltage
BMSVer	String	BMS software version
RemainTime	String	Remain time for charge
AvailableLineCurrent	String	Available charge power from port

Table 6. HTTP response message to control system of charge_start API

Body Key	Type	Direction
Result	String	Response code
ChargeType	Int	EV charge type
SmartType	Int	Smart charge type
ChargeTimeUnit	Int	Unit time of charge

MaxNumPort	String	Max number port for focused charge
MaxChargePower	String	Max charge power for focused ports
ExpWaitTime	String	Expected EV waiting time to start
ExpWaitProb	String	Expected EV waiting probability to start
CarPerTime	String	Average number of EV per Time
ChargePowerPerTime	String	Average charge power per Time
BusChargeSchedule	String	Charge schedule for bus garage

Table 7 and **Table 8** show designed HTTP request and response messages of charge_stop API by using JSON

Table 7. HTTP request message to smart EV charge schedule engine charge_stop API

Body Key	Type	Direction
ChargerID	String	ID of charge port
ReqDate	String	Date of request
ChargeMode	String	Mode of charger
ChargerStatus	String	Status of charger
ChargeEnergy	String	Charged power from port
CardNo	String	Card information
BMSEnergy	String	Charged power on BMS
ChargingTime	String	Total time for charge
ChargeBill	String	Total fee for charge
ChargeMethod	String	Method of pay
ChargeStatus	String	Status of charger
ChargeStartTime	String	Charge start time

Table 8. HTTP response message to control system of charge_stop API

Body Key	Type	Direction
Result	String	Response code
Message	String	Error logs

Table 9 and **Table 10** show designed HTTP request and response messages of status_update API by using JSON

Table 9. HTTP request message to smart EV charge schedule engine status_update API

Body Key	Type	Direction
ChargerID	String	ID of charge port
ReqDate	String	Date of request
ChargeMode	String	Mode of charger
ChargerStatus	String	Status of charger
ChargeEnergy	String	Charged power from port
CardNo	String	Card information
ChargeReq	String	Requested amount of charge
ChargeWatt	String	Requested amount of charge power
ChargeBill	String	Requested amount of charge fee
ChargeMethod	String	Method of pay
CurrentEnergy	String	Amount of charged power
CurrentPrice	String	Amount of charge fee
CurrentUnitPrice	String	Current charge unit fee
BattStatus	String	Battery status

BattSOCPercent	String	Battery SOC percentage	
BattMAXSOC	String	Battery max SOC	
BattSOC	String	Current battery SOC	
BattVolt	String	Current battery voltage	
BattTemp	String	Current battery temperature	
BMSVer	String	BMS software version	
RemainTime	String	Remain time for charge	

Table 10. HTTP response message to control system of status_update API

Body Key	Type	Direction
Result	String	Response code
Message	String	Error logs

Table 11 and **Table 12** show designed HTTP request and response messages of bus_schedule API by using JSON

Table 11. HTTP request message to control system of bus_schedule API

Body Key	Type	Direction
ChargerID	String	ID of charge port
ReqDate	String	Date of request
StartnEndTime	String	Bus garage start and finish time
ChargeTimeUnit	String	Charge unit time
ChargeFeePerTime	String	Charge unit fee per each time
BusSchedule	String	Bus driving schedule

Table 12. HTTP response message to control system of bus_schedule API

Body Key	Type	Direction
Result	String	Response code
Message	String	Error logs

3.3 EV charge Power Distribution Core

EV charge power distribution core communicates with power switchboard core by using local TCP/IP network. After initialization of EV charge power distribution core and power switchboard core, power switchboard core requests translated EV charge schedule to EV charge distribution core. When EV comes at station and starts charging, power switchboard core sends start notice information to EV charge power distribution core. Power switchboard core sends status notice information periodically to EV charge power distribution core during charging EV. After charging, power switchboard core sends stop notice information to EV charge power distribution core. EV charge power distribution core is designed in C and C++ languages. **Fig. 10** shows designed communication sequence between EV charge power distribution core and power switchboard core
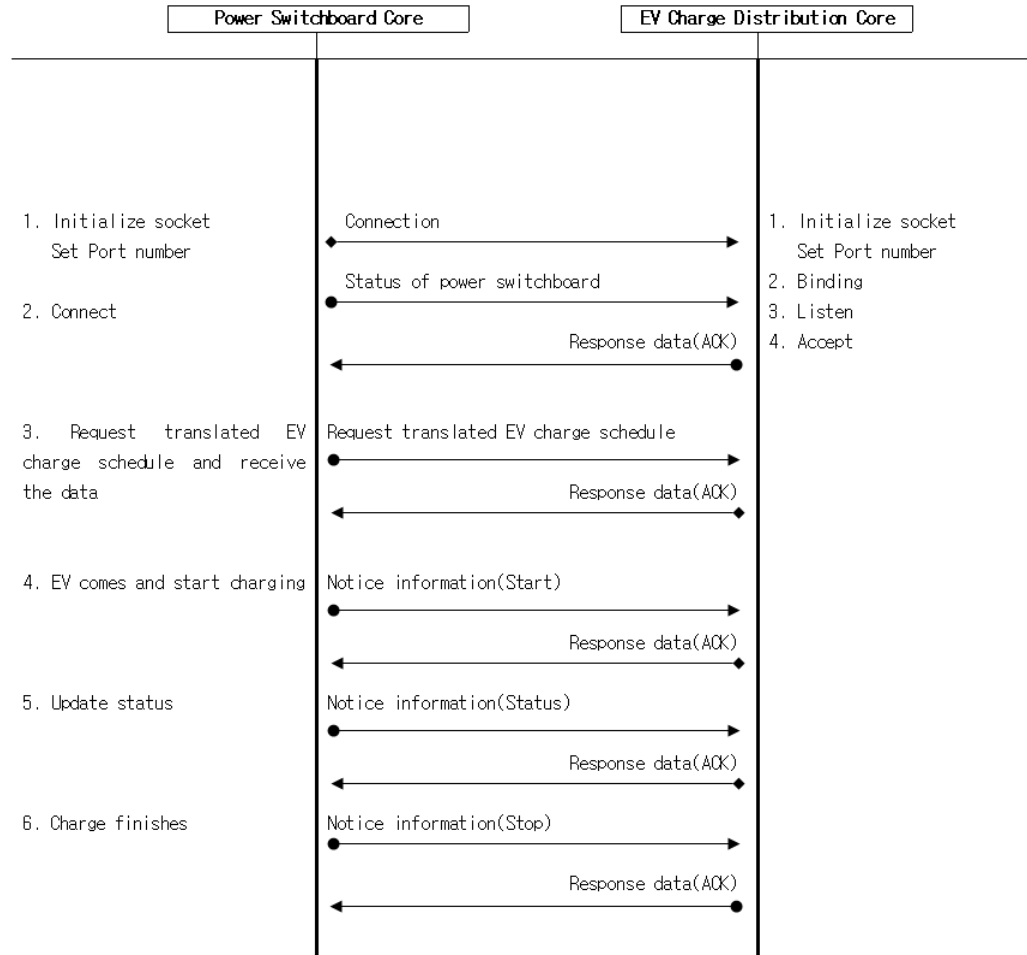
Fig. 10. Communication sequence between EV charge power distribution core and power switchboard core

4. Implementation Results

4.1 Smart EV Charge Schedule Engine System

Smart EV charge schedule engine system is designed for embedded Linux operation system. The engine system is implemented on 64-bit 16.04 Ubuntu by using Flask and Python. In addition, graphic user interface(GUI) of simulator is implemented on hypertext markup language(HTML). **Fig. 11** shows the implementation results of smart EV charge schedule engine, normal EV charge scheduler, and bus garage EV charge scheduler. **Fig. 12** shows the implementation result of simulator.

Fig. 11. Implementation results of smart EV charge schedule engine, normal EV charge scheduler, and bus garage EV charge scheduler

Fig. 12. Implementation results of simulator

4.2 EV Charge Power Distribution Core

EV charge power distribution core is designed for Windows operation system in embedded system. The core is implemented on 64-bit Windows embedded 7 standard by using C and C++. Power switchboard core is implemented on the same with EV charge power distribution core, however, it uses C#. The communication between EV charge power distribution core and power switchboard core is implemented by using local TCP/IP network with port number 6388. **Fig. 13** shows the implementation result of EV charge power distribution core that communicates with power switchboard core.

Fig. 13. Implementation result of EV charge power distribution core

5. Conclusion

In this paper, distributed charge signal processing software for smart slow and quick EV charge is proposed and designed based on ISO/IEC 15118. Distributed charge signal processing software is designed into two parts, one is smart EV charge schedule engine and the other is EV charge power distribution core. Two parts are based on ISO/IEC 15118. Smart EV charge schedule engine collects EV data and analyzes the collected data for calculating smart EV charge schedule. EV charge power distribution core translates the EV charge schedule from smart EV charge schedule engine, and commands to power switchboard core how much charge power and time on each EV charge port. The designed software is implemented on each embedded system and the results are shown in section 4.

Unfortunately, in republic of Korea, there is no EV supporting ISO/IEC 15118 yet. Therefore, the software is designed for dealing with EVs that can communicate or not. In addition, in the near future, the implemented software will be verified on the real EV charge stations such as normal EV charge station and bus garage EV charge station. It is expected that the designed software provides optimized EV charge schedule on each EV charge station so that EV charge station can manage efficient power distribution.

References

[1] M. Etezadi-Amoli, K. Choma, and J. Stefani, "Rapid-Charge Electric-Vehicle Stations," *IEEE Trans. on Power Delivery*, Vol. 25, No. 3, pp. 1883-1887, 2010. Article (CrossRef Link).

[2] Y. W. Son, J. H. Cho, and Y. E. Kim, "Distribution Switchboard for Slow Charger of EV able to Distribute as Power Capacity," *Trans. of the Korea Society of Automotive Engineers*, Vol. 26, No. 2, pp. 187-195, 2018. Article (CrossRef Link).

[3] Green, Erin H., Steven J. Skerlos, and James J. Winebrake, "Increase electric vehicle policy efficiency and effectiveness by reducing mainstream market bias," *Energy Policy*, Vol. 65, pp. 262-566, 2014. Article (CrossRef Link).

[4] The Edison Electric Institute and the Institute for Electric Innovation, "Plug-in Electric Vehicle Sales Forecast Through 2025 and the Charging Infrastructure Required," 2017.
Article (CrossRef Link).

[5] SNE Research, "Global EV Charger Industsry, Standardization Trend and Market Forecast(2014~2023)," 2016. Article (CrossRef Link).

[6] International Organization for Standardization, "ISO 15118." Article (CrossRef Link).

[7] Jianfang Xin, Qi Zhu, Guangjun Liang, Tiaojiao Zhang, and Su Zhao, "Performance Analysis of Cellular Networks with D2D communication Based on Queuing Theory Model," *KSII Trans. on Internet and Information Systems*, Vol. 12, No. 6, pp. 2450-2469, 2018. Article (CrossRef Link).

[8] Trippe, Annette, Tobias Massier, and Thomas Hamacher, "Optimized charging of electric vehicles with regard to battery constraints-Case study: Singaporean car park," In *Energytech, 2013 IEEE*, pp. 1-6, 2013. Article (CrossRef Link).

[9] Franco, John F., Marcos J. Rider, and Ruben Romero, "A mixed-integer linear programming model for the electric vehicle charging coordination problem in unbalanced electrical distribution systems," *IEEE Trans. on Smart Grid*, Vol. 6, No. 5, pp. 2200-2210, 2015.
Article (CrossRef Link).

[10] Richardson, Peter, Damian Flynn, and Andrew Keane, "Optimal charging of electric vehicles in low-voltage distribution systems," *IEEE Trans. on Power Systems*, Vol. 27, No.1, pp. 268-279 , 2012. Article (CrossRef Link).

Tae Uk Chang received the B.S. degree in Science from Hanyang University, Seoul, Korea, in 1992. He has been a CEO in Quantum Solution, Inc. Seoul, Korea, since 2015. He is working toward to M.S. and Ph.D. degrees in Multimedia from Seoul Women's University, Seoul, Korea. His research interests are energy convergence networks and time sensitive networks.

Young Su Ryu received the B.S. degree in Information & Communication Engineering from Sejong University, Seoul, Korea, in 2015. He has been an assistant researcher in Korea Electronics Technology Institute (KETI), Korea, since 2014. His research interests are digital broadcasting, energy storage system, and cooperative communication.

Seul Ki Song received the B.S. degree in Multimedia from Seoul Women's University, Seoul, Korea, in 2015 and M.S. degree in Information Media from Seoul Women's University in 2017. She has been an assistant researcher in Korea Electronics Technology Institute (KETI), Seoul, Korea, since 2017. Her research interests are digital broadcasting and digital communications system.

Ki Won Kwon received B.S. and M.S. degrees in computer engineering from Kwangwoon University, Korea, in 1997 and 1999, He also received the Ph.D. degree in the School of Electrical & Electronics Engineering from Chung-Ang University, Korea, in 2011. In 1999, he joined in KETI, Korea, where he is presently a principal researcher at the Smart Network Research Center. His research interests are in the area of and advanced broadcasting/communication system and energy convergence networks.

Jong Ho Paik received the B.S., M.S., and Ph.D. degrees in the school of Electrical and Electronic Engineering from Chung-Ang University, Seoul, Korea, in 1994, 1997, and 2007, respectively. He was a Director with Advanced Mobile Research Center at Korea Electronics Technology Institute (KETI) by 2011. Since 2011, he is currently an associate professor in the department of Software Convergence, Seoul Women's University, Seoul. His research interests are in the areas of web-based communication, software testing, wireless/wired communications system design, video communications system design and system architecture for realizing advanced digital communications system and for advanced mobile broadcasting networks as well.