Heliyon 5 (2019) e02474

Contents lists available at ScienceDirect

Heliyon

journal homepage: www.heliyon.com

Low-cost web-based Supervisory Control and Data Acquisition system for a microgrid testbed: A case study in design and implementation for academic and research applications

Carlos Vargas-Salgado [a,b,*], Jesus Aguila-Leon [c], Cristian Chiñas-Palacios [c], Elías Hurtado-Perez [a]

[a] Departamento de Ingeniería Eléctrica, Universitat Politècnica de València, Valencia, Spain
[b] Instituto Universitario de Ingeniería Energética, Universitat Politècnica de València, Valencia, Spain
[c] Departamento de Estudios del Agua y de la Energía, Universidad de Guadalajara, Centro Universitario de Tonalá, Tonalá, Mexico

ARTICLE INFO

Keywords:
Electrical engineering
Energy
Data analysis
Data analytics
Data visualization
Control system design
Power control system
Electrical system
Renewable energy resources
Raspberry
Hybrid renewable energy system
Web-based SCADA
Cloud computing
Arduino
Remote control and wireless monitoring

ABSTRACT

This paper presents the design and implementation of a low-cost Supervisory Control and Data Acquisition system based on a Web interface to be applied to a Hybrid Renewable Energy System (HRES) microgrid. This development will provide a reliable and low-cost control and data acquisition systems for the Renewable Energy Laboratory at Universitat Politècnica de València (LabDER-UPV) in Spain, oriented to the research on microgrid stability and energy generation. The developed low-cost SCADA operates on a microgrid that incorporates a photovoltaic array, a wind turbine, a biomass gasification plant and a battery bank as an energy storage system. Sensors and power meters for electrical parameters, such as voltage, current, frequency, power factor, power generation, and energy consumption, were processed digitally and integrated into Arduino-based devices. A master device on a Raspberry-PI board was set up to send all this information to a local database (DB), and a MySQL Web-DB linked to a Web SCADA interface, programmed in HTML5. The communications protocols include TCP/IP, I2C, SPI, and Serial communication; Arduino-based slave devices communicate with the master Raspberry-PI using NRF24L01 wireless radio frequency transceivers. Finally, a comparison between a standard SCADA against the developed Web-based SCADA system is carried out. The results of the operative tests and the cost comparison of the own-designed developed Web-SCADA system prove its reliability and low-cost, on average an 86% cheaper than a standard brandmark solution, for controlling, monitoring and data logging information, as well as for local and remote operation system when applied to the HRES microgrid testbed.

1. Introduction

Electricity demand has steadily increased due to the growths of the population around the world. In parallel, conventional grids have evolved into intelligent grids, better known as Smart Grids, and the renewable energy sources participation in the electricity generation has increased, in many cases in the form of microgrid systems. This penetration of decentralised renewable sources in the grid, like microgrids, has produced the inclusion of Information Technologies for the last decade to provide management of energy, data and communications issues for those systems, but this massive inclusion has a lack of standardisation [1, 2]. Because microgrid systems are themselves, the integration of many renewable energy sources and energy storage systems, a microgrid can be designed following one of two main control topologies: centralised or decentralised. No matter which topology is selected, data flow and communications are essential for any decision-making controller [3, 4]. Conventional microgrid controllers are often based on Programmable Logic Controllers (PLC) [5], dedicated computers microgrid simulators [6] and microcontroller-based devices [7, 8]. Selection of a proper controller should be addressed accordingly to the microgrid application, financial budget and security issues. Due to the lack of standardisation on control topology and controller hardware technology, there are many communication protocols available and very different characteristics should be considered [9].

In this paper, the methodology for the development of a low-cost SCADA system for an experimental microgrid test-bed for academic and research proposes is presented, as well as the results of the operational tests and the required investment for the system implementation.

* Corresponding author.
E-mail address: carvarsa@upvnet.up.es (C. Vargas-Salgado).

https://doi.org/10.1016/j.heliyon.2019.e02474
Received 19 November 2018; Received in revised form 8 March 2019; Accepted 11 September 2019

Since SCADA systems are nowadays becoming an essential part of electric power systems, such as microgrids, it is essential to consider an accurate testbed development. Five main approaches for testbed developments, shown in Table 1, can be considered [10].

The low-cost SCADA system developed in this paper is intended to operate over a physical replication of a microgrid, so the fidelity and reliability conditions could be excellent, according to the evaluation of Table 1. Conventional SCADA systems for a microgrid are expensive due to the need for data loggers, controllers, sensors and other devices. Therefore their use is limited to industrial applications. Therefore, the development of a low-cost SCADA system development and integration methodology should improve the cost-effectiveness and repeatability associated with the deployment of SCADA systems in microgrid physical testbeds.

A microgrid management system should be aware of the current status of the system, including sensing, data collecting, and communications. Standard parameters to be measured in microgrids are environmental variables such as solar irradiance, temperature and wind speed. Also, electrical parameters such as frequency, apparent, active and reactive electric powers should be collected. Some SCADA systems are specialised in this type of monitoring [11, 12], and, by measuring such electrical parameters, they allow for a forecast of the microgrid operation and also for the microgrid health and ageing assessment, as presented in [13, 14]. Low-cost monitoring systems is a relevant topic, especially for academic and research applications where financial budgets are often limited. Therefore, efforts have been made for a cheaper solution to collect and display data from electricity, gas or environmental sensors in experimental microgrids [15, 16, 17, 18], given industrial solutions are often expensive and not suitable for small scale applications.

An additional improvement to these control and data systems is the possibility to be accessed from any place with internet access. In this case, every device should be connected to "the Cloud", Therefore, electronic devices must send data via Internet by means of a communication protocol to keep them stored on a Web database that enables to display them on a Web page [19, 20], allowing microgrid interoperability [21]. Several papers mentioned the integration of the Raspberry Pi with Arduino [22, 23]. The development of a low-cost SCADA system for a stand-alone photovoltaic system is presented in [24], where the authors measured environmental variables and power generation from the photovoltaic system using an Arduino UNO development board. The cost of development reported is as low as $ 62. However, this system was limited only to monitoring a single renewable energy source and to wired communication [25]. presents a low-cost SCADA system for wireless remote control and monitoring for a single power inverter. The hardware used by the authors includes an Arduino development board, a Raspberry development board, an ESP12E wireless transceiver and a Wi-Fi shield for Arduino.

This paper presents the design and implementation of a low-cost SCADA system applied to an experimental microgrid. The system presented is more complicated than considered in [24] and [25] because it integrates wireless control and monitoring for several renewable energy sources and an energy storage system. The proposed system is an alternative for commercial SCADA and a solution for modular affordable monitoring and control systems for small to medium scale applications in renewable energy laboratories.

Table 1
Testbed development approaches, taken from [10].

Testbed approach	Fidelity	Repeatability	Accuracy	Safety	Cost-effective	Reliability	Scalability
Physical replication	Excellent	Poor	Moderate	Poor	Poor	Excellent	Poor
Simulated	Low	Moderate	Poor	Excellent	Excellent	Poor	High
Virtual	Moderate	High	Moderate	Excellent	Moderate	Moderate	Moderate
Virtual-Physical	High	High	Excellent	Excellent	High	High	Moderate
Hybrid	High	High	Excellent	High	High	High	Moderate

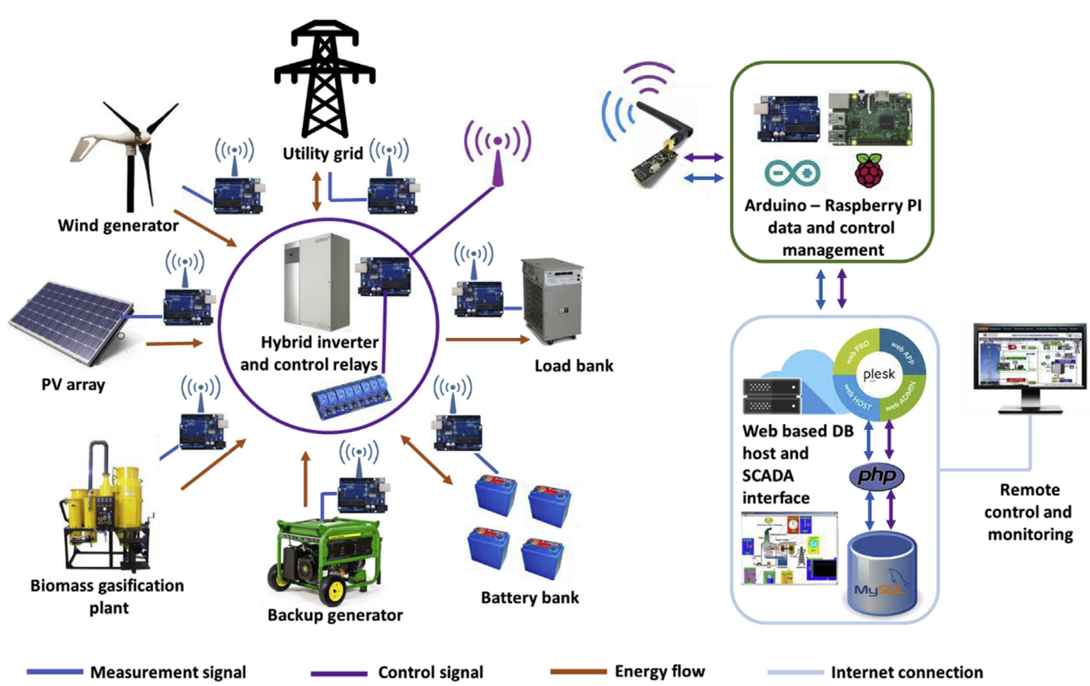

Fig. 1. Overall system architecture.

Table 2
LabDER-UPV microgrid main features.

Description	Main features
Photovoltaic array	2.1 kW, 12 solar panels. Connected to a Xantrex GT grid-tied inverter.
Wind power system	3.5 kW @ 12 m/s wind speed. Installed on a 24-meter tower from ground level.
Biomass power plant	10 kW @ 30 Nm3/h from syngas. 13 kg/h biomass consumption from wood chips and pellets.
Battery bank	12 kWh power capacity, four batteries from 12 V @ 250 Ah.
Fuel backup generator	Petrol 9 kW, 230 VAC @ 50 Hz PRAMAC S12000.
Test-load bank	10 kW, 240 VAC @ 50 Hz resistive load bank (30 × 330W resistors)

Table 3
Meteorological parameters measured in the microgrid understudy.

Environmental measurements	Units	Sensor	Measuring range
Solar irradiance	W/ m^2	CEBEK C0121 Solar cell	0–1100 W/m^2; ±40W/m^2
Environmental temperature	°C	DHT22	-40 to 80 °C; ±0.5 °C
Wind speed	m/s	FGHGF Anemometer 0–5V	0–32,4 m/s; ±1 m/s
Relative environmental humidity	%	DHT22	0–100%; ±5%

Table 4
Electrical parameters measured in the microgrid understudy.

Electrical measurements	Units	Sensor	Measuring range
Current	A	YHDC SCT-013-030	0–100 A; +-3%
Voltage	V	PCB Mount Transformer VB 2.3/2/12	200–260 V; +-1V

2. Design

The low-cost Web-based SCADA system was implemented in a microgrid at LabDER-UPV [26, 27] composed by a photovoltaic (PV) array, a small-power wind turbine, a biomass gasification plant, a battery-based energy storage system and a fuel backup generator. All the energy sources, as shown in Fig. 1, are connected to a hybrid inverter, which allows the microgrid to operate in both ways: stand-alone or grid-tied to feed a programmed load. Table 2 shows the main features of the microgrid.

The renewable and backup energy sources are connected to an AC bus managed by a Xantrex XW hybrid inverter, which communicates wirelessly to the Arduino-Raspberry PI base station for data acquisition and control signal management. Remote control and monitoring the microgrid is available through a Web host with a MySQL database. The Web hosting platform used is PLESK, which allows users to set up Websites and configure a Web server through a control panel with a simple, intuitive and easy-to-use interface. PLESK bases its programming language in PHP and MySQL, versions 7.1 and 5.5, respectively.

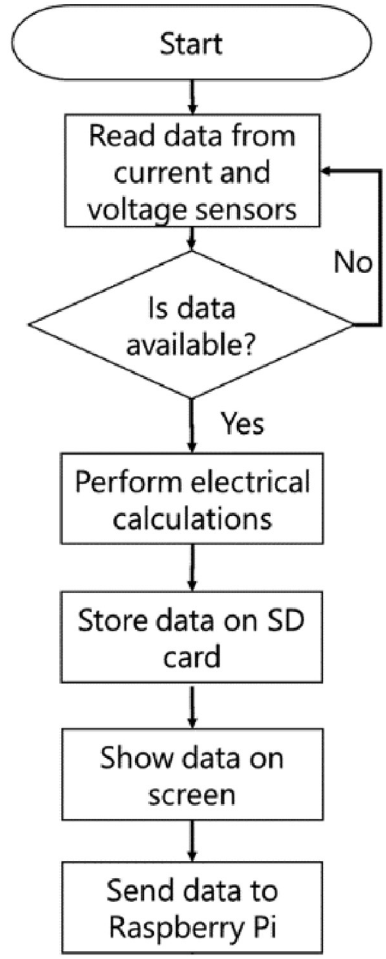

Fig. 3. Flowchart of the algorithm to store the measured and calculated variables.

3. Methodology

The implemented low-cost experimental microgrid platform has a functional cloud-based own-developed SCADA system, as previously shown in Fig. 1.

3.1. Design of measurement and control devices

To collect weather data and measure the electrical parameters of the microgrid, detailed at Tables 3 and 4, an Arduino wireless power meter

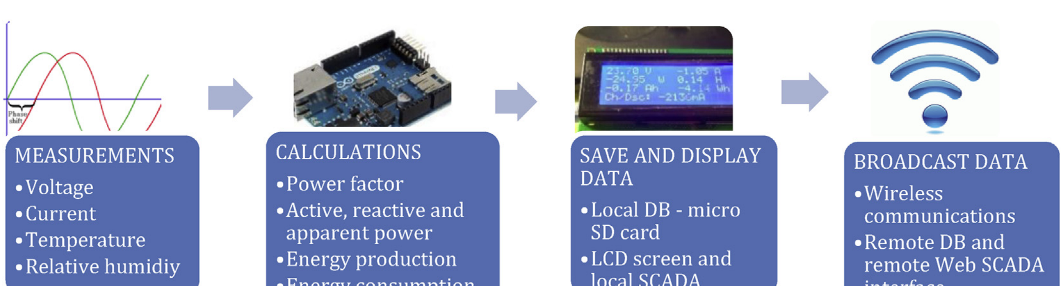

Fig. 2. Main tasks of the Arduino wireless power meter (AWPM).

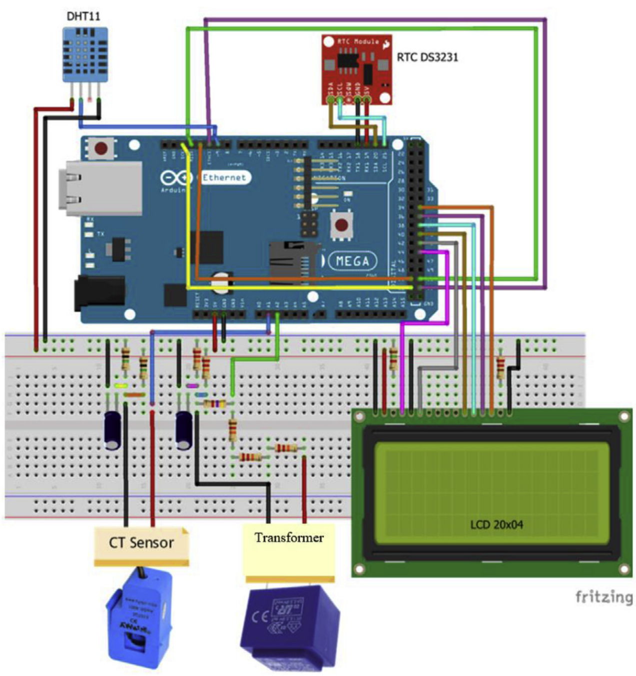

Fig. 4. AWPM main components connection diagram.

(AWPM) is designed and built. Other related parameters were calculated from those measurements by using the energy-monitoring library also mentioned in Tables 3 and 4. The device manufacturer has recommended these libraries, but some of them have been modified to obtain more information from the measuring devices, according to the work goal to create a single complete system. An integrated Arduino-based base station broadcasted these data via wireless, using the radio frequency transceiver module NRF24L01 through the SPI (Serial Peripheral Interface) synchronous protocol. The whole system is displayed in Fig. 2.

Fig. 3 shows a flowchart of the process to store the measured variables by using the AWPM. Calibration of the current and voltage measurements is an essential issue for the AWPM implementation; this task is carried out by adjusting calibration coefficients included in the code. These coefficients are deduced by the calibration of the AWPM with a commercial Sentron PAC3200 Power Meter up to reach a precision of ±5% on average.

The AWPM designed, operates with a voltage transformer and an SCT-013 non-intrusive current transformer sensor. The transformer ratio is 12:1, reducing de grid voltage to a safe level that can be adjusted by a voltage divider to operate at the Arduino analog input voltage level (0–5 VDC). The analog input A2 is used to measure the voltage. The current transformer SCT-013 measures the instantaneous current, and the Arduino reads the value through the analog input A1. Fig. 4 shows the connection of the AWPM main components.

The phase difference between voltage and current phase displacement is determined by using a zero-crossing detection algorithm, programmed in the power calculations library used for the AWPM. This algorithm is based on the work of [28, 29], and [30] where the interaction between continuous-time functions and the discrete event is modelled. Active, apparent, reactive power and power factor are calculated employing Eqs. (3), (4), (5), and (6) respectively, taken from [31] for AC circuits analysis.

$$V_{RMS} = \sqrt{\frac{1}{N}\sum_{n=0}^{N-1} V^2(n)} \tag{1}$$

$$I_{RMS} = \sqrt{\frac{1}{N}\sum_{n=0}^{N-1} I^2(n)} \tag{2}$$

$$P = \frac{1}{N}\sum_{n=0}^{N-1} V(n)\cdot I(n) \tag{3}$$

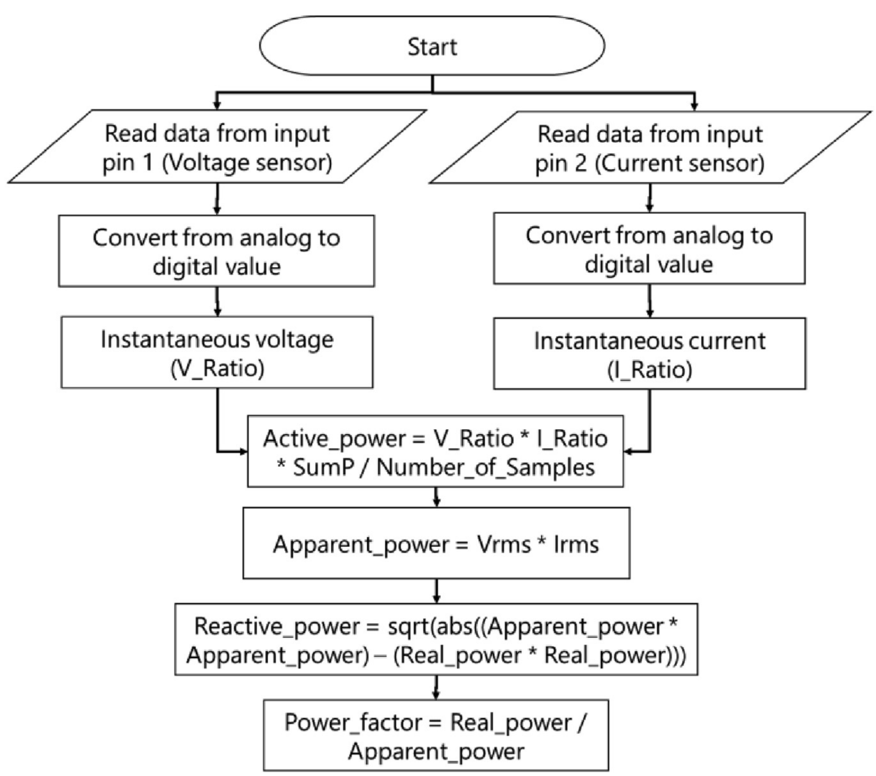

Fig. 5. Flowchart with the algorithm to deduce power factor, active, apparent and reactive power.

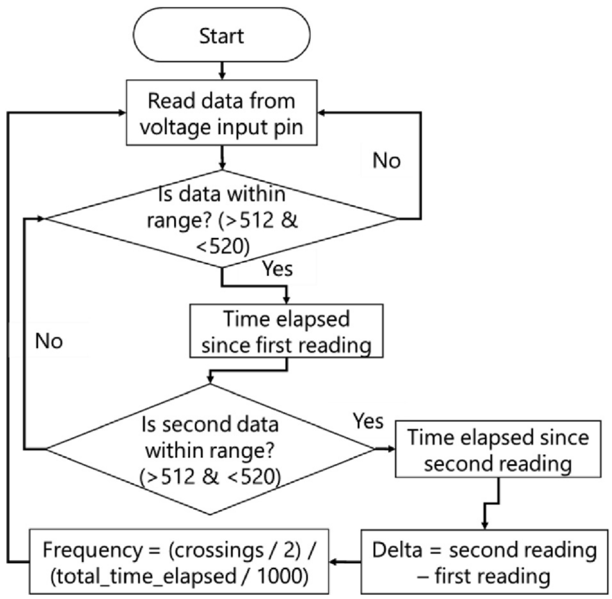

Fig. 6. Flowchart and code for grid frequency calculation.

$$S = V_{RMS} \cdot I_{RMS} \tag{4}$$

$$Q = \sqrt[2]{|S^2 - P^2|} \tag{5}$$

$$\cos\varphi = \frac{P}{S} \tag{6}$$

V_{RMS} is the voltage root mean square value (Eq. 1); I_{RMS} is the current intensity root mean square value (Eq. 2); P is the active power (Eq. 3); S is apparent power (Eq. 4); Q is reactive power (Eq. 5) and $\cos\varphi$ is the power factor (Eq. 6). Based on Eqs. (3), (4), (5), and (6), Fig. 5 shows the code dedicated to calculating the active, reactive and apparent power and the power factor.

The Emonlib. h library for Arduino performs the rest of the calculations. This library has been modified to measure frequency additionally. Fig. 6 shows the flowchart for the calculation of the grid frequency. After reading the analog pin reference and using the voltage transformer connected to the Arduino board, the algorithm detects every time an AC signal crosses zero. Given the 10-bit resolution range of the analog to digital converter (ADC), if a value is between 512 and 520 (intermediate range of the $2^{10} = 1024$ possible values) means that the signal has crossed the zero reference of the AC voltage.

Fig. 7a shows the general scheme of the Arduino wireless power meter (AWPM), and Fig. 7b displays the Arduino board responsible for the meteorological data collector (AWMDC) that compiles wind speed, solar radiation, environmental temperature and humidity with a DHT22 sensor. Again, a base station receives the data and broadcasts them via wireless communication, using the NRF24L01 Radiofrequency transceiver module.

The operation of the microgrid requires connecting and disconnecting the different renewable sources according to the preprogrammed load and the amount of energy coming from those sources. The Arduino wireless switch controller (AWSC) accomplishes this task. Fig. 7c shows its overall structure that an Arduino Mega board, a radiofrequency transceiver module NRF24L01 and a Relay module. Its operation depends on commands sent by the Arduino-Raspberry Pi 3 wireless base station (ARWBS), shown in Fig. 7d.

ARWBS is based on the integration of an Arduino board and a Raspberry Pi, allowing the system to log data and store it into a local DB, as well as in a cloud DB. The communications of the Arduino and Raspberry Pi 3 use the I2C serial protocol. With this structure, the Arduino is acting as an interface between the Raspberry Pi and all the other Arduino-based data collectors. A SCADA system Website user interface has been developed to control and monitor the entire microgrid.

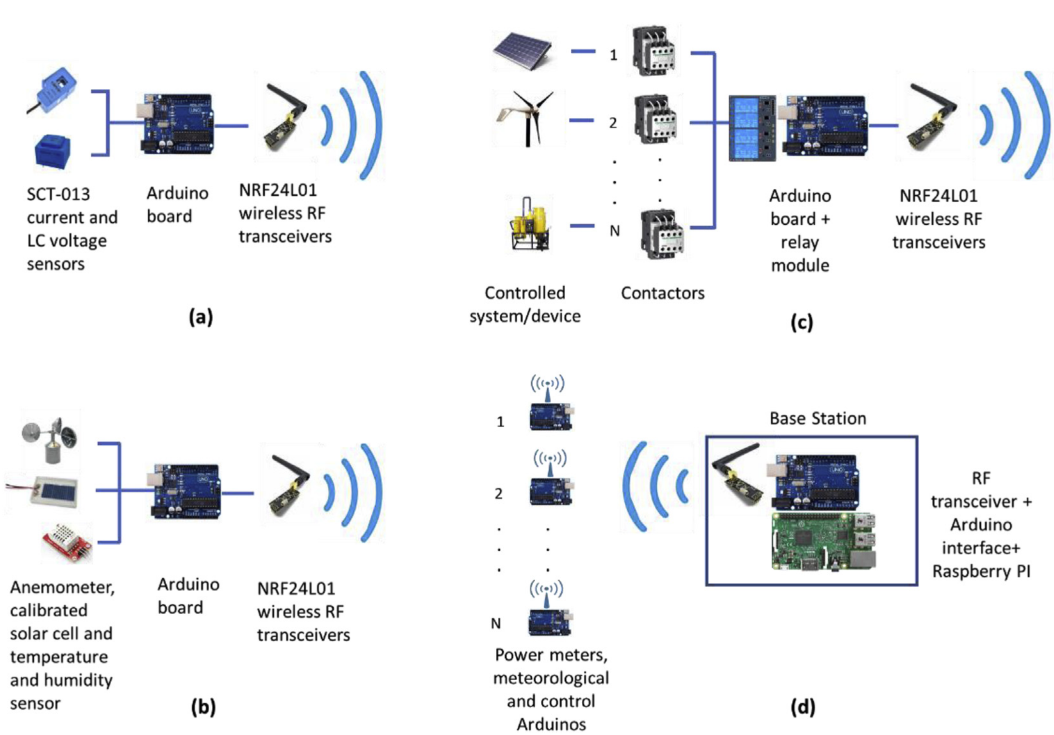

Fig. 7. Scheme of the Arduino wireless devises (a) Arduino wireless power meter (AWPM); (b) Arduino wireless meteorological data collector AWMDC); (c) Arduino wireless switch controller (AWSC); (d) Arduino-Raspberry PI 3 wireless base station (ARWB).

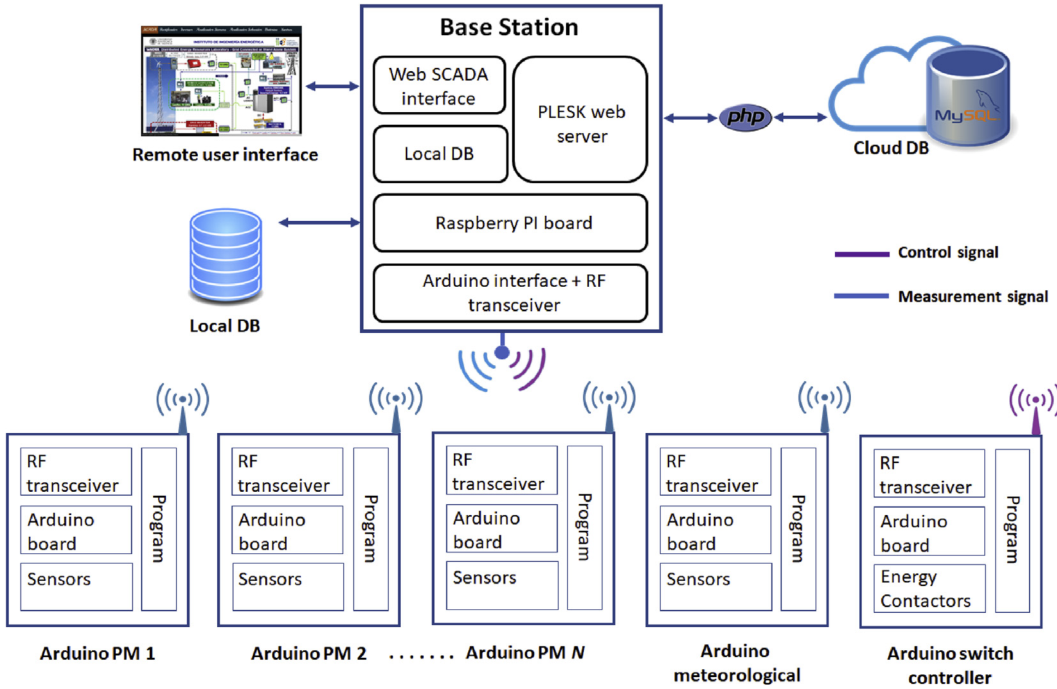

Fig. 8. Communications structure.

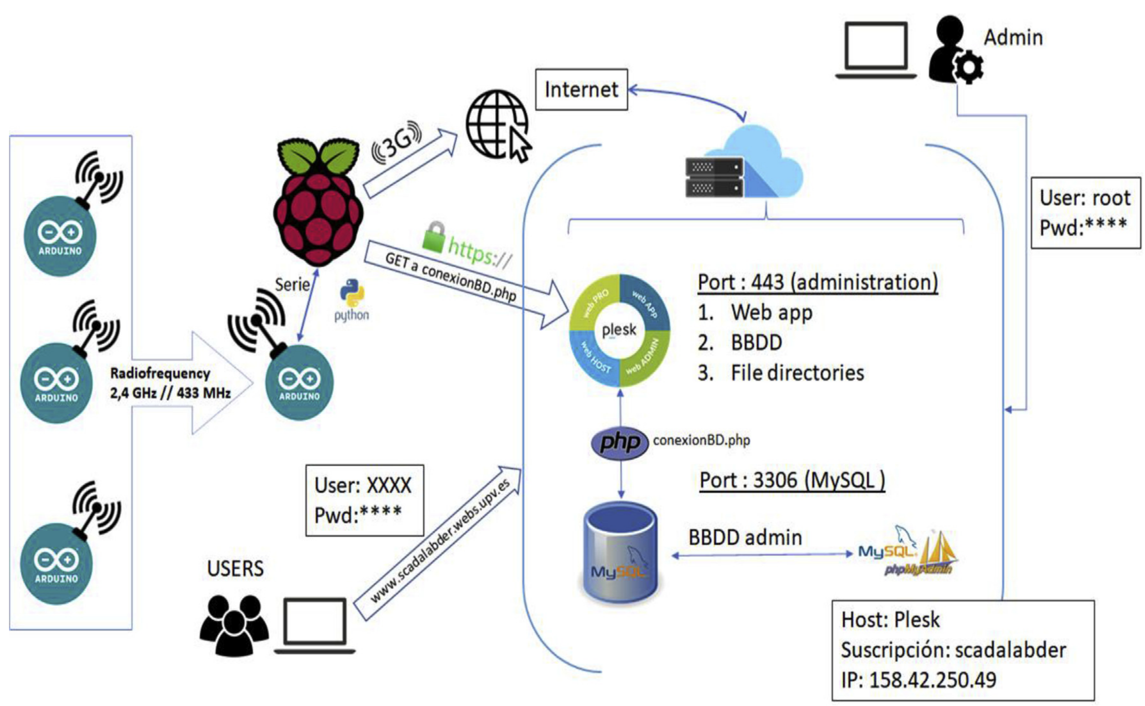

Fig. 9. Connection process with the MySQL DB.

This interface is linked to the ARWBS by TCP/IP communications and hosted in PLESK Web using a MySQL DB.

3.2. Communications and data logging

An NRF24L01 Radiofrequency transceiver, operating at 2.4 GHz and using an SPI protocol to manage the communication with an Arduino board, carries out the wireless communications. AWPMs, AWMDC, and AWSC are wirelessly linked to the ARWBS by using radiofrequency, as shown in Fig. 8. The users interact with the microgrid Web SCADA interface by using HTML5, JavaScript and PHP programming languages, hosted on the PLESK Web server. Cloud DB reads and writes data, updating continuously at specific sample rates, for each variable following the data refresh and operation commands sent by the ARWBS to each microgrid device. Besides, a local DB records the information as a backup for preventing data losses due to wireless communication or internet connection failures.

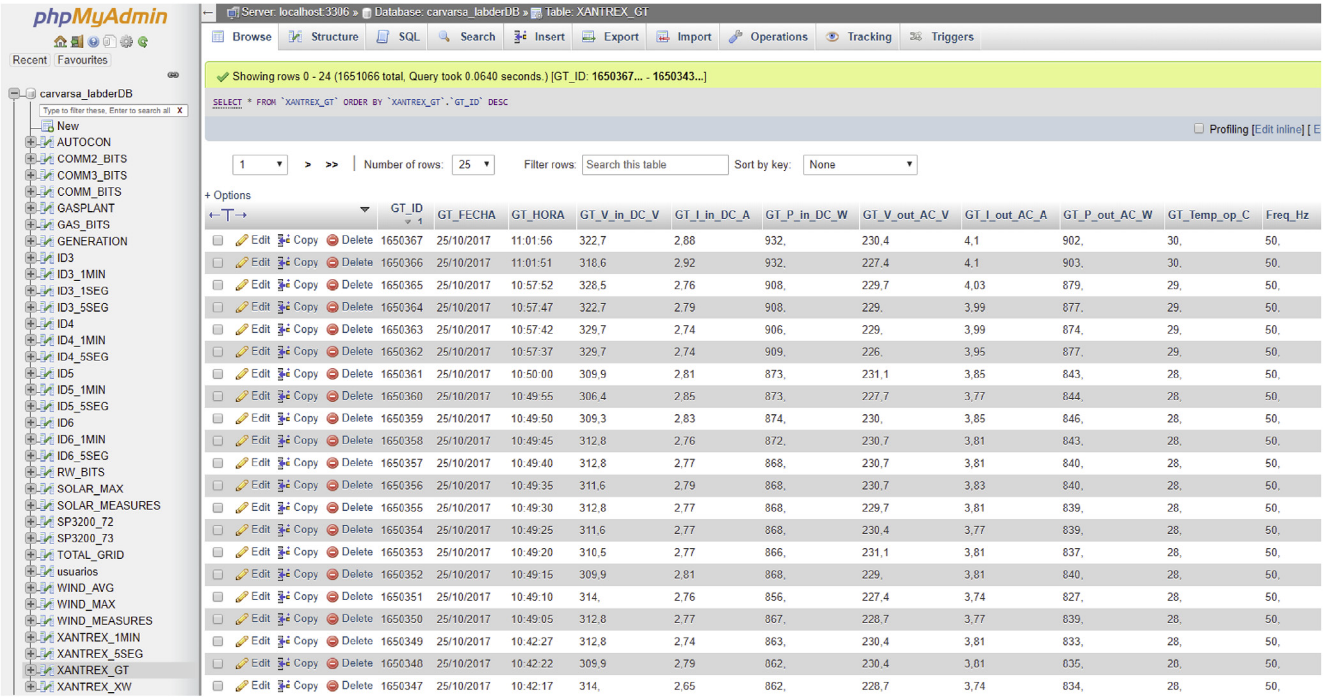

Fig. 10. Information obtained from the sensors and stored in the DB.

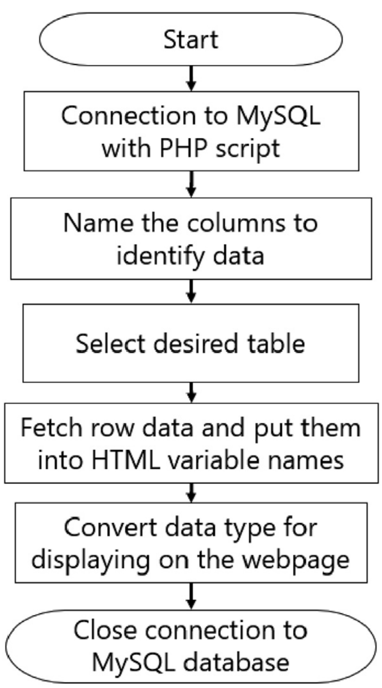

Fig. 11. Block diagram of MySQL query processing and PHP request to fetch data from the MySQL DB.

3.3. Cloud DB and web SCADA

Users interact with the microgrid using a Web SCADA interface that operates over a MySQL cloud DB, writing and consulting data using an own-developed Graphical User Interface (GUI). PHP makes queries from Web SCADA GUI through the PLESK platform and, subsequently, modifies the MySQL cloud DB, updating data, according to user commands or the automated data refresh option. Fig. 9 shows the general data transmission process.

Once communication is established between Plesk web server and MySQL DB, the next step is to generate a table, adequately fetched, from AWPM in a real-time monitoring process with data on power, meteorological measurements and status of the microgrid components. The communication between the DB and the raspberry pi is carried out using a 3G modem using a SIM card, but it is also possible to use a wifi network. As time progresses, more data will be recorded, increasing the size of the table and, therefore, the space memory could reach the maximum storage permitted by a Plesk regular account (6 GB). Moreover, it will require more data if the sampling rate is very high, in the order of 1 s for all variables. Annual maintenance is programmed; it includes a back up of the stored data and memory clean up, avoiding to overcome the maximum storage permitted. Fig. 10 represents the information obtained from the grid-tied PV inverter. Such data are stored in the MySQL DB.

Data is collected every second and the Web SCADA fetch data from MySQL DB, written in SQL queries with PHP acting as a link of the remote web interface and DB server. The PHP query sentences aim to get access

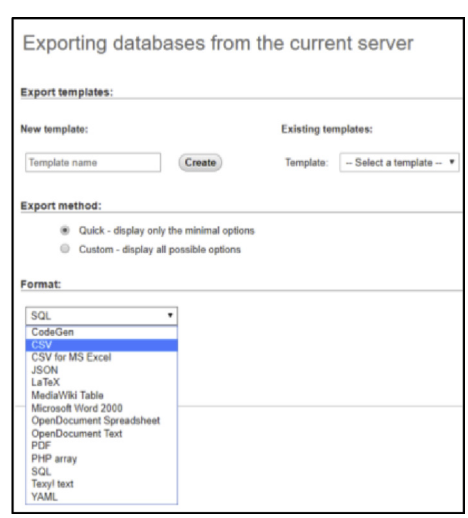

Fig. 12. Formats available in PLESK to export data from the MySQL DB.

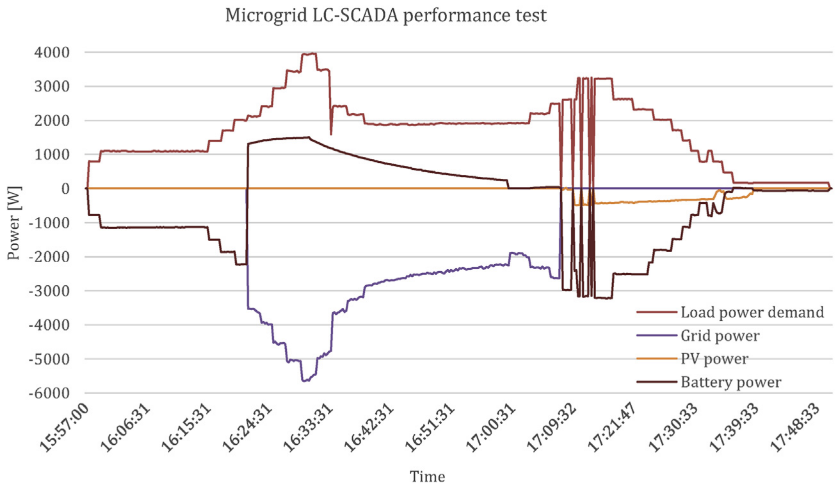

Fig. 13. LC-SCADA system data acquisition test over the microgrid.

to voltage, current, power, and energy data located in their respective column of the table. Fig. 11 shows the flowchart for the code to read and update data register from the DB.

As previously mentioned, the user operates the microgrid through an own-developed Web SCADA interface that makes PHP queries to the PLESK server via port 443 and to the cloud DB using port 3306 through the ARWBS using TCP/IP. There are three different types of registered tables within the cloud DB: measured data, microgrid operation conditions and user credentials. The information displayed on the Web interface is timely refreshed according to the previous query requests, and, when the load or any energy sources is changed, the user or the preset programme sends a request to cloud DB utilising the Web SCADA. Such request is read by the ARWBS that sends the corresponding request to the AWSC to close or open the physical switches or relays.

All the devices connect to the remote PLESK server interact with the system employing an HTML5 graphical user interface that allows the user to set up operation parameters for the microgrid and monitoring and supervising data. Data available in the PLESK server can be exported in other formats, as shown in Fig. 12.

4. Results

Experiments were carried out in the Laboratory of Renewable Energies (LabDER) at Universitat Politècnica de València. Such experiments allowed to test the functionality and performance of the low-cost SCADA system, Fig. 1 displays the microgrid components, together with its energy, data and control signals flows. The figure also includes the storage of information in a remote database and the access to a remote monitoring and control graphical interface developed in HTML5 and JavaScript over an internet connection. Once all the components of the monitoring Arduino-based devices implementation, the software and the database development have been installed, a SCADA system was ready to be used in the microgrid. Fig. 13 shows some of the recorded data obtained in a microgrid experiment carried out in June 2018.

During the experiment, a load energy demand went from 800 W to a maximum of 3.97 kW, and it was covered with contributions from PV array, the utility grid and the battery bank. Details are presented in Fig. 13, where positive values correspond to the energy demanded by the system, and negative ones are those supplied by the energy sources. It should be noticed that the battery bank could work reversibly as a load or an energy source, as usual in energy storage systems in microgrid applications. PV power fluctuations are due to cloud appearance, as it was detected by the meteorological data gathered by the AWMDC, that also indicates that the maximum solar irradiance during this short test was 600 W/m^2 at 35 °C on the surface of solar panels (Fig. 14).

Medium and long-term tests in the microgrid using the SCADA system were also addressed. Fig. 15 shows the results for the medium term, one-day duration test. Power from the wind turbine and the PV array, accordingly to the current meteorological conditions during the day, and the power demanded from the microgrid are displayed. Power demand

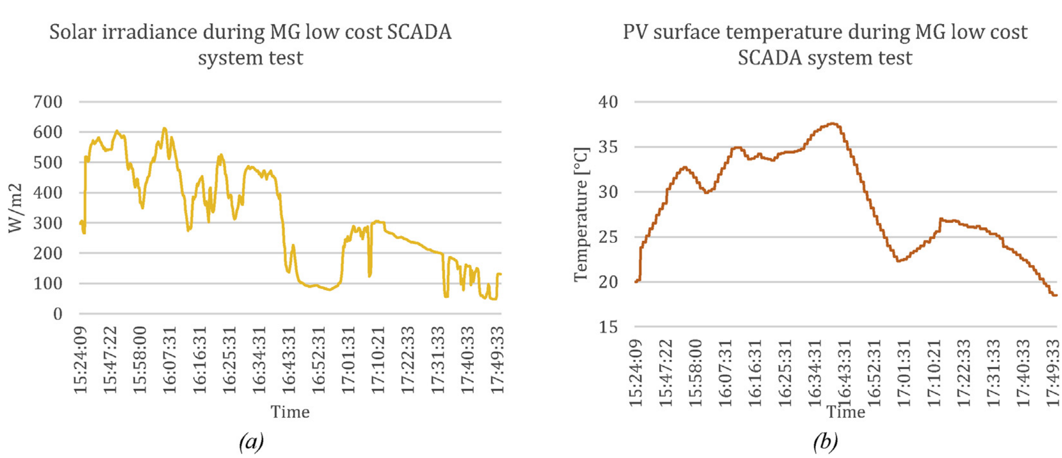

Fig. 14. (a) Solar irradiance (b) PV surface temperature data collecting for a short-term test of the LC-SCADA system.

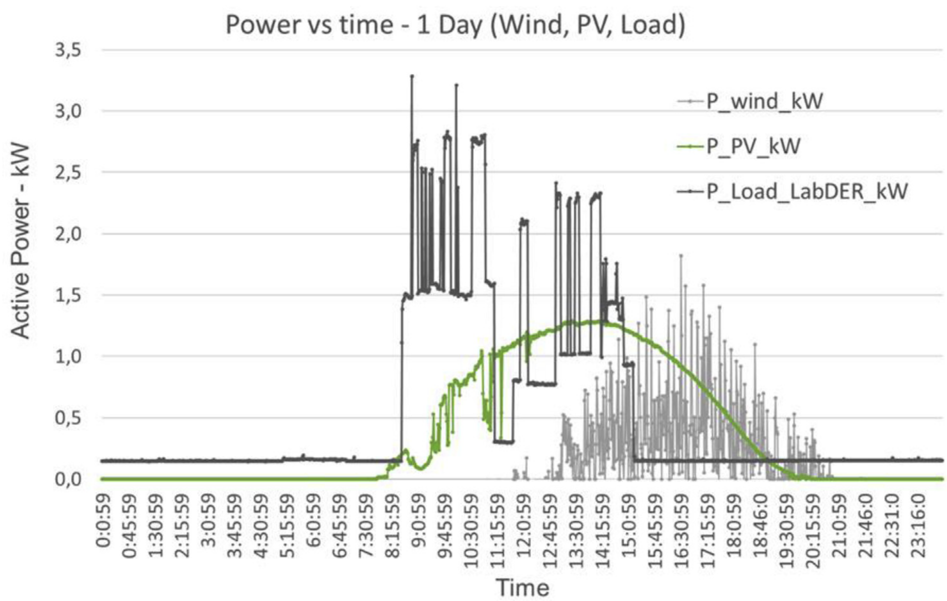

Fig. 15. One-day test - LC-SCADA system.

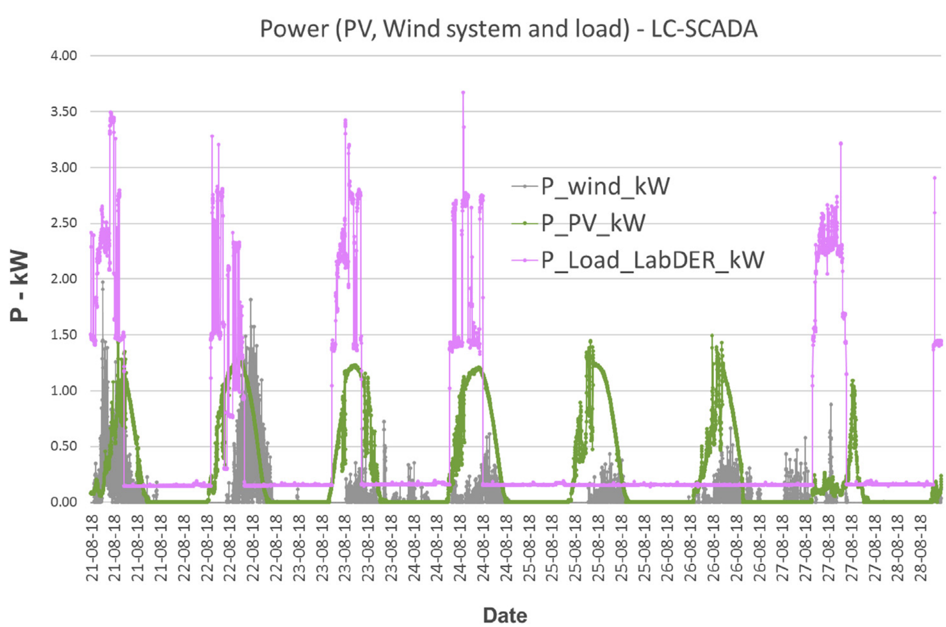

Fig. 16. One-week test - LC-SCADA system.

Table 5
The average standard deviation between AWPM and Sentron PAC3200 power meter measurements.

Variable	Average standard deviation (grid-tied test)	Average standard deviation (stand-alone test)
Active power	2.720	2.268
AC bus Voltage	1.734	1.121
AC bus Intensity	0.068	0.107
AC bus Frequency	0.058	0.079

ranges on average from 0.3 to 3.3 kW, and it includes the use of lights, personal computers from researchers and the low-cost SCADA system-related devices.

The long-term test covers an entire week. Fig. 16 shows the obtained data for wind power, solar power, and users power demand. The reliability of the system is enough for obtaining the data required for analysing the system.

Table 5 shows the average standard deviation deduced from the difference between values obtained with the AWPM and a SIEMENS SENTRON PAC3200. For a set of 6,506 measurements, the AWPM

Table 6
LC-SCADA system implementation cost.

Device	Qty	Unit Cost	Total cost
Arduino based wireless single-phase Power Meter. Contains 1 Arduino UNO, 1 ethernet shield, 1 NRF24L01 transceiver, 1 SCT-013 current transformer, 1 VB 2.3/2/12 voltage isolated transformer, and miscellaneous accessories.	9	100 € [1]	900 €
Arduino - Raspberry PI base station. Contains: 1 Arduino UNO, 1 NRF24L01 transceiver, 1 Raspberry Pi and miscellaneous accessories.	1	80 € [1]	80 €
Arduino based wireless meteorological module. Contains 1 Arduino UNO, 1 NRF24L01 transceiver, 1 DHT temperature and humidity sensor, 1 solar irradiance sensor, 1 anemometer analog input reading.	1	50 € [1]	50 €
Arduino based wireless switching module. 1 Arduino UNO, 1 NRF24L01 transceiver, 1 8-channel relay output relay.	1	25 € [1]	25 €
PLESK Web server annual fee	1	60 € [2]	60 €
Web SCADA system interface	1	Free	Free
Other components	1	200 € [1]	200 €
TOTAL			1,315 €

[1] Unitary cost prices as listed in Amazon. es Website.
[2] Unitary cost prices as listed in Plesk.com Website.

Table 7
Standard brandmark SCADA system implementation cost.

Device	Qty	Unit Cost	Total cost
SENTRON PAC 3200 Power Meter	9	600 € [1]	5,500 €
Meteorological data logger	1	200 € [1]	200 €
OMRON Programmable Logic Controller CJ1M with serial communication CJ1WSCU31, ethernet communication CJ1WETN21, relay output CJ1WOC211 and power source CJ1W-PA205R modules.	1	2,200 € [2]	2,200 €
PLESK Web server annual fee	1	60 € [3]	60 €
CX-Supervisor SCADA system	1	1,500 € [2]	1,500 €
Other components	1	200 € [4]	200 €
Total			**9,360 €**

[1] Unitary cost prices as listed in PCE-instruments.com Website.
[2] Unitary cost prices as listed in Mouser. es Website.
[3] Unitary cost prices as listed in Plesk.com Website.
[4] Unitary cost prices as listed in Amazon. es Website.

measurement performance test was carried for two different cases: grid-tied and off-grid operation mode of the microgrid.

The highest deviation occurs in the active power measurements. It is also noticeable how the microgrid has a better bus frequency performance when it is working in the grid-tied mode because when the microgrid operates in the off-grid mode, the frequency depends on the backup generator or the gasification plant, in which it is more complicated to have a stable frequency.

All these comparisons between the own-developed Arduino based power meter and a commercial Sentron PAC 3200 denote the AWPM could be a reliable solution for low-cost data acquisition systems.

Analysis of cost can be deduced from Table 6, that shows the price in Euros for the low-cost SCADA components according to the Amazon. es and Plesk.com Websites. Using these values, the total implementation cost for the low-cost SCADA system is 1.315 €. Meanwhile, a similar solution implemented with brandmark devices for a standard SCADA system solution would cost around 9,360 €. The prices of the main components, detailed in Table 7, are available in PCE-instruments.com, Mouser. es, Plesk.com, and Amazon. es Web sites. It took six months to develop the low-cost SCADA, while the brandmark system could be implemented in about three months.

5. Conclusions

Experiments prove that the proposed low-cost SCADA could monitor an experimental microgrid successfully. Several tests were carried out to validate the system robustness, data collecting and device communication efficiency with a remote DB. As a result, it was obtained a successful validation of the system operability. Measuring instruments were calibrated, reducing the error between the developed AWPM and a commercial Sentron PAC 3200 power meter, and the deviations were acceptable (Table 5). The system can be supervised both via a local computer or via the Web SCADA interface linked to a remote database. The implementation of the developed system microgrid has a low cost, in the order of 85% cheaper when compared with standard SCADA systems, and has high versatility, allowing for the addition of new functions and devices. Finally, this system could be considered for other types of applications.

Future projects related to open-source software should focus on the integration of new technologies, such as IoT (Sensors, devices and appliances connected to the internet), blockchain (for doing data transactions securely) and big data (Analysis of large volume of data efficiently) in Arduino and Raspberry projects.

Declarations

Author contribution statement

Carlos Vargas-Salgado: Conceived and designed the experiments; Contributed reagents, materials, analysis tools or data; Wrote the paper.

Cristian D. Chinas: Performed the experiments; Wrote the paper.

Jesus Aguila-Leon: Performed the experiments; Analyzed and interpreted the data.

Elias Hurtado: Analyzed and interpreted the data; Contributed reagents, materials, analysis tools or data.

Funding statement

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Competing interest statement

The authors declare no conflict of interest.

Additional information

No additional information is available for this paper.

References

[1] S. Lan, Research on application mode of wireless and carrier dual-mode communication in regional microgrid, in: 2018 IEEE 3rd International Conference on Cloud Computing and Big Data Analysis (ICCCBDA), 2018, pp. 486–489.
[2] D. Moga, D. Petreus, N. Stroia, Web based solution for remote monitoring of an islanded microgrid, in: IECON 2016 - 42nd Annual Conference of the IEEE Industrial Electronics Society, 2016, pp. 4258–4262.
[3] M. Baranwal, A. Askarian, S. Salapaka, M. Salapaka, A distributed architecture for robust and optimal control of DC microgrids, IEEE Trans. Ind. Electron. 66 (4) (Apr. 2019) 3082–3092.
[4] L. Mariam, M. Basu, M.F. Conlon, Microgrid: architecture, policy and future trends, Renew. Sustain. Energy Rev. 64 (2016) 477–489.
[5] O.V. Gnana Swathika, K. Karthikeyan, S. Hemamalini, R. Balakrishnan, PLC based LV-DG synchronization in real-time microgrid network, ARPN J. Eng. Appl. Sci. 11 (5) (2016) 3193–3197.
[6] A. Merabet, K. Tawfique Ahmed, H. Ibrahim, R. Beguenane, A.M.Y.M. Ghias, Energy management and control system for laboratory scale microgrid based wind-PV-battery, IEEE Trans. Sustain. Energy 8 (1) (2017) 145–154.

[7] J. Zhuang, G. Shen, J. Yu, T. Xiang, X. Wang, The design and implementation of intelligent microgrid monitoring system based on WEB, Procedia Comput. Sci. 107 (Jan. 2017) 4–8.
[8] S. Sujeeth, O.V. Gnana Swathika, IoT based automated protection and control of DC microgrids, in: Proc. 2nd Int. Conf. Inven. Syst. Control. ICISC 2018, No. Icisc, 2018, pp. 1422–1426.
[9] Y. Lopes, N.C. Fernandes, K. Obraczka, "Smart grid communication: Requirements and SCADA protocols analysis," in *2018 Simposio Brasileiro de Sistemas Eletricos*, SBSE), 2018, pp. 1–6.
[10] Q. Qassim, et al., A survey of SCADA testbed implementation approaches, Indian J. Sci. Technol. 10 (26) (2017) 1–8.
[11] H. Bentarzi, M. Tsebia, A. Abdelmoumene, PMU based SCADA enhancement in smart power grid, in: 2018 IEEE 12th International Conference on Compatibility, Power Electronics and Power Engineering (CPE-POWERENG 2018), 2018, pp. 1–6.
[12] K. Candelario, C. Booth, A. St. Leger, S.J. Matthews, Investigating a Raspberry Pi cluster for detecting anomalies in the smart grid, in: 2017 IEEE MIT Undergraduate Research Technology Conference (URTC), 2017, pp. 1–4.
[13] D. Peharda, I. Ivanković, N. Jaman, Using data from SCADA for centralized transformer monitoring applications, Procedia Eng. 202 (Jan. 2017) 65–75.
[14] J. Dai, W. Yang, J. Cao, D. Liu, X. Long, Ageing assessment of a wind turbine over time by interpreting wind farm SCADA data, Renew. Energy 116 (Feb. 2018) 199–208.
[15] S.A.S. Obayes, I.R.K. Al-Saedi, F.M. Mohammed, Prototype wireless controller system based on raspberry pi and arduino for engraving machine, in: 2017 UKSim-AMSS 19th International Conference on Computer Modelling & Simulation (UKSim), 2017, pp. 69–74.
[16] R.Q. Cetina, A.J. Roscoe, A.C. Atoche, Low-cost power systems metrology laboratory based on raspberry Pi, in: 2018 First International Colloquium on Smart Grid Metrology (SmaGriMet), 2018, pp. 1–2.
[17] M. Poongothai, P.M. Subramanian, A. Rajeswari, Design and implementation of IoT based smart laboratory, in: 2018 5th International Conference on Industrial Engineering and Applications (ICIEA), 2018, pp. 169–173.
[18] D. Watson, T. Chakraborty, M. Rodgers, The need for SCADA communication in a Wind R&D Park, Sustain. Energy Technol. Assessments 11 (Sep. 2015) 65–70.
[19] S.M. Patil, M. Vijayalashmi, R. Tapaskar, IoT based solar energy monitoring system, in: 2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS), 2017, pp. 1574–1579.
[20] C.-S. Choi, J.-D. Jeong, I.-W. Lee, W.-K. Park, LoRa based renewable energy monitoring system with open IoT platform, in: 2018 International Conference on Electronics, Information, and Communication (ICEIC), 2018, pp. 1–2.
[21] V.H. Nguyen, Q.T. Tran, Y. Besanger, SCADA as a service approach for interoperability of micro-grid platforms, Sustain. Energy, Grids Networks 8 (Dec. 2016) 26–36.
[22] A.D. Deshmukh, U.B. Shinde, A low cost environment monitoring system using raspberry Pi and arduino with Zigbee, in: 2016 International Conference on Inventive Computation Technologies (ICICT), 2016, pp. 1–6.
[23] S. Ferdoush, X. Li, Wireless sensor network system design using raspberry pi and arduino for environmental monitoring applications, Procedia Comput. Sci. 34 (Jan. 2014) 103–110.
[24] I. Allafi, T. Iqbal, Low-cost SCADA system using arduino and reliance SCADA for a stand-alone photovoltaic system, J. Sol. Energy 2018 (2018) 1–8.
[25] J.L. Sarinda, T. Iqbal, G. Mann, Low-cost and open source SCADA options for remote control and monitoring of inverters, in: 2017 IEEE 30th Canadian Conference on Electrical and Computer Engineering 1–4, 2017.
[26] E. Hurtado, E. Peñalvo-López, A. Pérez-Navarro, C. Vargas, D. Alfonso, Optimization of a hybrid renewable system for high feasibility application in non-connected zones, Appl. Energy 155 (2015).
[27] A. Pérez-Navarro, et al., Experimental verification of hybrid renewable systems as feasible energy sources, Renew. Energy 86 (2016).
[28] F.E. Cellier, Combined continuous/discrete simulation: applications, techniques and tools, in: Proceedings of the 18th Conference on Winter Simulation, 1986, pp. 24–33.
[29] L.F. Shampine, I. Gladwell, R.W. Brankin, Reliable solution of special event location problems for ODEs, ACM Trans. Math Software 17 (1) (1991) 11–25.
[30] T. Park, P.I. Barton, State event location in differential-algebraic models, ACM Trans. Model Comput. Simulat 6 (2) (1996) 137–165.
[31] R.L. Boylestad, Introductory Circuits Analysis, thirteenth ed., Pearson, 2016.