energies

Article

A ROS-Based Energy Management System for a Prototype Fuel Cell Hybrid Vehicle

Savvas Piperidis [1,*], Iason Chrysomallis [2], Stavros Georgakopoulos [3], Nikolaos Ghionis [2], Lefteris Doitsidis [1] and Nikos Tsourveloudis [1]

[1] Intelligent Systems and Robotics Laboratory, School of Production Engineering and Management, Technical University of Crete, 73100 Chania, Greece; ldoitsidis@dpem.tuc.gr (L.D.); nikost@dpem.tuc.gr (N.T.)
[2] School of Electrical and Computer Engineering, Technical University of Crete, 73100 Chania, Greece; ichrysomallis@isc.tuc.gr (I.C.); ghionis3@otenet.gr (N.G.)
[3] School of Production Engineering and Management, Technical University of Crete, 73100 Chania, Greece; sgeorgakopoulos@isc.tuc.gr
* Correspondence: spiperidis@isc.tuc.gr; Tel.: +30-28-2103-7292

Citation: Piperidis, S.; Chrysomallis, I.; Georgakopoulos, S.; Ghionis N.; Doitsidis L.; Tsourveloudis, N. A ROS Based Energy Management System for a Prototype Fuel Cell Hybrid Vehicle. *Energies* **2021**, *14*, 1964. https://doi.org/10.3390/en14071964

Academic Editors: Juan Pérez Torreglosa and Pablo García Triviño

Received: 18 February 2021
Accepted: 26 March 2021
Published: 1 April 2021

Abstract: The automotive industry has been rapidly transforming and moving further from internal combustion engines, towards hybrid or electric vehicles. A key component for the successful adoption of the aforementioned approach is their Energy Management Systems (EMSs). In the proposed work, we describe in detail a custom EMS, with unique characteristics, which was developed and installed in a hydrogen-powered prototype vehicle. The development of the EMS was based on off-the-shelf components and the adoption of a Robot Operating System (ROS), a meta-operating system developed for robotic-oriented applications. Our approach offers soft real-time control and the ability to organize the controller of the EMS as a straightforward and comprehensive message system that provides the necessary inter-process communication at the core of the EMS control procedure. We describe in detail the software-based implementation and validate our approach through experimental results obtained while the prototype was racing in a low-energy consumption competition.

Keywords: energy management; energy harvesting; finite state machine; hybrid power systems; hybrid electric vehicles; ROS; supercapacitors

1. Introduction

The world of automotive engineering has been facing a challenging change that goes far beyond the typical expected evolution caused by technological progress. Global environmental, social, financial and political issues are forcing the powertrain technology of the automobiles' industry to change its basis: from the Internal Combustion Engine (ICE) to electric motors. This ongoing change, after more than a century of ICE domination in the automotive industry, is provoking an avalanche of modifications and new designs like novel electric Energy Management Systems (EMSs) and regenerative systems capable of electric energy monitoring, distribution, harvesting and storage. Methods, devices like Fuel Cells (FC), batteries, charging devices, DC/DC converters, super-capacitors, hardware and software related to the control of electric motors-generators have become popular research and development objectives, not only at the automotive industry, but also within the scientific community.

The idea of Hybrid power source Electric Vehicles (HEVs) seems to have a great potential in the future of electromobility. The combination of the different characteristics of the alternative power sources utilized at the HEV's powertrain improves the performance and the efficiency of the whole system [1,2]. Several related works are found in the literature. In [3], a hybrid power system consisting of a 90-kW Proton Exchange Membrane Fuel Cell (PEMFC) and a 600-F Super-Capacitors (SC) bank was developed and installed on an HEV. The efficiency of the system was measured at 96% and the performance and range were adequate not only for an experimental HEV, but also for a typical commercially available passenger car. The control strategy of the electrical power system in this project is based on a set of rules and the control procedure is a sequential algorithm, described by a flow chart. In another related work, the authors propose an EMS installed on a hybrid electric bus [4]. The system aims at maintaining the state of energy of the SCs' bank between two limits, which are computed online. The EMS controller follows a set of rules, derived from the state of energy of the SC bank and three other parameters: the maximum speed, the acceleration during propulsion trips and the deceleration during braking trips. The FC and the SC bank are modeled in a simulation environment where real road trip data are used to measure the proposed system's performance. A different EMS strategy described in [5], takes into account the FC performance, the DC/DC converters' characteristics and the SCs' charging and discharging characteristics to improve the energy consumption, to increase the system's efficiency and maintain the state of charge of the SCs above a predetermined value. The system is tested and verified on a reduced scale laboratory bench, using an electronic load to simulate the power demand of an EV. The control algorithm is sequential and it is described by a flow chart, representing a set of rules. Marzougui et al. [6] proposed a two level control strategy for the EMS of an HEV. The equivalent consumption management strategy is at the high level of control whereas at the low level, PI controllers regulate the FC and SC currents, according to the system's specifications and limitations. This work includes experimental results in simulations. In [7,8], novel knowledge-based, multiphysics-constrained and machine learning-based energy management strategies are proposed for hybrid electric buses.

A study of two energy management strategies for HEVs is presented in [9]. The first is based on a sequential algorithm that follows a set of rules. The second strategy utilizes a sequential algorithm based on a set of fuzzy logic controller rules. Both algorithms are described by flow chart diagrams. The strategies are tested, compared and verified in a simulation environment. In [10], the authors focus on the comparison of four HEV powertrain configurations along with the analogous EMSs: peaking power source strategy, operating mode control strategy, fuzzy logic control strategy and equivalent consumption minimization strategy. The control algorithms are described by flow chart diagrams, used to represent the set of rules for each strategy. Lawrence et al. [11] presented an analysis of the systems in a prototype HEV trying to determine if it is possible to increase the vehicle's efficiency through strategic control of the power provided for auxiliary systems, like the air conditioning or the power steering system. The strategy is based on optimization techniques and it is validated in a simulation environment. The control algorithm is described by a flow chart representing a set of rules. Wu and Gao in [12] study the minimization of the cost, the volume, the weight and the number of the cells in an FC and the units of an SC bank, combined together to form the powertrain of the HEV. The proposed system is tested using a simulation environment.

Nowadays, the hybrid topology is applied not only to vehicles, tramways and buses but also to ships. Although most of the EMSs in the literature emphasize on other factors than the dynamic characteristics of the respective power generation units of the hybrid system, Tang et al. propose the wavelet method, as a tool to decouple the power demand of a ship into high and low frequency [13]. After the decoupling, the transient power is designed to be supplied by the SC bank, while the base power is supplied by the FC according to a set of rules. The proposed EMS is tested and verified in a simulation environment.

The popularity of hybrid techniques is presented in detail in the following surveys [14–16], which include more than 200 complete works in the scientific field of HEVs, equipped with a combination of hydrogen FC and an energy storage bank of SCs or some kind of rechargeable battery. Many different EMS strategies are presented and tested, including, but not limited to, rule or optimization based strategies, fuzzy logic based control, adaptive neuro-fuzzy inference systems, filtration-frequency decomposition based strategies, optimal control strategies, equivalent consumption minimization strategies, dynamic programming, genetic algorithms, neural network, particle swarm optimization techniques and polynomial control techniques. The authors in [14–16] compose an outline of the state of the art EMSs, operating in prototype HEVs, test bench models, commercially available vehicles, pilot projects' platforms or simulated environments. They present, study, verify and test the variations of architectures and topologies of the electrical hybrid power systems, highlighting their main advantages and disadvantages. They, also, offer access to experimental data, collected from the operation of real vehicles or simulated models.

Regarding the methodology of the EMS control procedure, most of the controller programs found in the literature are sequential algorithms, implementing a set of rules. In certain cases, a flow chart is used for representing the set of rules, which is the base of the control procedure [3,5,9–11]. The computational model of the Finite State Machine (FSM), may be found in [17–19]. In [17] the authors conduct a comparative analysis on FC based emergency power system of a more-electric aircraft, using simulation and experimental test bench may be found. The analysis refers to the following EMS techniques: the FSM control strategy, the rule-based fuzzy logic strategy, the classical proportional-integral control strategy, the frequency decoupling/fuzzy logic control strategy, and the equivalent consumption minimization strategy. A state machine strategy based on droop control is described in [18] for a hybrid FC-battery-SC tramway. The proposed EMS strategy is evaluated using real driving data and a hybrid system model of a tramway, made by commercially available devices. In a more recent work [19] Li et al. introduce a state machine control technique based on the equivalent consumption method for an FC-SC hybrid tramcar. Four states, representing different values of the state of charge of the SC bank, are used at this state machine. The proposed control strategy was tested and evaluated by the RT-LAB semi-physical simulation platform.

In the proposed work we describe in detail a custom EMS, with unique characteristics, which was developed and installed in Spyros Louis (*Spyridon Louis*, commonly known as *Spyros Louis* was the Greek Olympic champion of the marathon race at the first Olympic games of modern history, held in Athens in 1896), a racing prototype single-seater car developed and raced by the Technical University of Crete Eco Racing (TUCer) team [20], in low consumption marathons, Figure 1. The prototype EMS is based on open source software and low cost off-the-shelf components and devices. Its controller strategy uses an FSM, where the transitions between the states are governed by a set of typical IF-THEN rules. The software framework of the EMS was designed and developed in the Robot Operating System (ROS) environment, which is a meta-operating system developed for robotic oriented applications [21]. It offers soft real-time control and the ability to organize the controller of the EMS as a straightforward and comprehensive message system that provides the necessary inter-process communication at the core of the EMS control procedure. The program modules were designed and developed, aiming at low complexity, expandability and modularity in an easy to maintain software framework. It is composed of a set of ROS nodes-procedures that are sending, receiving and logging user defined messages, that monitor the sensors' readings and the controller's commands. The features of the ROS development environment were exploited for the creation of a soft real-time controller, without the overhead of complicated real-time programming languages and techniques. The aforementioned approach was extensively tested in the laboratory and successfully validated with the participation of the vehicle in the Shell Eco-marathon (SEM) Europe 2019, proven to be safe, efficient and robust.

Figure 1. The prototype *Spyros Louis* at the Shell Eco Marathon Europe, London, 2019.

Based on an extensive literature survey, there is no comprehensive analysis of the software framework or the software programming techniques used to control an EMS. Moreover, to the best of our knowledge, this work is the first implementation of a ROS-based EMS controller. The ROS development environment benefits greatly the successful implementation of the EMS through the adoption of already proven and robust tools for developing a real-time control system, from the robotics domain. We believe that our approach can benefit the research community by proving a methodology for rapid prototyping of not only EMS systems for electric vehicles, but also for expanding the capabilities of their integration and interaction, with other components of an autonomous operation framework which can be supported by ROS, as it has already been demonstrated in a variety of cases [22]. It is worth noticing that the scope of the paper is not to present an optimized EMS, rather to provide a framework which is *completely independent* from the proposed EMS strategy. ROS allows us to easily accommodate, any given approach for developing an EMS and integrate it, in a functional prototype, as it was highlighted by the controller strategy which uses a rule based FSM.

The rest of the paper is organized as follows: in Section 2, there is a description of the vehicle's electrical system structure and its basic components. Also, the two modes of the vehicle's operation, normal and regenerative, are analyzed in terms of energy flow and power switches' operation. Section 3 focuses on the ROS-based software implementation of the EMS and presents the FSM model of its controller. Initially we present in detail the ROS nodes and messages' topics that compose the software framework and describe its interconnections and data formatting. We also give a description of the alphabet, the states and the transitions of the FSM, used at the core of the EMS controller. The evaluation of the proposed approach, using a prototype HEV, is presented in Section 4. As the proposed system was a part of the vehicle's powertrain during SEM Europe 2019, this section focuses on the study of data collected from the races. The conclusions from the development of the prototype ROS-based EMS, along with thoughts for further work and future research directions are presented in Section 5.

2. Electrical System

The overall hardware architecture adopted in this work for the design of the electrical system is presented in Figure 2. The system is composed of a PEMFC, an SC bank, an isolated buck-boost DC/DC converter, an electric Motor/Generator (M/G), a Motor Driver

(MD), an Electronic Control Unit (ECU), three Solid State Relays (SSRs) and three high power diodes (the exact components are listed in Appendix A).

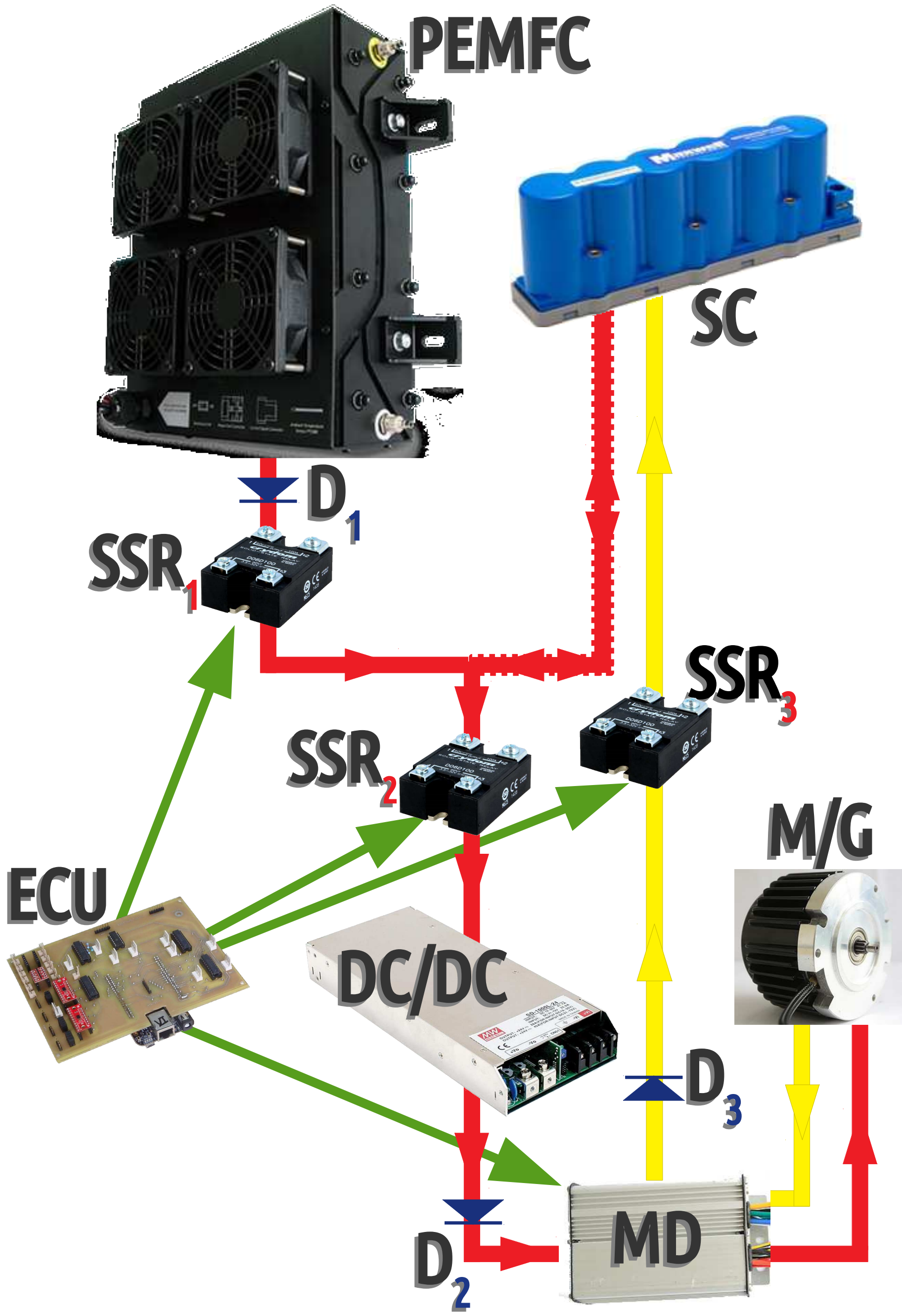

Figure 2. The architecture of the electrical system, where PEMFC is the Proton Exchange Membrane Fuel Cell, SC is the Super-Capacitors bank, DC/DC is the isolated buck-boost DC/DC converter, M/G is the electric Motor/Generator, MD is the Motor Driver, ECU is the Electronic Control Unit, SSRs are the Solid State Relays and Ds are the high power diodes. The red line indicates the energy flow during the *normal mode* operation. The dotted part of the red line indicates bidirectional flow of energy. The yellow line indicates the energy flow during the *regenerative mode* operation. The green line represents the communication between the ECU and the actuators of the EMS.

The electrical system of the HEV has two modes of operation. Each one of these utilizes the components of the electrical system in a different way. In *normal mode* operation, the vehicle accelerates or travels with constant speed. The PEMFC and the SC bank are combined together as power sources, to deliver the necessary energy to the M/G. The SSR_1 and SSR_2 are in *ON* state, meaning that they allow the electric current to flow through them, while the SSR_3 is in *OFF* state. When this mode is active, the M/G consumes energy and acts as an electric motor. The path of the energy flow, for this mode of operation, is presented in Figure 2 using red lines. A part of the red line route of energy flow is dotted. This is to indicate the bidirectional flow of energy that might occur during normal mode: the PEMFC may, during low consumption periods of the drive cycle, charge the SC bank and, in parallel, drive the electric motor, while in periods of high energy demands both the PEMFC and the SC bank are combined together as power sources to provide the necessary electric energy to the electric motor.

In *regenerative mode* operation, the driver stops pushing the acceleration pedal and the vehicle decelerates. The PEMFC is disconnected from the power rail. The SC bank, in turn, is connected to the power rail for charging. This switching task is accomplished by the use of the SSRs. During *regenerative mode* operation the SSR_1 and SSR_2 are in *OFF* state, while the SSR_3 is in *ON* state. Due to the momentum of the vehicle the driving wheel of the HEV keeps on rotating, forces the M/G to rotate and produce electric energy, acting as a generator during this mode. The harvested energy is routed and stored in the SC bank, as shown in Figure 2 using yellow lines.

A 48 V–1 kW PEMFC stack is used as the primary power source of the Spyros Louis HEV (Figure 3). It was the preferable option over the alternative available fuel cell types [23]. A key benefit of this choice, that makes PEMFCs operation relatively more sufficient for vehicles and transportation applications, is the combination of small dimensions with great dynamic characteristics and high energy efficiency [24].

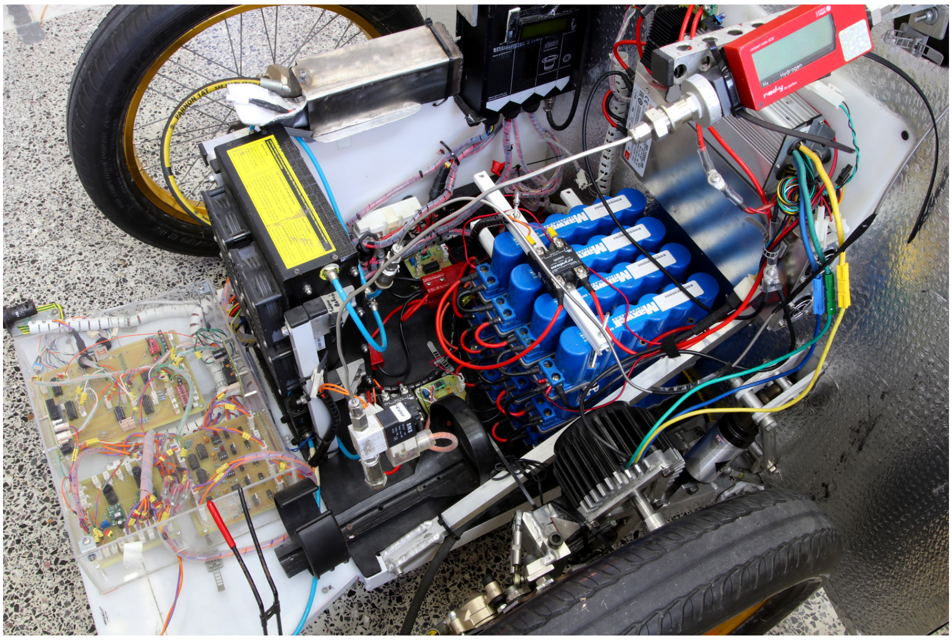

Figure 3. The powertrain compartment of the Spyros Louis HEV.

During the *normal mode* operation of the HEV, the PEMFC needs an auxiliary, high power density source to avoid the fuel starvation phenomenon due to the on-the-road dynamic power demand [25,26]. The auxiliary power system of the Spyros Louis HEV consists of an SC bank. During steep accelerations the vehicle's electric motor demands bursts of power. The SC component of the powertrain fulfills these high power density demands in the most effective way, without affecting or harming the reliability and efficiency of the PEMFC operation [27]. Furthermore, the SC bank is utilized as an energy storage component during *regenerative mode* operation. The vehicle's SC bank consists of 12 modules, configured in 3 parallel components, each composed of four 58 F/16 V modules, connected in series. The total capacitance of the SC bank is 43.5 F at 64 V.

The voltage output of power supply devices like the PEMFCs varies, depending on the current drawn by their load. The voltage of the PEMFC output drops significantly as the load of the MD increases. This behavior decreases the efficiency and the overall performance of the MD. The task that the DC/DC converter performs inside the electrical system, is to maintain a constant voltage value in the DC bus where the MD is connected, regardless the vehicle's power demands. The input of the DC/DC converter is connected in parallel with both the PEMFC and the SC bank as shown in Figure 2. The proposed approach is not algorithmic and the contribution of the two different power sources of the DC/DC converter is passive: the fuel cell contributes to the power demands of the DC/DC converter according to its polarization curve while the SCs provide the necessary additional power when the driver asks for high power bursts. Spyros Louis HEV utilizes a 1 kW buck-boost DC/DC converter, with an input voltage range of 19–72 V and an output voltage of 48 V.

During the *normal mode* operation of the HEV, the DC/DC converter operates as a boost converter. The traction at the Spyros Louis HEV is provided to its rear right wheel, by a three phase brushless 1 kW / 48 V M/G. In *regenerative mode* operation, the M/G operates as a generator and produces electric energy and voltage values up to 55 V, depending on the HEV's velocity and the SCs' state of charge. The M/G is driven by a 12 FET, 36–72 V/40 A MD.

The vehicle's ECU is a custom made plug in board, that extends the functionality and capabilities of an open source, ARM based, Linux-ROS powered development platform [28] (the detailed specifications are presented in Appendix B). Among others, the ECU executes the ROS-based controller program interchanging the FSM states of the EMS, monitors the sensors' measurements, transmits signals to control the SSRs' switching activity and manages the flow of energy inside the powertrain of the HEV.

The SSRs actuate the alternative energy paths in the powertrain of the HEV during the *normal* or the *regenerative mode* of vehicle's operation. They have high frequency switching capabilities in relatively high voltages values. Also, they offer increased lifetime compared to the typical electromechanical relays, as they have no moving parts to wear out. Three 40 A/200 V SSRs are used in the vehicle's electrical system. Three 60 A/200 V ultrafast diodes are utilized at the electrical system as a safety precaution. As presented in Figure 2, the diode D_1 prevents the reverse current from flowing to the PEMFC. D_2 allows current to flow in the MD during *normal mode* operation but blocks reverse current from flowing to the DC/DC in *regenerative mode* operation. D_3, blocks any current flow from the SC bank to the MD in *normal mode* operation while allows SC bank charging during *regenerative mode* operation.

3. Energy Management System Software Implementation

ROS, as already mentioned, provides soft real-time control and communication features by utilizing an *anonymous-asynchronous publish-subscribe* mechanism, between the program nodes that constitute the controller. Real time control and inter-process communication is necessary to synchronize the EMS automation procedure. In the typical ROS environment of the proposed EMS, multiple program nodes, running *simultaneously* in the ROS multi-thread environment, compose the automation procedure. The timing they provide for actuating the alternative paths of energy flow at the EMS has to be robust, reliable and fully programmable. When this is not the case, severe damages will occur at several devices of the electrical system, destroying the powertrain of the HEV. The challenging and time consuming real-time programming task using multi thread methods, or other techniques, is replaced at the proposed system by the ROS node message-publishing time schedule and the use of timestamps. The ROS timing procedure is transparent to the programmer, that has only to choose the frequency of exchanging data between the program nodes. Thus, the ROS-based source code is reduced in size, easy to comprehend and edit, easy to maintain and debug and significantly more intuitive than the one produced by a typical real-time, multi thread programming language. As it will be described in detail, in Section 4, ROS soft real-time programming technique proved to be adequate for the time scheduling of the necessary latency periods and the SSRs' switching activity, for the safe transition between the different modes of the EMS.

ROS programs are organized in program nodes, that are active simultaneously using multitasking techniques. Each one of these nodes may publish data messages in topics and in the same time, act as a subscriber to topics broadcast by some other node. The publishing procedure is time scheduled by the ROS core and fully programmable, Figure 4. Whenever a subscriber to a topic receives a new message, a control action is triggered. The EMS controller program was divided in discrete parts/nodes for monitoring sensors' data, logging useful information and routing the controller commands to the actuators of the electrical system of the HEV. Each node is independent from the others and may be in running mode, or not, without affecting the rest nodes. The EMS controller was programmed effortlessly at a high level of abstraction, with the help of the ROS messaging system and the timing tools, offered as a default feature of the ROS environment. The beta source code of the EMS program/nodes was edited, tested and debugged in less than a week and not only is easily maintained but it also may be enriched with new features and functionalities on the fly, by adding the respective nodes.

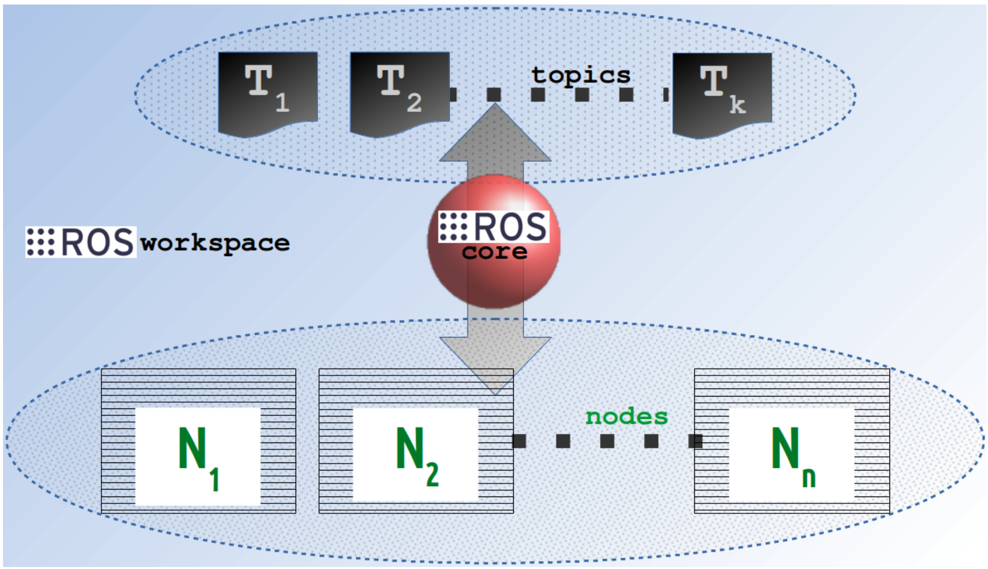

Figure 4. ROS programs are organized in program nodes. Each node may be a publisher or a subscriber to certain message topics. The messaging procedure is time scheduled and programmable.

Inter-process communication is, also, an essential ROS feature, totally transparent to the programmer. ROS ensures that the data flow implementation, in the EMS multi process controller program, is running smoothly. Using library functions from the default ROS messaging system, the necessary data for the control procedure are published in topics by publisher program nodes, with a user defined frequency, when and if they are available. Whenever a message is received by a subscriber program node, a certain control *action* is triggered. Thus, for example, data collected from a node monitoring the sensors' readings are published in the respective topic with a user defined frequency. Receiving such a message in a node, subscriber to the same topic, will result to certain control actions, as defined in the subscriber's node source code. Moreover, the ROS-based EMS controller program is a modular software framework that offers design flexibility, as nodes may be programmed ad hoc and may be added or removed anytime. During the software development or debugging phases, nodes may be activated or terminated, from inside the user space, even during program execution.

Although the ROS-based EMS offers features of high computational cost, like soft real-time control and messaging, still it has reasonable hardware requirements and can be integrated into low power, low cost, open source, terminal based Linux development boards with compact dimensions. These highlighted arguments are crucial for the development of the racing Spyros Louis prototype HEV, with emphasis to low consumption and dimensions. ROS is an open source software that offers a worldwide ecosystem of open source programs and features, organized in the form of software packages, in public repositories. Also, its software package management tools allows the collaboration with the community and facilitates the publication of the EMS controller program in the form of a ROS software package [29]. More than a year of testing and racing with Spyros Louis HEV, proved that the ROS-based EMS software was reliable, efficient and robust.

The following EMS nodes were programmed using the Python programming language:

data-acquisition node (dN): implements the data acquisition task by accessing and monitoring the sensors. It publishes the collected values to the *data-acquisition Topic*.

control node (cN): processes *data-acquisition Topic* message's data and implements the control procedure. The *control node* can easily accommodate the integration of any given EMS approach in the ROS framework. In the present work, a *hand-tuned* rule based FSM was chosen as a proof of concept, to highlight the applicability and functionality of the proposed software frame. The *control node* publishes some of the control commands as a broadcast to the *control Topic* and it is a subscriber to the *data-acquisition Topic*.

logger node (lN): implements the data logging task. It is a subscriber to both *data-acquisition Topic* and *control Topic*.

power-off node (pN): implements the system shutdown task. It sends a power-off system call as soon as the HEV's power switch is turned off.

Two message topics are currently used by the ROS broadcasting system for data publication and subscription among the EMS's program nodes, the *data-acquisition Topic* and the *control Topic*, Figure 5:

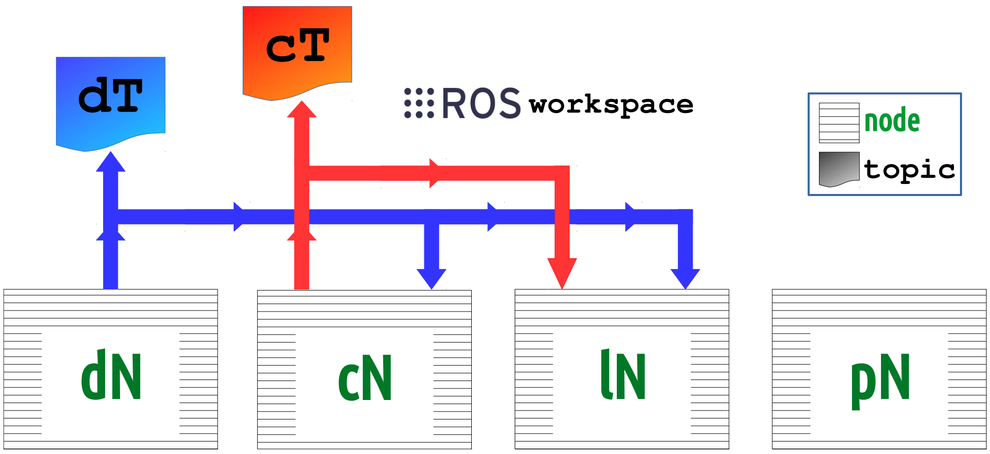

Figure 5. The EMS ROS workspace.

data-acquisition Topic (dT): receives messages from the *dN* and forwards them to both the *cN* and the *lN*. The sensors are sampled at a rate of 4 Hz. The messages' data contain the following measurements:

- the velocity of the HEV,
- the output voltage and current of the PEMFC,
- the voltage of the SC bank and
- the current consumed or generated by the HEV's M/G.

control Topic (cT): receives messages from the *cN* and forwards them to the *lN*, with a frequency of 4 Hz. These messages contain the values of two control parameters, the accelerator pedal switch signal and the active FSM state. The accelerator switch signal indicates whether the driver is pressing the accelerator pedal or not, meaning that the HEV is driven in *normal* or in *regenerative mode*, respectively.

The *cN* is based on an FSM, used as a computational model, consisting of a set of states and transitions. This is a dynamic approach that describes the evolution over time of a set of discrete and continuous state variables [30]. This technique was chosen because it produces an object-oriented depiction of the system, easily translated to ROS nodes using an object oriented programming language like Python [31]. Also, the definition of the finite set of states, the alphabet and state transition function is a strict but at the same time straightforward method helping error elimination and debugging. The FSM technique is intuitive and easily conceivable, even for someone with less or none knowledge about the EMS. It is less extensive and complicated than the other representation and programming techniques, like the algorithm or the flow chart. Moreover, it provides the ability of making changes to the EMS by adding or removing states or transitions or both of them. Finally, it is best suited to systems with sequential logic, like the EMS described at this paper.

The FSM, utilized as *cN* program node, can be described as the tuple $A = (Q, \Sigma, \delta, q_0, F)$ [30], where:

FC signal: denotes a confirmation signal received from the PEMFC. The signal declares an error-free, stand-by condition of the PEMFC.

accelerator pedal switch: indicates whether the driver is currently pressing down the accelerator pedal,

velocity: indicates the instant velocity value of the HEV and

velocity threshold: indicates a velocity value. Below this value there is no regenerative electric energy harvesting during the *regeneration mode* operation. Namely, the M/G will not produce any electric energy, acting as generator, if the HEV is moving at a velocity with lower value than the *velocity threshold*.

The EMS operation is composed by six states, presented at Table 1 and described below. The transition time between different states is 0.25 s and follows the publishing frequency of the sensors' data published at the data-acquisition Topic (dT).

Table 1. Description of the FSM states along with the respective state of the SSRs.

State	State Description	SSR_1	SSR_2	SSR_3
q_0	initialization	OFF	OFF	OFF
q_1	idle	ON	ON	OFF
q_2	normal mode operation	ON	ON	OFF
q_3	intermediate step to regeneration mode	OFF	OFF	OFF
q_4	regeneration mode operation	OFF	OFF	ON
q_5	intermediate step to normal mode	OFF	OFF	OFF

Initialization state: the initial FSM state, q_0, represents the system start up state. The ECU initializes its control outputs and awaits for the *FC signal*. The *FC signal* is a confirmation signal produced by the PEMFC to declare its error-free, stand-by condition. All the SSRs are at the OFF state.

Idle: after receiving the *FC signal* from the PEMFC, denoting its normal stand-by condition, the FSM is shifted to the idle state q_1. At this state, the HEV is motionless, or it is moving with a velocity less than the *velocity threshold*. The SSR_1 and the SSR_2 are ON, to permit the power flow to the DC/DC converter, while the SSR_3 is OFF since there is no regenerative energy flow.

Normal mode operation: the moment the driver presses the accelerator pedal and the *accelerator pedal switch* is turned on, the FSM transitions to q_2, that represents the *normal mode* operation of the HEV. Taking into consideration the throttle signal, different volumes of acceleration are being instructed to the MD. The state of the SSRs is the same as in the q_1 state.

Intermediate step from normal to regeneration mode: when the driver wants to decelerate the vehicle and releases the accelerator pedal, the *accelerator pedal switch* is turned off and the FSM progresses to q_3. This state is an intermediate fail-safe state, ensuring that enough time is given to the SSRs to be set properly, before entering the *regeneration mode* operation. All the SSRs are OFF.

Regeneration mode operation: if the *accelerator pedal switch* remains turned off, the FSM is shifted from q_3 to q_4 state, that corresponds to the *regeneration mode* operation. There are two alternatives for exiting from q_4. According to the first, the exit may happen if the velocity value decreases below the *velocity threshold*. In this case the FSM will move back to the q_1 idle state. According to the second alternative, the exit from q_4 may happen if the driver presses the accelerator pedal that turns the *accelerator pedal switch* on. At this case, the FSM transitions to q_5. The SSR_1 and the SSR_2 are OFF, while the SSR_3 is ON to utilize the alternative energy flow during the regenerative mode, indicated by the yellow line at Figure 2.

Intermediate step from regenerative to normal mode: q_5 is another intermediate fail-safe state, ensuring that enough time is given to the SSRs to be set properly before the FSM exits the *regenerative* and enters the *normal mode* operation. All the SSRs are OFF.

4. System's Evaluation

The proposed ROS-based EMS system was thoroughly tested on the prototype HEVs and the prototype chassis dynamometer [32] of the TUCer team. The prototype urban concept racing HEV Spyros Louis, equipped with the proposed EMS, took part at the SEM Europe 2019 (Figure 1). To successfully complete the race, the vehicle had to complete 11 laps in a closed track, Figure 6, in no more than 39 min. Each time that the vehicle was completing a lap, it had to perform a *stop-and-go* procedure of 5 s. The length of the track was 1420 m. The EMS operated faultlessly and proved its reliability and robustness. Moreover, the fine tuning process during the trial laps and between the races was straightforward, thanks to the modular design of the ROS-based controller and the deployment of the FSM. The Figures 7–12 are diagrams presenting the values of the most important EMS parameters. These values were collected using data logged during the three first laps of the first race of the SEM Europe 2019.

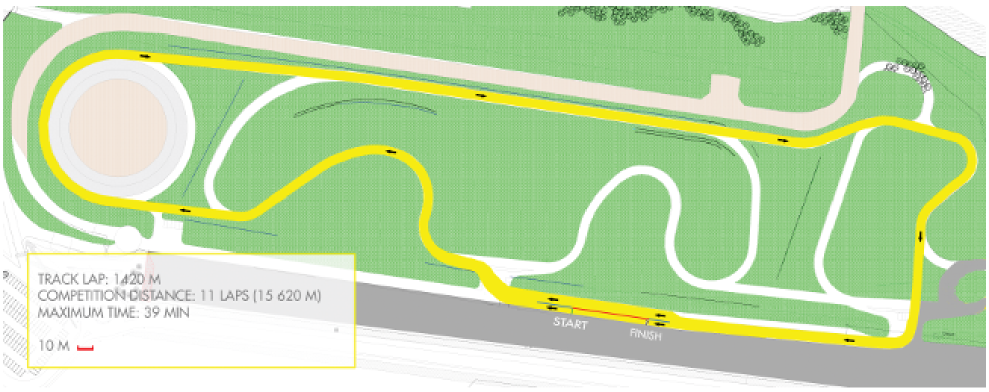

Figure 6. The track map of the Shell Eco-marathon Europe 2019.

Figure 7 depicts the velocity values of the HEV, along with the states' transitions during the 215 s period of the first race lap. The HEV accelerates until the 140th second. The FSM state is shifted from the q_1 state to q_2 at the first second, as the vehicle starts to operate at *normal mode* and remains unchanged until the 140th second. At the period from the 140th second to the end of lap there are successional transitions of the EMS operation between normal and regenerative mode. At this part of the lap there were successive turns on the race track, where the vehicle was successively accelerated and decelerated, before and after the turn respectively. The states' transitions follow the variations of the velocity profile. Thus, for example, during the first deceleration phase, starting at the 140th second, the FSM is shifted from the q_2 state to the intermediate q_3 and from there instantly to the q_4 state, where it remains for a couple of seconds until the driver accelerates again.

At this acceleration phase the FSM is shifted from q_4 to the intermediate fail safe q_5 and instantly back to q_2, where it remains during the rest of the normal mode operation. For the last 5 s of the lap, the vehicle is standstill due to the stop-and-go procedure, and the FSM state finally changes to q_1, as soon as the velocity value of the vehicle decreases below the velocity threshold.

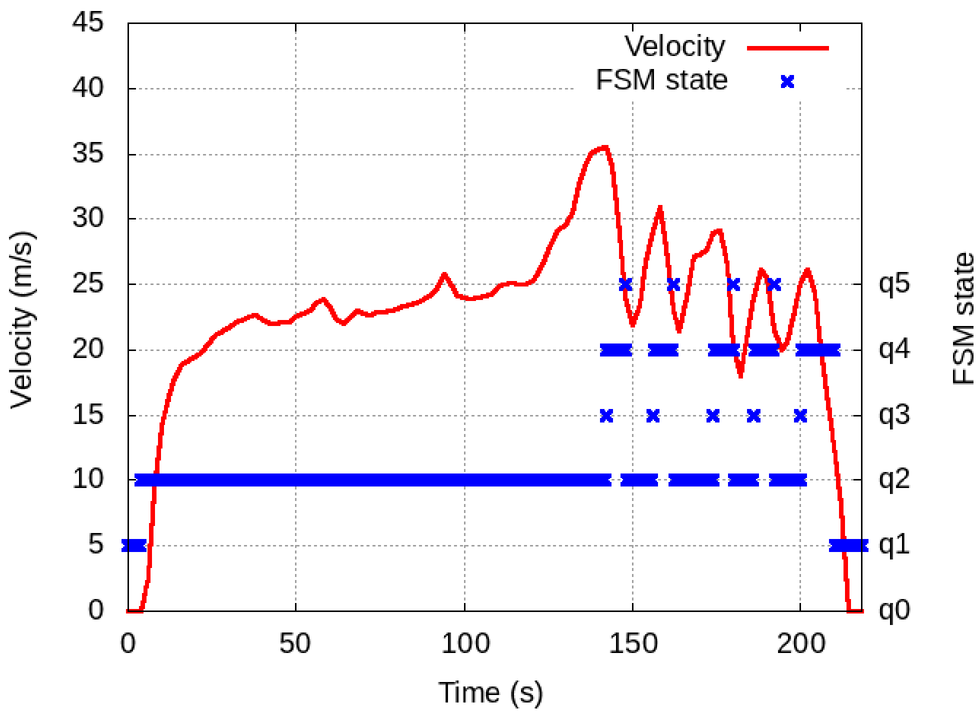

Figure 7. Velocity and FSM states diagram, created from data logged during the 1st lap of the Shell Eco-marathon Europe 2019.

Figure 8 depicts the motor power along with the SC voltage values' variations during the 215 s of the first race lap. As the HEV accelerates until the 140th second the electric power consumed by the motor varies between 75 W and 1080 W approximately. At the same period the voltage value of the SC bank is reducing, almost continuously, as electric energy flows from the bank to the motor and the HEV's PEMFC cannot sufficiently compensate the accelerations' energy bursts. The SC bank voltage looses more than 12 V at this phase of normal mode operation. After the 140th second, during the successional transitions of the EMS operation between normal and regenerative mode due to the race track turns, the motor power turns to zero whenever the HEV decelerates. This happens, for example at the period from the 144th to the 149th second. At the same period there is a sudden increase st the SC bank voltage value, due to the electric energy harvesting during the regenerative mode operation. This procedure of successive transitions between acceleration and deceleration of the HEV is repeated for five times in total, after the 140th second and until the end of the lap. At the last 5 seconds a stop-and-go interval takes place and the FC is charging the SC bank, while the motor power consumption is zero as the vehicle is standstill.

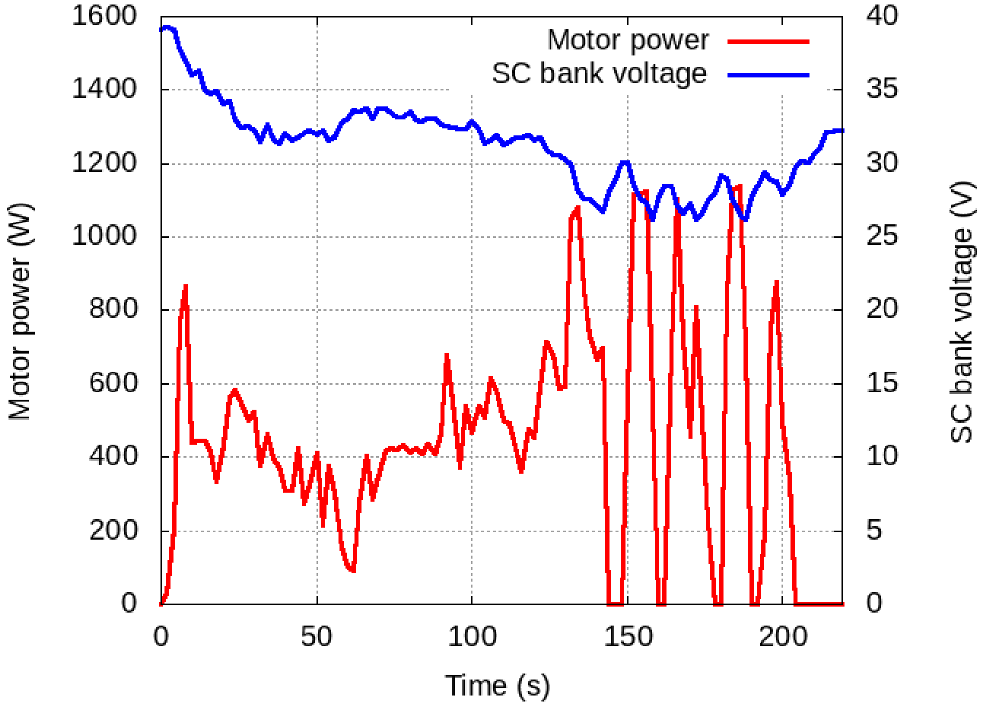

Figure 8. Power and SC voltage diagram, created from data logged during the 1st lap of the Shell Eco-marathon Europe 2019.

Figure 9 depicts the velocity values of the HEV, along with the states' transitions during the 200 s that lasted the second race lap. This lap is slightly quicker than the first one. The HEV accelerates until the 53th second, when it starts to decelerate before the first wide turn of the track, due to its higher, than in the first lap, velocity. At the start of the lap the FSM state is q_1 and immediately shifts to q_2 as the vehicle starts moving. The next state transition will occur at the 53th second when the vehicle decelerates and a regenerative mode operation begins with a successive transition from the q_2 state to the intermediate q_3, to q_4 and to the intermediate q_5. From the 60th until the 75th second and from the 87th until the 125th second of the lap the vehicle also accelerates. The FSM state is q_2 for the whole period from the 53th to 125th second, although there is a brief interval of deceleration between 76th and 85th second. At this interval, there is no transition to a regenerative mode operation because the driver keeps on pressing the acceleration pedal, though less than before and thus the vehicle decelerates. After the 125th second and until the end of the lap, the successional transitions of the EMS operation between normal and regenerative mode are repeated the same way, as at the end of the first lap. After the 195th second, when the velocity value of the vehicle decreases below the velocity threshold, the FSM state changes to q_1. The FSM remains at the q_1 state until the end of the second lap as a stop-and-go procedure follows at the last 5 seconds and the vehicle is at a standstill.

Figure 10 presents the motor power along with the SC variations of the voltage values, during the 200 s of the second race lap. From the start, until the 125th second the electric power values vary between 0 and 1100 W approximately. In general the power values are higher than the ones of the first lap, due to the higher velocity values of the second lap. At the same period, the voltage value of the SC bank is continuously reducing, apart from the periods of vehicle deceleration, as for example from the 74th until the 88th second, where the motor power consumption is reducing and thus the FC has the opportunity to charge the SC bank. A phase of successional transitions between normal and regenerative mode operation follows, due to the race track turns. At this phase, the motor power falls to zero whenever the HEV decelerates, as for example at the period from the 135th to the 144th second. At the same period the SC bank voltage value increases, due to the electric energy harvesting during the regenerative mode operation. At the end of the second lap, at the stop-and-go interval of 5 s, the SC bank voltage value increases as the SCs are charging.

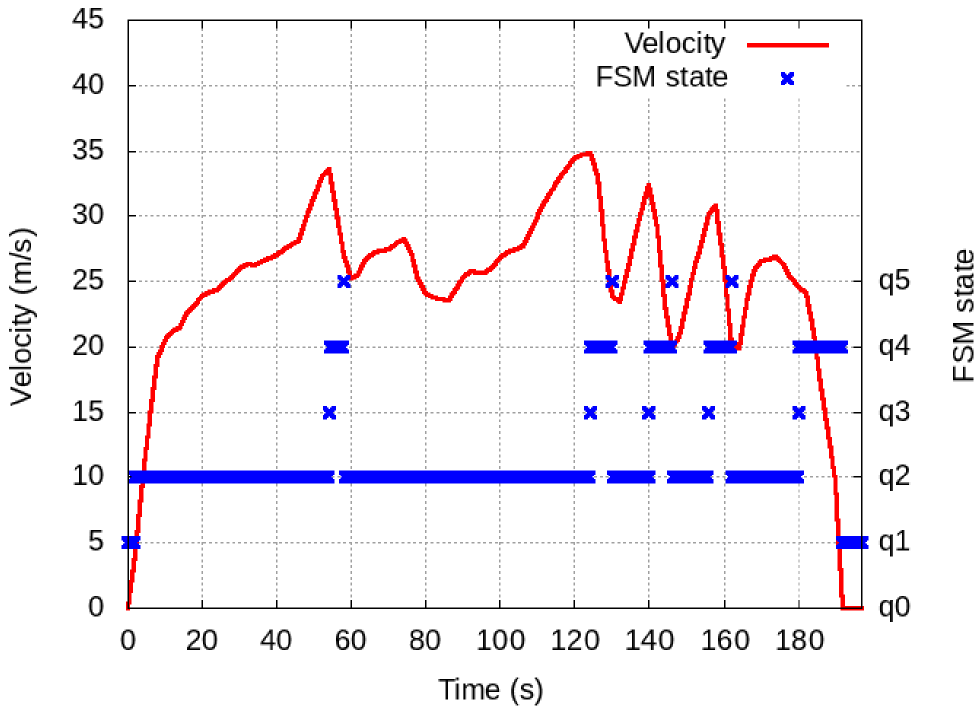

Figure 9. Velocity and FSM states diagram, created from data logged during the 2nd lap of the Shell Eco-marathon Europe 2019.

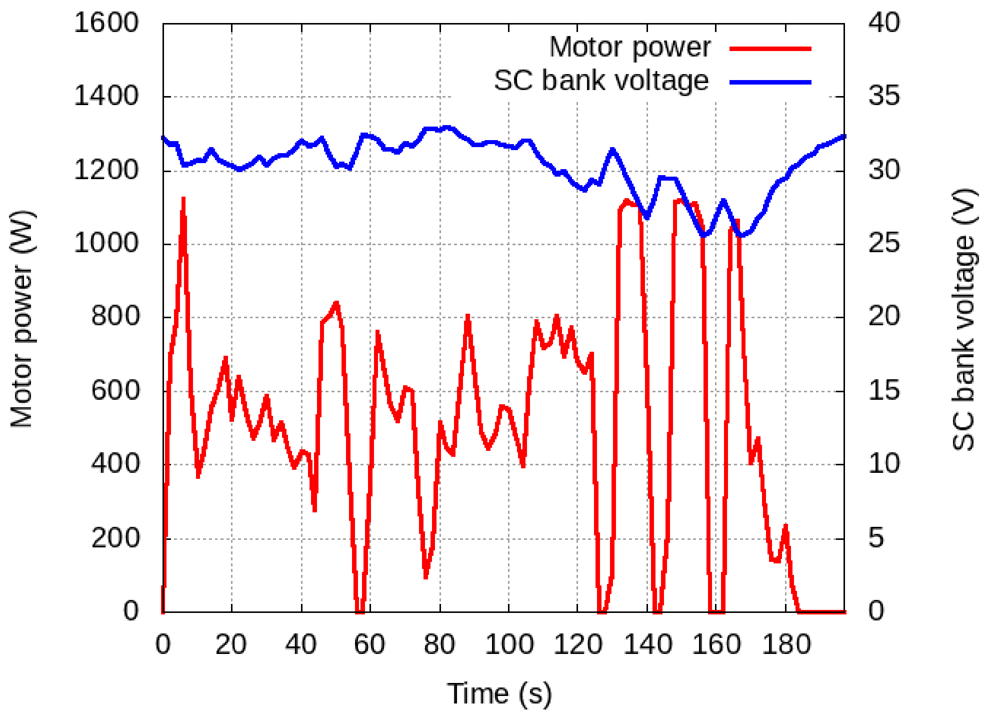

Figure 10. Power and SC voltage diagram, created from data logged during the 2nd lap of the Shell Eco-marathon Europe 2019.

Figures 11 and 12 describe the third lap of the race, that is completed in 210 s. The velocity values of the HEV are increasing after the start of the lap, as it accelerates until the 135th second, apart form a deceleration interval from the 48th to the 58th second. At the start, the FSM state is shifted from the q_1 state to q_2, as the vehicle operates at *normal mode*. The state does not change until the 135th second, apart from the interval from the 48th to the 58th second where a *regenerative mode* phase shifts the FSM state from q_2 to q_3, to q_4 and q_5. From the 135th second until 5 s to the end of the lap there are four successional transitions from *normal* to *regenerative mode* operation where the velocity values are increasing and decreasing respectively. At this period, the transition from the $q2$ state to q_3, to q_4 and q_5, due to the activation and deactivation of a *regenerative mode* operation is repeated four times. After the 210th second, the velocity value of the vehicle decreases below the *velocity threshold* and the FSM state changes to q_1, and remains so until the end of the lap.

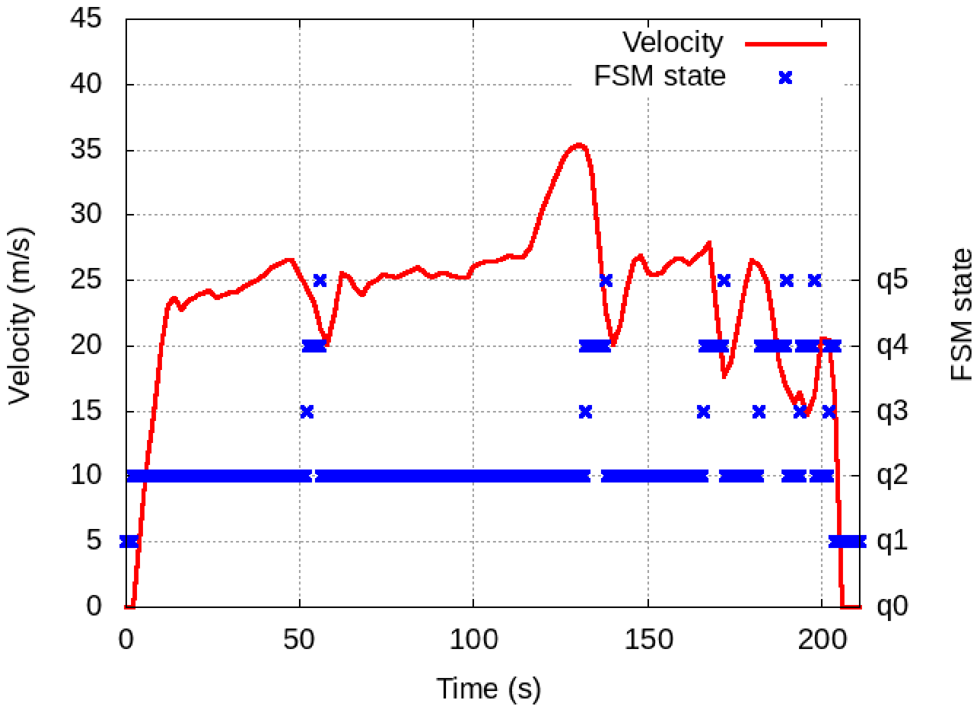

Figure 11. Velocity and FSM states diagram, created from data logged during the 3rd lap of the Shell Eco-marathon Europe 2019.

As shown in Figure 12, the electric power consumed by the motor varies between 0 and 1100 W approximately until the 122th second. The zero values correspond to a *regenerative mode* phase from the 48th to the 58th second. At the same period, from the start until the 122th second, the voltage value of the SC bank is reducing, apart from the deceleration phase of the vehicle, from the 27th to the 58th second, where the value increases due to the low power consumption and the *regenerative mode* phase. After the 122th second, there are four successional transitions of the EMS operation between *normal* and *regenerative mode*, the motor power turns to zero whenever the HEV decelerates and the SC bank voltage value increases and decreases respectively. This happens, for example at the period from the 165th to the 172th second. At the last 5 s of the lap the motor power is zero and the SC bank voltage increases, as the vehicle is standstill due a stop-and-go interval.

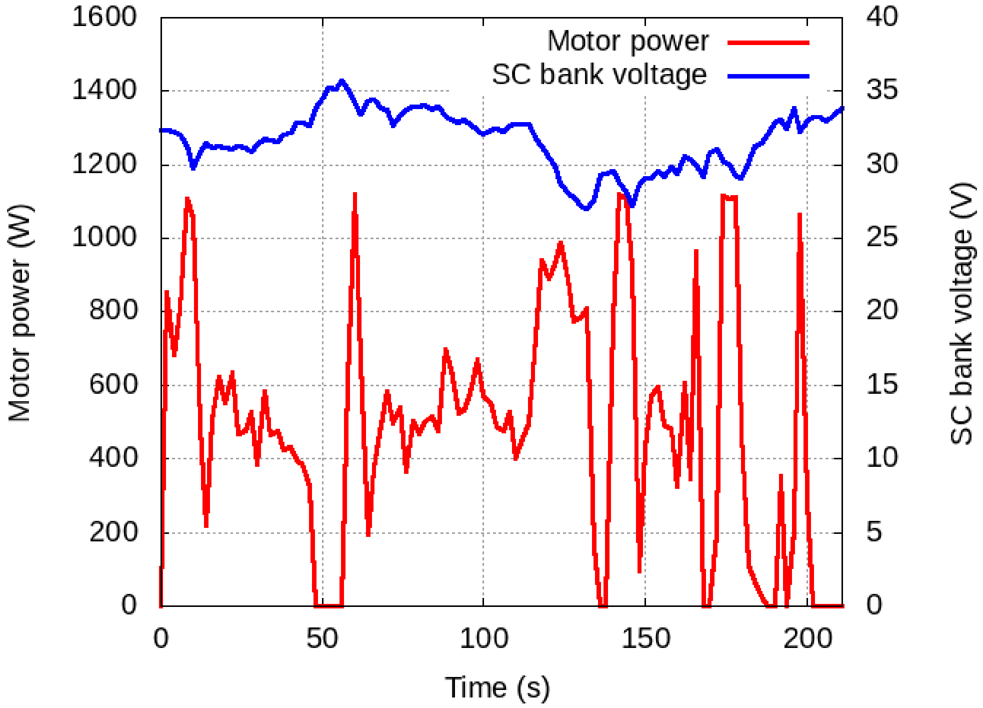

Figure 12. Power and SC voltage diagram, created from data logged during the 3rd lap of the Shell Eco-marathon Europe 2019.

5. Conclusions and Further Work

The development of the Spyros Louis EMS showed that the ROS is a powerful prototyping tool for creating software capable of controlling the automation utilized at the powertrain of HEVs. Apart from the user-friendly and relatively straightforward programming procedure, the ROS software frame offers modularity, reliability and trouble free operation when executing in a low power, low cost, open source, terminal-based Linux development boards with compact dimensions. The ROS-based EMS competed at the races, as a part of the Spyros Louis HEV powertrain, but also operated faultlessly for more than a year now, supporting the educational, research and racing activities of the TUCer team.

The control algorithm of the proposed EMS is a rule-based technique. Rules were edited empirically, using the racing experience of the TUCer team's engineers and drivers. We are currently working on the adoption of optimization techniques that will improve the performance of the ROS-based EMS in terms of low consumption and reliability. Another aspect that we are currently investigating is the integration of the ROS-based EMS in prototype autonomous vehicles (i.e., [33]), therefore allowing the autonomy system to also consider the energy consumption in its driving strategies, aiming at a holistic approach towards sustainable autonomous driving in an urban environment.

Author Contributions: Conceptualization, S.P.; methodology, S.P.; software, S.P., I.C. and N.G.; validation, S.G., I.C. and N.G.; formal analysis, S.G., I.C. and N.G.; investigation, S.G., I.C., N.G. and L.D.; resources, S.P.; data curation, I.C.; writing—original draft preparation, S.G., I.C. and S.P.; writing—review and editing, L.D. and N.T.; visualization, S.P.; supervision, S.P., L.D. and N.T.; project administration, S.P.; funding acquisition, S.P. and N.T. All authors have read and agreed to the published version of the manuscript.

Funding: This work has been partially funded by the Technical University of Crete internal project *TUC Eco Racing team*.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data sharing not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

ECU	Electronic Control Unit
EMS	Energy Management Systems
FSM	Finite State Machine
FC	Fuel Cell
HEV	Hybrid Electric Vehicles
ICE	Internal Combustion Engine
MD	Motor Driver
M/G	Motor/Generator
PEMFC	Proton Exchange Membrane Fuel Cell
ROS	Robotic Operating System
SEM	Shell Eco-marathon
SC	Super Capacitor
SSD	Solid State Relay
TUCer	Technical University of Crete Eco Racing

Appendix A. Electrical System Components

Table A1. Specifications of the electrical system's off-the-shelf components.

Component	Specifications
FC: Horizon 1000XP	PEMFC, 48 V, 1 kW, air cooled
SC: Maxwell BMOD0058E016B02	16 V, 58 F, 22 mOhms
DC/DC: Mean Well SD1000L48	48 V, 21 A, input 19–72 V, buck-boost isolated
M/G: MAC M12500	3-phase, 48 V, 1 kW, 4700 rpm
MD: MAC 12 FET	36–72 V, 40 A
ECU: BeagleBoneBlack Rev. C	ARM, Linux-ROS-based custom plug in board
SSD: Crydom D1D40	SPST-NO, 100 V, 40 A
D: ON Semiconductor RURG3020CC	ultra fast 200 V, 30 A, V_f = 1 V @ 30 A

Appendix B. ECU Development Platform's Specifications

Table A2. Specifications of the ECU development platform.

Specification	Value
processor	AM3358 ARM Cortex-A8
maximum processor speed	1 GHz
co-processors	2xPRUs, ARM Cortex-M3, SGX PowerVR
analog pins	7 @ 1.8 V
GPIO	65 @ 3.3 V
memory	512 MB DDR3, 4 GB on-board storage eMMC, microSD
network	10/100 Ethernet
supported interfaces	4xUART, 12xPWM, LCD, GPMC, MMC1, 2xSPI, 2xI2C, A/D Converter, 2xCAN Bus

References

1. Song, W.; Kim, J.; Park, H.; Nho, E.; Kim, I.; Kim, H.; Chun, T.; Choi, N. Power control of a combined system with fuel cell and supercapacitor. In Proceedings of the 8th International Conference on Power Electronics—ECCE Asia, Jeju, Korea, 29 May–2 June 2011; pp. 2663–2668.
2. Gualous, H.; Hissel, D.; Bontour, S.; Harel, F.; Kauffmann, J.M. Power management of an embedded fuel cell - supercapacitor APU. In Proceedings of the 2005 European Conference on Power Electronics and Applications, Dresden, Germany, 11–14 September 2005; p. 8.
3. Fathabadi, H. Fuel cell hybrid electric vehicle (FCHEV): Novel fuel cell/SC hybrid power generation system. *Energy Convers. Manag.* **2018**, *156*, 192–201. [CrossRef]
4. Carignano, M.; Costa-Castello, R.; Nigro, N.; Junco, S. A Novel Energy Management Strategy for Fuel-Cell/Supercapacitor Hybrid Vehicles. *IFAC-PapersOnLine* **2017**, *50*, 10052–10057. [CrossRef]
5. Hong, Z.; Han, Y.; Li, Q.; Chen, W. Design of Energy Management System for Fuel Cell/Supercapacitor Hybrid Locomotive. In Proceedings of the 2016 IEEE Vehicle Power and Propulsion Conference (VPPC), Hangzhou, China, 17–20 October 2016; pp. 1–5.
6. Marzougui, H.; Amari, M.; Kadri, A.; Bacha, F.; Ghouili, J. Energy management of fuel cell/battery/ultracapacitor in electrical hybrid vehicle. *Int. J. Hydrog. Energy* **2017**, *42*, 8857–8869. [CrossRef]
7. Wu, J.; Wei, Z.; Li, W.; Wang, Y.; Li, Y.; Sauer, D.U. Battery Thermal- and Health-Constrained Energy Management for Hybrid Electric Bus Based on Soft Actor-Critic DRL Algorithm. *IEEE Trans. Ind. Inform.* **2021**, *17*, 3751–3761. [CrossRef]
8. Wu, J.; Wei, Z.; Liu, K.; Quan, Z.; Li, Y. Battery-Involved Energy Management for Hybrid Electric Bus Based on Expert-Assistance Deep Deterministic Policy Gradient Algorithm. *IEEE Trans. Veh. Technol.* **2020**, *69*, 12786–12796. [CrossRef]
9. Ferreira, A.A.; Pomilio, J.A.; Spiazzi, G.; de Araujo Silva, L. Energy Management Fuzzy Logic Supervisory for Electric Vehicle Power Supplies System. *IEEE Trans. Power Electron.* **2008**, *23*, 107–115. [CrossRef]
10. Hames, Y.; Kaya, K.; Baltacioglu, E.; Turksoy, A. Analysis of the control strategies for fuel saving in the hydrogen fuel cell vehicles. *Int. J. Hydrog. Energy* **2018**, *43*, 10810–10821. [CrossRef]
11. Lawrence, C.; ElShatshat, R.; Salama, M.; Fraser, R. An efficient auxiliary system controller for Fuel Cell Electric Vehicle (FCEV). *Energy* **2016**, *116*, 417–428. [CrossRef]
12. Wu, Y.; Gao, H. Optimization of Fuel Cell and Supercapacitor for Fuel-Cell Electric Vehicles. *IEEE Trans. Veh. Technol.* **2006**, *55*, 1748–1755. [CrossRef]
13. Tang, D.; Zio, E.; Yuan, Y.; Zhao, J.; Yan, X. The energy management and optimization strategy for fuel cell hybrid ships. In Proceedings of the 2nd International Conference on System Reliability and Safety (ICSRS), Milan, Italy, 20–22 December 2017; pp. 277–281.
14. Sulaiman, N.; Hannan, M.; Mohamed, A.; Majlan, E.; Daud, W.W. A review on energy management system for fuel cell hybrid electric vehicle: Issues and challenges. *Renew. Sustain. Energy Rev.* **2015**, *52*, 802–814. [CrossRef]
15. Li, H.; Ravey, A.; N'Diaye, A.; Djerdir, A. A Review of Energy Management Strategy for Fuel Cell Hybrid Electric Vehicle. In Proceedings of the 2017 IEEE Vehicle Power and Propulsion Conference (VPPC), Belfort, France, 14–17 December 2017; pp. 1–6.
16. Tazelaar, E.; Veenhuizen, B.; Jagerman, J.; Faassen, T. Energy Management Strategies for fuel cell hybrid vehicles; an overview. In Proceedings of the 2013 World Electric Vehicle Symposium and Exhibition (EVS27), Barcelona, Spain, 17–20 November 2013; pp. 1–12.
17. Njoya Motapon, S.; Dessaint, L.; Al-Haddad, K. A Comparative Study of Energy Management Schemes for a Fuel-Cell Hybrid Emergency Power System of More-Electric Aircraft. *IEEE Trans. Ind. Electron.* **2014**, *61*, 1320–1334. [CrossRef]
18. Li, Q.; Yang, H.; Han, Y.; Li, M.; Chen, W. A state machine strategy based on droop control for an energy management system of PEMFC-battery-supercapacitor hybrid tramway. *Int. J. Hydrog. Energy* **2016**, *41*, 16148–16159. [CrossRef]
19. Li, Q.; Su, B.; Pu, Y.; Han, Y.; Wang, T.; Yin, L.; Chen, W. A State Machine Control Based on Equivalent Consumption Minimization for Fuel Cell/ Supercapacitor Hybrid Tramway. *IEEE Trans. Transp. Electrif.* **2019**, *5*, 552–564. [CrossRef]
20. TUCer team. Technical University of Crete Eco Racing Team. Available online: https://www.tucer.tuc.gr (accessed on 23 January 2021).
21. Quigley, M.; Conley, K.; Gerkey, B.P.; Faust, J.; Foote, T.; Leibs, J.; Wheeler, R.C.; Ng, A.Y. ROS: An open-source Robot Operating System. In Proceedings of the ICRA Workshop Open Source Software, Kobe, Japan, 12–17 May 2009.
22. Koubaa, A. *Robot Operating System (ROS) The Complete Reference (Volume 2)*; Springer: Cham, Switzerland, 2017.
23. Energy Efficiency and Renewable Energy, Fuel Cell Technology Office. Comparison of Fuel Cell Technologies. Available online: https://www.energy.gov/eere/fuelcells/comparison-fuel-cell-technologies (accessed on 27 January 2021).
24. Mat, Z.B.A.; Madya.; Kar, Y.B.; Hassan, S.H.B.A.; Talik, N.A.B. Proton exchange membrane (PEM) and solid oxide (SOFC) fuel cell based vehicles-a review. In Proceedings of the 2017 2nd IEEE International Conference on Intelligent Transportation Engineering (ICITE), Singapore, 1–3 September 2017; pp. 123–126.
25. Thounthong, P.; Sethakul, P. Analysis of a Fuel Starvation Phenomenon of a PEM Fuel Cell. In Proceedings of the 2007 Power Conversion Conference, Nagoya, Japan, 2–5 April 2007; pp. 731–738.
26. Naidu, B.R.; Panda, G. A hybrid fuel cell-supercapacitor system employing adaptive control to overcome fuel starvation phenomenon of fuel cell. In Proceedings of the 2017 Innovations in Power and Advanced Computing Technologies (i-PACT), Vellore, India, 21–22 April 2017; pp. 1–5.

27. Thounthong, P.; Sethakul, P.; Rael, S.; Davat, B. Performance Investigation of Fuel Cell/Battery and Fuel Cell/Supercapacitor Hybrid Sources for Electric Vehicle Applications. In Proceedings of the 2008 4th IET Conference on Power Electronics, Machines and Drives, York, UK, 2–4 April 2008; pp. 455–459.
28. BeagleBoard. BeagleBone Black Development Platform.Available online: https://beagleboard.org/black (accessed on 27 January 2021).
29. ROS Index. ROS and ROS 2 Resources. Available online: https://index.ros.org (accessed on 27 January 2021).
30. Kohavi, Z.; Jha, N.K. *Switching and Finite Automata Theory*, 3rd ed.; Cambridge University Press: Cambridge, UK, 2009.
31. Tzortzis, G.; Amargianos, A.; Piperidis, S.; Koutroulis, E.; Tsourveloudis, N.C. Development of a compact regenerative braking system for electric vehicles. In Proceedings of the 23rd Mediterranean Conference on Control and Automation (MED), Torremolinos, Spain, 16–19 June 2015; pp. 102–108.
32. Piperidis, S.; Chrysomallis, I.; Georgakopoulos, S.; Stefanoulis, T.; Ghionis, N.; Katsifas, V.; Tsourveloudis, N.C. Development of a ROS controlled chassis dynamometer for lightweight, single seater EVs. In Proceedings of the 28th Mediterranean Conference on Control and Automation (MED), Saint-Raphael, France, 15–18 September 2020.
33. Sarantinoudis, N.; Spanoudakis, P.; Doitsidis, L.; Stefanoulis, T.; Tsourveloudis, N. A study of electric vehicle prototype for shell eco-marathon. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Madrid, Spain, 30 October–5 November 2018.