

Development of an Embedded System for Building System Management Based on PV-Powered

Hussein Ali* and Ergun Erçelebi


Abstract—As complexity and the type of electrical applications perpetually expands, monitor and control will require new systems which are based on some features in such feature of self-powered and distantly management. Building management systems (BMSs) are developed as efficient solutions to monitor and control users electrical and mechanical applications. A BMS employs cheap sensors and a real-time module to create prior knowledge about the climate inside the building. As a result, these features give some attractive advantages in taking a precise decision and managing vital operations of electrical applications. In this paper, a Raspberry Pi as minicomputer for collecting information from server room or data centre for detection the existence of human, temperature and humidity sensor and gas sensors to control ventilation systems, Arduino platform is responsible for receiving sensors signal and control actions according to the statues in the server room. While the Xbee protocol has been used to transmit data from the sensing node (Raspberry PI3) to mentors and control nodes (Arduino mega2560). The prototype design has been verified experimentally.

Index Terms—Building management systems, Power Saving, Smart Power System, Raspberry PI 3, XBee modules.

I. INTRODUCTION

POWER CONSUMPTION and security systems for buildings is one of the topics has great interest at this time, all the time the institutions, researchers and governments looking for keeping every topic related power consumption, security systems for buildings, firefighting systems, cooling systems under development by conception of smart management systems. In the last decades, the application of embedded systems encouraged Stakeholders to develop embedded systems for managing and monitoring the energy units all the time. These systems are developed based on a modern wireless connection and minicomputers.

HUSSEIN ALI, is with Department of Electrical Engineering University of Gaziantep, Gaziantep, Turkey, (e-mail: hussain.a.abd1986@gmail.com).

 <https://orcid.org/0000-0003-2543-1205>

ERGUN ERÇELEBI, is with Department of Electrical Engineering University of Gaziantep, Gaziantep, Turkey, (e-mail: ercelebi@gantep.edu.tr).

 <https://orcid.org/0000-0002-4289-7026>

Manuscript received May 6, 2021; accepted July 26, 2021.

DOI: [10.17694/bajece.933353](https://doi.org/10.17694/bajece.933353)

Most embedded systems required energy storage to stabilize the operation of the system and prevent any damage to the OS of embedded systems. However, power consumption represents a critical concern for embedded devices. There are always need to increase battery life, which contributes to reduce the system's environmental impact. Previously, the improving battery life was exclusively responsibility by the hardware design, but those days are gone. Software in embedded systems has recently taken responsibility for controlling power consumption. In the same way, power consumption represents a major problem in embedded systems, while self-powered systems have become extensively used. As the device size has become smaller and thinner, this limitation prevents the system from incorporating a large capacity battery for long-term use [1].

In order to manage power consumption, the manufacturing people can perform on either hardware or software. On the other hand, their techniques are widely used such as; clock-gating clock off for the region does not need to work anymore, using multiple threshold voltage levels for different blocks with different operating speeds, power-gating turn off the block that is no longer working, and dynamic frequency/voltage scaling (DFVS), and frequency/voltage grading in order to appropriate workloads [2–5]. On the software, solutions are often focused on optimizing compiler. These assistances to build small and highly efficient execution code.

Regardless the software and interface facilities, a data centre must provide many important features like: consistent, stable power, even in the event of a utility outage, as well as cooling, physical site security, lighting, firefighting, access control and other systems [6]. at this time, many of these systems are wildly used in a typical building, but uses it in a data centre requires added capabilities or special measures owing to the critical purposes. for instance, fire suppression should ideally enable extinguishing fires without destroying servers and other equipment.

Data centres remain critical unite in automation systems, which encompass energy-hungry infrastructures that run large-scale Internet-based services. Energy consumption models are pivotal in designing and optimizing energy-efficient operations to curb excessive energy consumption in data centres [7].

A detailed BMS was designed and implemented as the

primary part of system to collect data and monitor alarms in the event of malfunctions or significant failures. Another important point related server's room and data centre protection that each data centre servers includes data base and software like example digital transfer projects, these databases need to working without any reasons for failure at any time. For these reasons building management system for server's room and data centre founded in order to solving some of failure reasons.

However, the protection of fire due to overheating in electronic components and controlling to air ventilation remotely has still obscure [3,4,7]..

In this paper, a Raspberry Pi has been used as min computer unite to collect sensors signal from server room or data centre such that signals for detection the existence of human, temperature and humidity sensor and gas sensors to control ventilation systems, Arduino Mega2560 responsible for receiving a decision from Raspberry Pi and take control actions according to the server room statues. While Xbee protocol has been used to transmit data from the sensing node to the mentoring node.

II. COMPONENTS OF DEVELOPED BMS

The core parts of the proposed design of BMS are Raspberry Pi 4B and Arduino mega2560 [8,9] .The Raspberry Pi 4B works as a mini-computer that is responsible to collect sensors measurement from two types of sensors SHT3x for temperature/humidity and MQ2 for gas/smoke sensing. In addition to that, Raspberry Pi includes a program responsible for takes actions according to the sensors reading and send this action to Arduino mega2560 through Xbee module. Arduino mega2560 is responsible to take fast actions under the critical situation of fire or smoke occurred in the closed room. In addition to the developed system has the capability to control the temperature and humidity of the room and also take action to turn on or turn off the cooling system based on the status. These actions are taken according to the control signal coming from the mini-computer. The block diagram of the developed system is illustrated in Fig1.

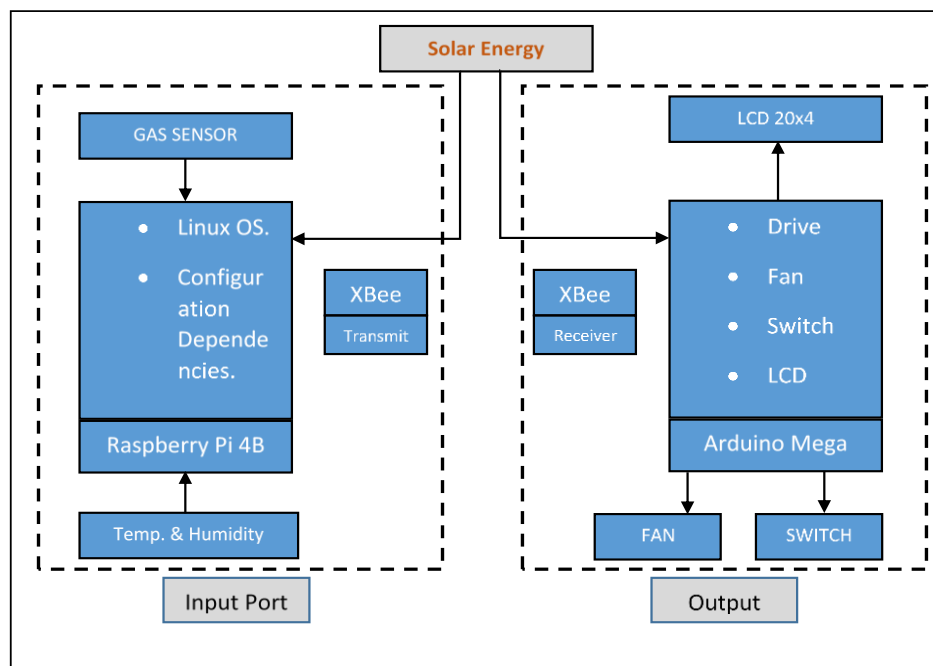


Fig1 Conceptual design of the proposed system.

According to the conceptual block diagram of BMS, Raspberry Pi putting in a data centre room. The two types of sensors connect with Raspberry Pi via digital pins, the benefits of these sensors are checking the environment inside the room like temperature, humidity, and during fire alarm statues the control action is directly open the door. Moreover, fire status appears on LCD.

A. Algorithms for data collate and take control of actions

Fig 2 describes the algorithm of Raspberry Pi Python code. This algorithm is used for data collection from different sensors inside the server room or data center and send by using XBee protocol to the receiver site. The receiver site consists of Arduino mega2560 with XBee shield, the input/output pins of

Arduino mega2560 are utilised to qualified control signal according to action that been taken. Also, LCD is used to display the situation of control action. The algorithm of control action is shown in Fig 3. This algorithm is written in C++ program code in order to proper the architecture of Arduino.

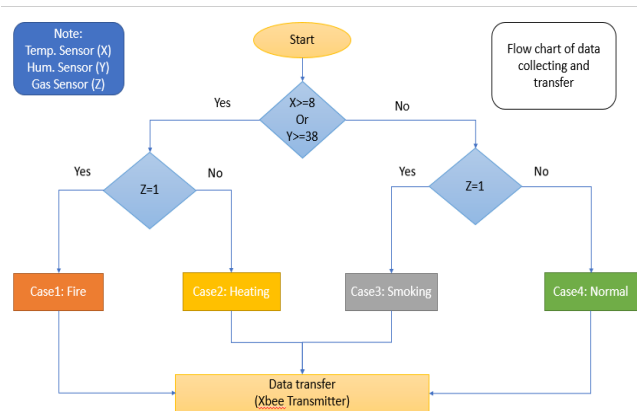


Fig 2. flow chart of data collection and transfer via Xbee to receiver site.

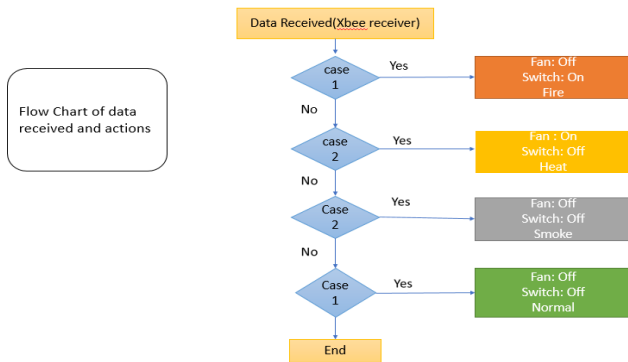


Fig3. flow chart of received data and take actions at Arduino MEGA 2560.

III. SIMULATION RESULTS AND DESIGN DESCRIPTION

The developed system has been implemented using LabVIEW simulation program based on the block diagram in Fig 1. On the other hand, the prototype hardware consists of Raspberry Pi for collecting data of sensors reading from inside the building all the time[10]. The Xbee s2 module is connected with Raspberry Pi in order to ensure a fast transfer of measured data to the monitoring side. On the monitoring and control side, Arduino connected with the Xbee shield in order to ensure receiving sensors signals from Raspberry Pi and begin to take action for cooling, fire alarm, access control. The Arduino MEGA 2560 pins are used as output for sending decision signals to access control devices. The switch or really consists of electric-magnetic used as electromechanical door locking, which is under fire status the Arduino sending open door action to electromechanical lock that would be satisfaction more safety stranded inside the building.

In addition to that the setting for the cooling system to turn on at temperature 8 °C and humidity 38 presents. Furthermore, the setting values of temperature and humidity for the cooling system is easily adjustable. Under fire case if there heating and smoke the fire case will be starting and safety action will be starting also, the Smoke case can appear on HMI (LCD) in case there is such a level of smoke inside. Finally, the solar panel has been used as main source for system power, MPPT and batteries also in order to keep PC economy. The simulation blocks of the developed BMS under the simulation program appears in Fig 4. While, the interface blocks of the BMS have appeared in Fig 5.

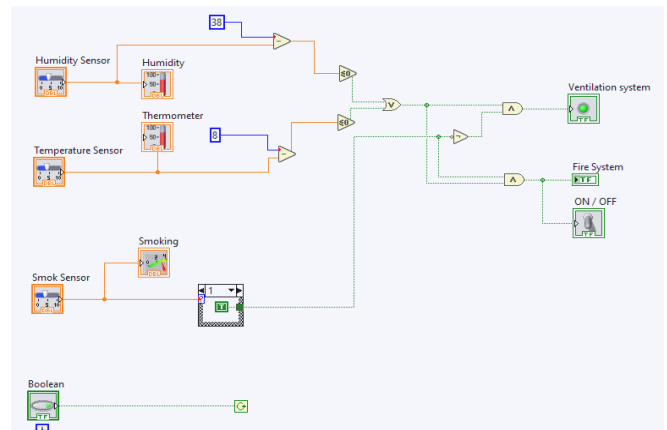


Fig4.Simulation blocks for developed BMS system under simulation program.

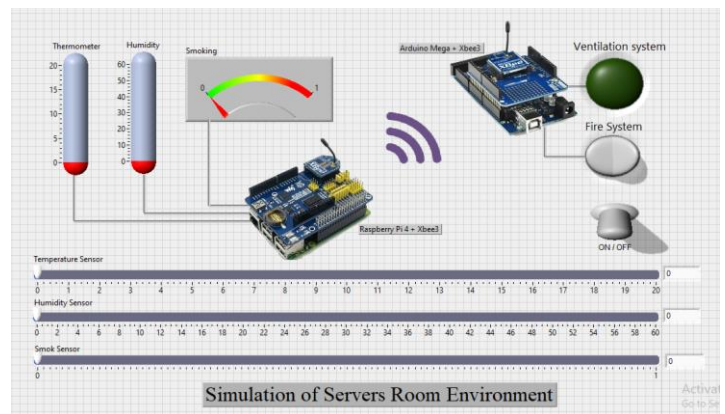


Fig 5 interface blocks for developed BMS system under simulation program.

The prototype hardware implementation for the BMS is shown in Fig 6. finally, the system is powered by a PV -source with a battery the connection of the BMS energy source appears in Fig 7.

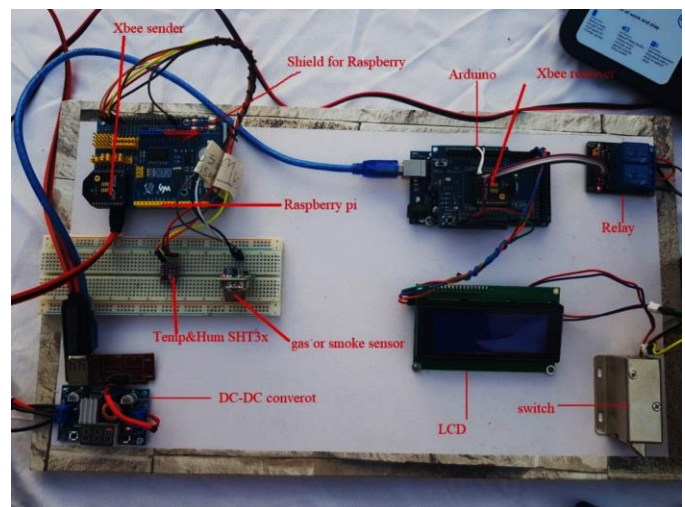


Fig 6 Hardware implementation of the developed BMS.

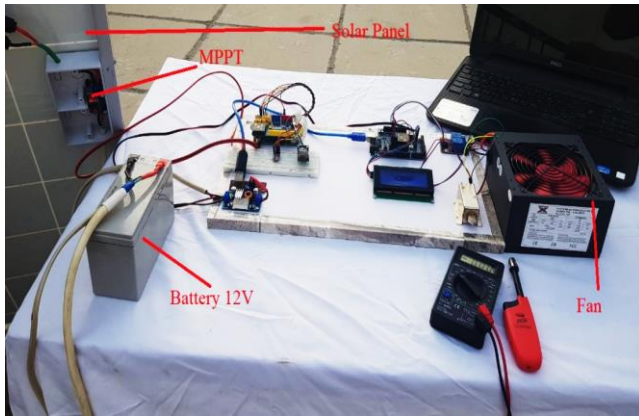


Fig 7. Hardware implementation of solar and MPPT.

IV. DISCUSSION

The critical aspect of the developed design was how we could clarify the main environmental issue inside the server room or data centre room. Therefore, the LabVIEW simulation software was utilised to evaluate the performance of developed BMS. In performance evaluate four different cases have been studied; heating sensor SHT3x, humidity sensor SHT3x, smoke and gas leakage detector sensor MQ2 (fire), and smoke and gas leakage detector sensor MQ2 (smoke). The status of these events appears on the monitoring LCD and the results are listed in Table I.

- *Case 1: Heating sensor SHT3x:*When the temperature degree inside closed building change to 8°C or greater than 8°C, the temperature sensor inside the server room will send the signal statues to the Raspberry pi in order to collect information for all sensors inside the room, after that the Raspberry pi will send the signals by using XBEE protocol to Arduino outside to the room near to the door of the server room, Arduino will take

control action according to the above-mentioned (see in Fig 3). This activates will be available on the LCD monitor device installed outside. The result of simulation interface of this case appears in Fig 8a.

- *Case 2 Humidity sensor SHT3x:* When the humidity presents inside closed building change to equal 38 or greater than 38, the humidity sensor inside the server room will send the signal statues to the Raspberry pi in order to collect information for all sensors inside the room, after that the Raspberry pi will send the signals by using XBEE protocol for Arduino outside to the room near to the door of the server room. Fig 8b describes the simulation interface of case 2.

- *Case 3: Smoke and gas leakage detector sensor MQ2 (fire):* When the heating and smoke statues level change inside a closed building, the temperature sensor and smoke sensor inside the server room will send the signal statues to the Raspberry pi in order to collect information for all sensors inside the room, after that the Raspberry pi will send the signals by using XBEE protocol for Arduino outside to the room near to the door of the server room. In this event, all activities stop run and the magnetic lock of the door is directly open. Fig 8c describes case 3.

- *Case 4 Smoke and gas leakage detector sensor MQ2 (smoke):*In this case, the smoke sensor will be sensitive for the low level from smoke, if there is and little smoke inside the server room without heating that means there are no fire statues, so the smoke sensor will send the signal to the Raspberry pi in order to collect data for all sensors inside the room, after that the Raspberry pi will send the signals by using XBEE protocol for Arduino outside to the room near to the door of the server room, Arduino will take action according to the statues. Fig 8 d describes case 4.

TABLE I
BMS SYSTEM OPERATION STATUS ON LCD.

Statues	LCD- status	Action
Heating and Humidity	Temperature: 28°C Humidity: 50%RH Status: Hot Cooling	Cooling system is turn on
Smoke /gas leakage	Temperature: 25°C Humidity: 56%RH Status: Smoking Normal	smoke inside normal/ fire alarm
Smoke/ gas leakage (Fire)	Temperature: 27°C Humidity: 53%RH Status: Fire Site Under Maint	Fire inside the door unluck and turn off all activities

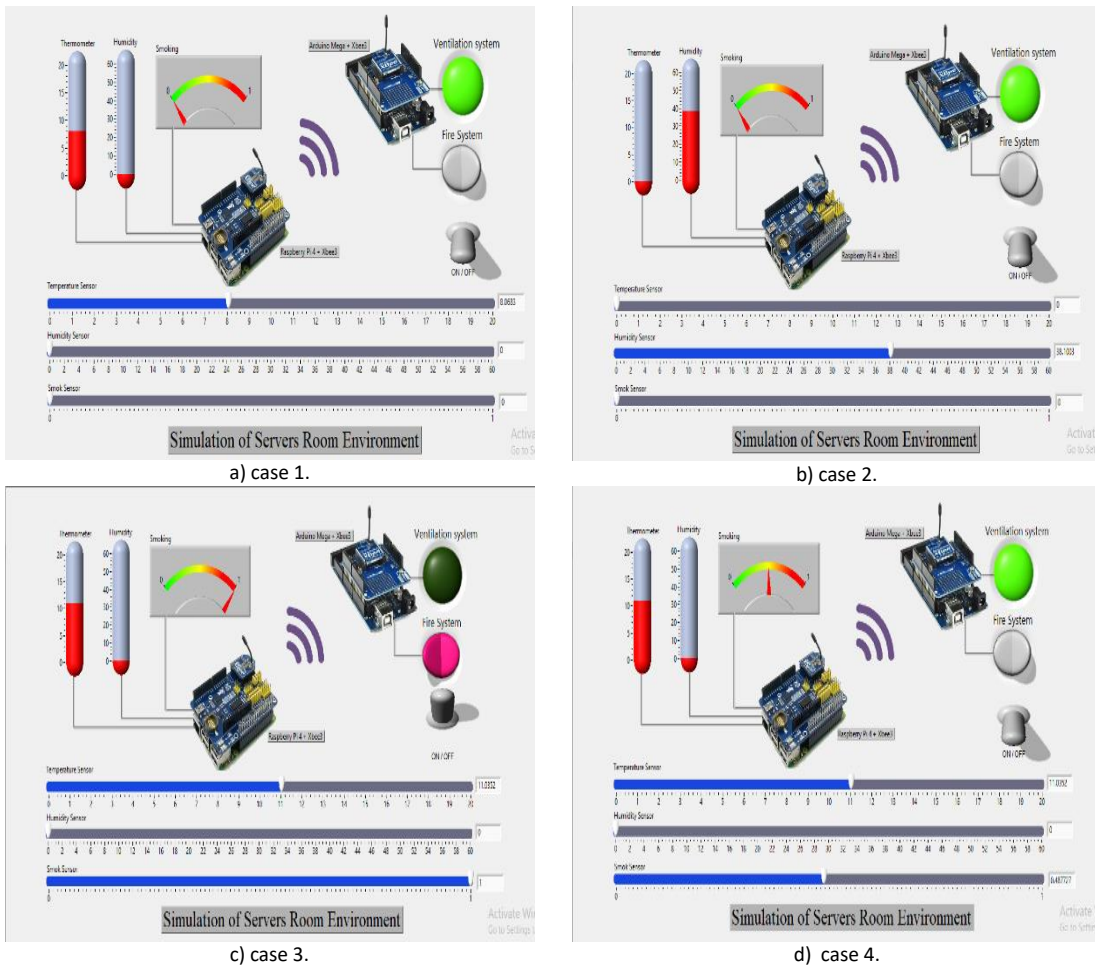


Fig 8 evaluate the BMS performance against different events.

TABLE II
COMPARISON WITH SAME SYSTEM IN LITERATURE

References	Control characteristics	Design details
[6]	Raspberry Pi	Re-distributed heat on building by using temperature sensors. The proposed system depends on some classical control strategy and the flexibility of the open-source Building energy Management System.
[11]	1.Arduino Mega 2560 2. Raspberry Pi	Development a remotely controllable home energy management and control system with support control through mobile.
[12]	1. Arduino Uno 2. Raspberry Pi	Developed energy management system in order to energy conservation at household.
Proposed BMS	1.Arduino Mega 2560 2. Raspberry Pi	Classical control is used and in dangerous conditions, the decision is taken automatically depend on prior settings while the action appears to the user.

A primary design objective of the BMS system for the server room is the minimum total cost. The total cost increases with increasing the number of features because increase the required number of sensors such as camera or motion detection sensor. The number of major components and the prices of these components for BMS of server room are tabulated in Table III.

The systems developed [6,11,12] using a traditional power source which includes a battery for supplying the components of the system. In contrast to that, the proposed BMS has a feature that is self-powered by PV and with a backup energy source by battery.

TABLE III. THE NUMBER OF MAJOR COMPONENTS FOR BMS WITH COST OF EACH COMPONENT.

Component	Number of items	Local price
Raspberry Pi	1	25\$
Arduino Mega 2560	1	7\$
Xbee S2 Module	2	12\$
Solar panel 10W, 12 V+MPPT	1	15\$
Battery	1	4\$
Heating sensor SHT3x	1	0.5\$
Gas Sensor (MQ2) module	1	0.25\$
LCD 16x4	1	3\$
Approximately total price	8	66.75\$

V. CONCLUSION

In this paper, the design and implementation of a flexible solution for a building management system for the server room or data centre is presented, which essentially provides safety as well as comfortable work at a data centre. In addition to that, it is featured by simply install, cheap technologies for design, also the control and watch of devices insides a building can be easier without visiting the place of a device. Moreover, it provides a status of temperatures, fire if that occurred inside and smoke leak by using wireless technology to transmits events inside building to the monitoring station. Accordingly, the building management system that has been presented in this paper works properly for electric power management, mainly in the server room or data centre.

ACKNOWLEDGMENT

The author likes to recognise the observable support from PhD. Ahmed K. Ali in the Al Mustansiriyah University, Baghdad-Iraq for his precious discussions and comments.

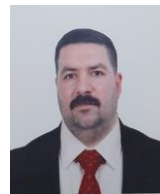
REFERENCES

- [1] Do QN, Le TC. Low Power Embedded System Design Using Code Compression 2015:0–4.
- [2] Emmett F, Biegel M. Power Reduction Through RTL Clock Gating. Power 2000.
- [3] Lee J, Nam BG, Yoo HJ. Dynamic Voltage and Frequency Scaling (DVFS) scheme for multi-domains power management. 2007 IEEE Asian Solid-State Circuits Conf A-SSCC 2007:360–3. doi:10.1109/ASSCC.2007.4425705.
- [4] System M, Li W, Member GS, Logenthiran T, Member S. Implemented IoT-Based Self-Learning Home Management System (SHMS) for Singapore 2018;5:2212–9.
- [5] Paolo Ricci P, Donatelli M, Falabella A, Mazza A, Onofri M. Evolution of the Building Management System in the INFN CNAF Tier-1 data center facility. J Phys Conf Ser 2017;898. doi:10.1088/1742-6596/898/8/082005.
- [6] Jones SR, Beardmore J, Gillott M, Boukhanouf R, Walker G. A control methodology for Building Energy Management Systems (BEMS) in heat networks with distributed generation. Energy Procedia 2018;153:295–302. doi:10.1016/j.egypro.2018.10.011.
- [7] Benson T, Anand A, Akella A, Zhang M. Understanding data center traffic characteristics. Comput Commun Rev 2010;40:92–9. doi:10.1145/1672308.1672325.
- [8] Pi UR, B RPM-. Internet of Things Based Home

Security. 2018 Fourth Int Conf Comput Commun Control Autom n.d.:1–6.

- [9] Audio C, Storage CE, Power C, Basics L, Debian I, External U, et al. Raspberry Pi ® User Guide. n.d.
- [10] Blaifi S, Moulahoum S, Colak I, Merrouche W. Monitoring and enhanced dynamic modeling of battery by genetic algorithm using LabVIEW applied in photovoltaic system. Electr Eng 2018;100:1021–38. doi:10.1007/s00202-017-0567-6.
- [11] Barnes V, Collins TK, Mills GA. Design and implementation of home energy and power management and control system. Midwest Symp Circuits Syst 2017;2017-Augus:241–4. doi:10.1109/MWSCAS.2017.8052905.
- [12] Dhobale KM, Gorgile SP, Gunjal PJ, Hirve KA, U. A. PM. IOT based Smart Water Supply management System. Ijarcce 2017;6:297–9. doi:10.17148/ijarcce.2017.6157.

BIOGRAPHIES



HUSSEIN ALI, Iraq, Baghdad in 2009, he received the B.S in computers and information technology from university of Anbar –College of computers.

From 2019 to 2021 he was preparing to M.S in Gaziantep university – electrical and electronic engineering college, he starting research in building management system and prepare article includes model for BMS system for data centre or control centre, Practical and Laboratory experiments for his thesis and acritical focused on the same subject for BMS system.



ERGUN ERCELEBI, received the B.S. degree in electrical and electronics engineering from M.E.T.U., Gaziantep, Turkey in 1990 and M.S. and PhD. degrees in electrical and electronics engineering from the University of Gaziantep in 1992 and 1999, respectively.

Between 2000–2005, he was Assistant Professor of Electrical and Electronics Engineering at the University of Gaziantep. He was the head of computer engineering between 2003 and 2004. He is presently a full Professor.