SOLAR POWERED ELECTRIC VEHICLE INFORMATION SYSTEM

[1]JAKUB FORYSIAK, [2]KRZYSZTOF FUDALA, [3]PIOTR KRAWIRANDA, [4]JAKUB FELCENLOBEN, [4]ANDRZEJ ROMANOWSKI, [5]PRZEMYSŁAW KUCHARSKI

[1,2,3,4,5]Faculty of Electrical, Electronic, Computer and Control Engineering,Lodz University of Technology, Łódź, Poland
E-mail: [1]forysiakjakub@gmail.com, [2]krzyfud@gmail.com [3]piotrkrawiranda@gmail.com, [4]jakubf2007@gmail.com, [5]androm@kis.p.lodz.pl, [6]p.kucharski@samorzad.p.lodz.pl

Abstract - This paper provides detailed explanation of the architecture of a solar powered electric vehicle information system, including description of both hardware and software components. The primary target of this paper are the young constructors, intending to create their own vehicle; however, authors with all levels of experience may find useful ideas in the paper.

Keywords - Solar Powered Vehicle, Information System, Electric Car, iLumen European Solar Challenge

I. INTRODUCTION

The problem of creating a car that would use clean energy was analyzed by a group of students from Lodz University of Technology in 2014. They came up with the idea of creating first Polish solar powered electric car - Eagle One.
The car utilizes the energy generated from solar cells situated on the roof. Using that energy and built-in battery, the team drove more than 3000 kilometers during Bridgestone World Solar Challenge 2015 and 2817 kilometers during Sasol Solar Challenge 2016 in Republic of South Africa.

Having this experience, Lodz Solar Team decided to create the second model of their car,Eagle Two. Some of the modules of Eagle One were not working properly, so team decided to reinvent all of the parts - from mechanical components like suspension or the shape of the car's body, through electrical elements as solar panels and battery to completely new IT system.
The process of designing and impementation of one of the most significant components of the car - information system - will be the topic of this paper.

II. SYSTEM DESIGN

2.1 System overview

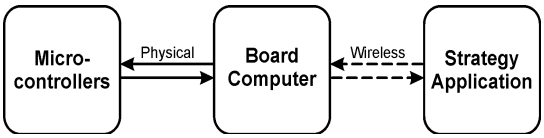

Fig. 1 System overview

The system is responsible for controlling the car, infotainment system and data acquisition for strategy purpose. Therefore, the team decided to divide it into three main parts as shown on Fig. 1. These layers are designed to perform specific tasks and in consequence they make use of different communication methods. The following chapter will describe the functionalities of all these modules.

2.2. Microcontrollers

The low-level part of the system is based on STM32 microcontroller family. Their main task is to distribute control signals and handle external components like BLDC motors or lights. To achieve that team decided to divide low-level system into three separate boards, as it would ensure better maintainability through modularity. It was also favorable solution because each part of the system can be placed closer to their areas of responsibility.
Motor Controller is in charge of steering motors and loading batteries from solar cells. It is based on a 32-bit microcontroller built into Nucleo L476RG board. The design uses the CAN bus utilities to communicate with the other modules within the Eagle Two. The main task of Motor Controller is steering, two BLDC motors inside the car using analog signal. Furthermore, it turns on the photovoltaic cells and sends to other modules their working parameters.
Driver Input Controller is mostly responsible for displaying the speed and current energy consumption of the car. It is vital that all data would be understandable and well visible but the device must not use huge amount energy. Therefore, in the designing phase it was decided to choose appropriate servomechanism that consumes energy only during movement. The other significant task performed by this board is gathering all inputs from switches and sensors - after getting them with proper frequency of sampling, Nucleo L476RG sends them to other components via CAN bus. What is more, this module is also responsible for on-board cruise control. Based on velocity feedback from motors it maintains constant speed with use of proportional-integral controller. Next module is responsible for lightning system used in Eagle Two. It is based on the same microprocessor as other boards. It communicates with other modules via CAN bus, but it does not send any data to others. Lights Controller has two main tasks:

- It turns on/off lights of the car. To achieve this, team used power MOSFETs with polymer fuses to prevent LED reflectors from burning.
- It is also responsible for displaying control lights situated in front of the driver.

2.3. Board computer

Board computer is based on one of the most popular microcomputer board in the world - Raspberry Pi 3B+. It is responsible for tasks listed in Tab. 1.

Tab.1 Tasks performed by board computer

	Task
1	Acquiring data from microcontrollers
2	Parsing the data
3	Creating WiFi network for strategist's PC to connect
4	Sending data through wireless communication to strategists
5	Creating back-ups of car data, when WiFi connection is lost
6	Data visualisation
7	Allowing changing car parameters using touch screen

To perform this tasks the team needed an appropriate framework that would operate properly on Raspberry Pi microcomputer, would offer libraries for communication both with microcontrollers and strategists and allow creating intuitive Graphical User Interface. After research team decided to use Qt C++ framework that is used in automotive industry by such producers like KDAB and Luxoft. It offers built-in modules for serial communication (used with microcontrollers) and wireless communication (used with strategist application). All of it is complemented with great GUI capabilities, which are used for data visualisation on 10 inch touch screen - part of the dashboard of the vehicle.

2.4. Strategy application

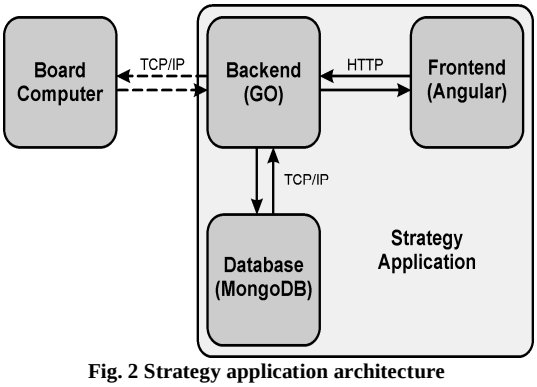

Fig. 2 Strategy application architecture

A crucial matter during World Solar Challenge is a strategy. In order to optimize energy consumption, every team has to define optimal speed in each race section. Therefore, the team decided to create a separate application, which both acquires car data and analyses it. The team divided the app into three modules as it is illustrated in Figure 2.

First one, called backend, is responsible for gathering data from the car, converting and saving it to database. It consists of a TCP server, which connects to a TCP client in board computer. Furthermore, to communicate with the Frontend module the team created HTTP server, which provide REST API. The API consists mostly of GET methods, the most important is endpoint, which provides all recent car parameters. For data storage, the team decided to use MongoDB, which is a documented database. Document is a data structure composed of field and value pairs. They are similar to JSON objects. The values of fields may include other documents, arrays, and arrays of documents. MongoDB provides flexible data models and real time aggregation, which are critical features for application development and analysis of huge sets of information. To enhance user experience and maximize portability the team decided to create graphical interface similar to modern websites. The team chose Angular framework to speed up development. User interface consists of live data view, diagnostic site with errors together with warnings and strategy tab, which provides vital information in case of race. Finally, the designed architecture let us deploy strategy app on the cloud, which was used during iLumen European Solar Challenge in Belgium..

III. COMMUNICATION

3.1. Inter-vehicular communication

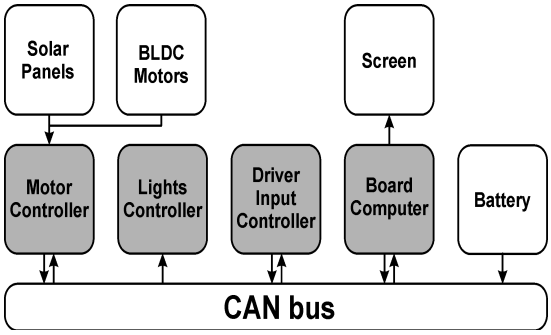

Fig. 3 Low-level car architecture

As shown in Fig. 3, designed system supports Controller Area Network (CAN bus). CAN is a multiplex, message-based standard that allows microcontrollers to communicate without any host computer. Systems marked in grey in Fig. 3 were designed to support core vehicle functionalities. CAN bus was set to operate in 250 kb/s. Such data rate was proven as sufficient for vehicle operation. Through that network all car information is shared, from throttle position to current battery temperature. Such solution allowed for complete telemetry of the whole car.

3.2. Wireless communication

Introducing wireless communication was inevitable, since according to the regulations of Bridgestone

World Solar Challenge, the escort car is behind the solar vehicle by approximately 60 meters. It means that to deliver the data to strategists, the system needs to support wireless communication. To achieve that, team decided to set up WIFI network in solar vehicle. Using this configuration ensured simplicity of reconnection between solar vehicle and strategists team. Thanks to capabilities of the board computer, WIFI network was easily manageable and error-proof. The range of communication reached approximately 250 meters thanks to application of two directional antennas.

As mentioned in chapter 2.4, this solution uses TCP/IP connection. The client side is inside the car, whereas the server one in the strategists application. This topology ensures the potential scalability of the network, team can connect more cars to one system.

The communication protocol is very straightforward. Every 100 milliseconds the board computer sends the whole packet of data from the vehicle. The data sent to strategists application includes:

- output voltages and currents of the MPPT.
- input from driver (throttle position, lights mode, drive mode and direction, handbrake status)
- battery data (state of charge, temperature of battery, max cell voltage, min cell voltage, charger information)
- BLDC motor data

One packet of data has 170 bytes. Knowing that the length of the frame is constant, there is no specific opening and closing of the packet - the fact that data will arrive both in correct order and unchanged is assured by TCP protocol. Thanks to that, team uses the same order of variables in the packet for sending and receiving data.

IV. TESTS

4.1. Integration tests

It is crucial to find all conflicts and problems during integration of all modules in the earliest possible stage of project. That would ensure the proper operation of the car during normal usage on the road. Because of the fact the communication with microcontrollers is complicated (needs external hardware - connectors and external power supply), the team decided to mock-up CAN Bus on the board computer. It allowed to prove the proper working of all parts of two modules of the system - Board Computer and Strategy application. The tested features included parsing of the data, graphical user interface, TCP communication between Raspberry Pi and strategists PC. Team simulated receiving 2 types of frames every 500 ms. After successful tests, the data volume was increased and whole Eagle Two dataflow was simulated. When team knew that everything is in order, the mockup module was removed and microcontrollers were introduced. Also these tests were fine and system was ready to run.

4.2. World Solar Challenge 2017

Bridgestone World Solar Challenge is not only the race, but also 13 days of preparation and testing on Hidden Valley Race Track, located in the suburbs of Darwin. Since there were not enough time in Łódź, team was forced to start tests in Australia. For the first two weeks team was allowed to test the car on Hidden Valley Raceway near Darwin. The track is about 2.9 km long, features 14 turns and 1.1 km long main straight. This mixture allowed the team to test different aspects of the design. During those initial weeks the team tested all crucial functionalities, including: input signal response, internal car communication, BLDC motor characteristics, solar panels operation and live telemetry. Thanks to that the team was able to define all problems and apply necessary fixes before the race. Main issues found were mainly caused by the extremely high temperatures inside the car or high insulation. Thanks to additional integration tests done before, IT system of the car was ready for the race Unfortunately due to extremely bad weather conditions, 2017's edition was very hard for teams - over half of them were transferred to Adventure Class, since they were not able to keep the appropriate tempo to finish the race on time. Unfortunately, Lodz Solar Team was in this group. First day of race was very successful. LST was on the second spot with a small loss to the first place.

Sadly, next days were much more difficult. Due to heavy rainfalls that happened for the first time in the history of the race in the middle of Australian desert, team was forced to put vehicle on the trailer to avoid destruction of electronic system. LST was transferred to Adventure Class, where there is no competition and all of the teams are not given any marks or places. Nonetheless, the team decided to try to beat its own record from previous edition of BWSC, 2113 person-kilometer distance (number of kilometers covered multiplied by people inside the car during journey) covered by Eagle One. Fortunately, the goal was completed - LST covered 4496 person-kilometers distance.

4.3. Ilumen ESC 2018

iLumen European Solar Challenge is the only 24 hours endurance race in the world for solar cars, which takes place every 2 years at the former F1 track of Circuit Zolder in Belgium. In 2018 LST decided to participate in this event. Information system of Eagle 2 needed some adjustments due to regulations of the event. As there was no escort car riding behind E2, the strategy application was deployed to the cloud solution. Thanks to switching to this mode, strategists team was able to see all telemetric data live in the pitlane. This information gave LST possibility to predict the consumption of the car and plan future operations. The system performed flawlessly during the entire event, not a single failure occured. Thanks in part to the reliability of the system, the team took the first place in Cruiser Class.

CONCLUSIONS

The solar powered electric vehicle information system was studied and major conclusions are as follows:

1. Live data tracking is crucial for efficient strategy during solar car races.
2. Strategy application deployment on cloud solution ensures ensure availability from all over the world for more users at the same time.
3. All electronics components should be of automotive standard, to ensure proper operation in extreme environment.
4. To make solar powered electric vehicles more similar to modern cars, comfort of the passengers should be emphasised. Constructors should take into consideration modules like satellite navigation, radio and other infotainment solutions and treat them as a necessity in designing their cars.
5. All internal car connectors should be fail safe, as majority of errors were connector-related.
6. Modularity of low-level boards ensures high level of maintainability.
7. The sending protocol consisting of data packets with the same length is very efficient, but it should be slightly changed. Some of the data are needed more frequently than the other, so it should be adjusted in the next version of the system.

REFERENCES

[1] W. Franz, H. Hartenstein, B. Bochow, "Internet on the road via inter-vehicle communications", Proc. GI/OCG Annual Conference: Workshop on Mobile Communications over Wireless LAN: Research and Applications, September 2001.

[2] A. Bohm, "State-of-the-art on network layer aspects for inter-vehicle communication", Technical Report IDE0748, June 2007.

[3] P. Krishnamurthy, "Information dissemination and information assurance in vehicular networks", Proc. iConference '08 Los Angeles, February 2008.

[4] B. Hughes, R. Meier, R. Cunningham, V. Cahill, "Towards realtime middleware for vehicular ad hoc networks", Proceedings of the 1st ACM international workshop on Vehicular ad hoc networks, pp. 95-96, 2004.

[5] C. Manasseh, R. Sengupta, "Middleware for cooperative vehicle-infrastructure systems", California PATH Research Report UCB-ITS-PRR-2008–2, January 2008.

[6] F. Salewski, S. Kowalewski, "Hardware/software design considerations for automotive embedded systems", IEEE Transactions on Industrial Informatics, vol. 4, no. 3, pp. 156-163, Aug. 2008.

[7] A. Perez, A., M. I. Garcia, M. Nieto, J. L. Pedraza, S. Rodriguez, J. Zamorano, "Argos: An Advanced In-Vehicle Data Recorder on a Massively Sensorized Vehicle for Car Driver Behavior Experimentation", IEEE Transactions on Intelligent Transportation Systems, pp. 463-473, 2010.

[8] Z. Huixin, yulong Hou, "The realization of multi-channel storing system", Computer Application and System Modeling (ICCASM) 2010 International Conference on, vol. 7, pp. V7–20-V7–22, 22–24 Oct. 2010.

[9] M. Pecha, "System for Acquisition of Data from Car Communication System", 2009.

[10] K. Etschberger, "Controller Area Network", IXXAT

[11] Automation GmbH, pp. 1-430, 2001.

★★★