HardwareX 1 (2017) 54–67

Contents lists available at ScienceDirect

HardwareX

journal homepage: www.elsevier.com/locate/ohx

Hardware Article

Open source IoT meter devices for smart and energy-efficient school buildings

Lidia Pocero [a], Dimitrios Amaxilatis [a], Georgios Mylonas [a,*], Ioannis Chatzigiannakis [b]

[a] Computer Technology Institute and Press "Diophantus", Patras, Greece
[b] Universita Di Roma "La Sapienza", Rome, Italy

ARTICLE INFO

Article history:
Received 3 December 2016
Accepted 14 February 2017

Keywords:
Open source hardware and software
Smart school building
Energy efficiency
IoT deployment
Validation
Environmental monitoring

ABSTRACT

One oft-cited strategy towards sustainability is improving energy efficiency inside public buildings. In this context, the educational buildings sector presents a very interesting and important case for the monitoring and management of buildings, since it addresses both energy and educational issues. In this work, we present and discuss the hardware IoT infrastructure substrate that provides real-time monitoring in multiple school buildings. We believe that such a system needs to follow an open design approach: rely on hardware-agnostic components that communicate over well-defined open interfaces. We present in detail the design of our hardware components, while also providing insights to the overall system design and a first set of results on their operation. The presented hardware components are utilized as the core hardware devices for GAIA, an EU research project aimed at the educational community. As our system has been deployed and tested in several public school buildings in Greece, we also report on its validation.

1. Introduction

One immediate step towards cutting down on carbon emissions is to improve the energy efficiency of buildings. Till now the dominant approach was to use energy-efficient infrastructure and materials to reduce energy consumption. An alternative approach, that has recently received emphasis, is the promotion of energy consumption awareness, sustainability and behavioral change on people living or working inside buildings. Both approaches have their merits; however, in many cases mainly due to implementation costs, the second option is more viable.

A key challenge for achieving sustainability and transforming people's behavior towards energy consumption is the need to educate them in such issues. An interesting starting point is the educational sector. Educational buildings constitute the 17% of the non-residential building stock (in m^2) in the EU [1]. Clearly, any reduction on the energy consumption within these buildings will have significant impact on achieving our environmental goals. Yet, a more important aspect is that by reinforcing the educational community on educating the new generations we create a multiplier on the overall energy reductions: promoting sustainable behaviors at school will also reflect behaviors at home.

The educational sector presents many technical challenges, since national educational systems have thousands of buildings spread throughout the area of a country each of which having very different characteristics (in terms of e.g., construction, age, size, design, etc.). These difficulties are also complemented by the diverse teaching strategies followed

* Corresponding author.
E-mail addresses: pocero@cti.gr (L. Pocero), amaxilat@cti.gr (D. Amaxilatis), mylonasg@cti.gr (G. Mylonas), ichatz@dis.uniroma1.it (I. Chatzigiannakis).

http://dx.doi.org/10.1016/j.ohx.2017.02.002
2468-0672/

across the different grades. Still, the most challenging aspect is that financial expenditure for the educational sector is constantly being reduced across all European countries [2]. Therefore any changes on the infrastructure or the implementation of new curriculum (or extra curriculum) activities need to have minimum budget expenses.

For all these reasons we believe that developing an open source platform that aims to provide a flexible and replicable foundation for solutions in this context is a very important starting point. We here present the basic building elements for delivering an IoT-based infrastructure for monitoring energy consumption and environmental parameters, fitted for the requirements of school buildings, available as open source hardware and software. We utilize open standards and components to maximize flexibility and manageability so that our platform can address diverse educational environments, while at the same time leave room for customization and adaptability to fit the purposes of different educational needs. Our platform is utilized within the context of the GAIA [3] (Green Awareness In Action) EU research program, which aims to promote sustainability aspects in educational communities and is already deployed our system in 12 Greek public school buildings.

Regarding the structure of this work, in Section 2 we discuss the requirements and goals for the system developed, while in Section 3 we present some aspects of the overall design. We continue with details on the IoT nodes utilized in school buildings, while Section 5 provides details on the Bill of Materials and cost-related aspects. A set of results along with certain insights from real-world deployments is provided in Section 6, while we conclude this work in Section 7 with a short discussion on the outcomes of our work.

2. Requirements and goals

We strongly believe that a platform suitable for the educational sector needs to address the following requirements:

- Create a real-time platform to collect environmental and energy consumption data from school buildings.
- Monitor human comfort parameters in school buildings and the outdoor environmental state. The sensor network should spread out all over the building incorporating various kind of sensor.
- Build a distributed sensor network, using low-cost devices that can provide the whole building's environmental reality compared to a single, expensive and more accurate device located at a specific point.
- Such an IoT infrastructure is deployed in phases. In each phase the IoT infrastructure is extended to cover additional buildings or new type of sensors are introduced, i.e., it will need to be regularly maintained, improved and extended.
- Interoperability with available open source hardware modules.

Our reference implementation is based on the principles of *open-source hardware and software*. We believe that the use of physical technology artifacts *designed and offered openly* is of great value in an environment that promotes the exchange of knowledge, like schools or universities.

- It is crucial to follow an open standards approach: use open source software, protocols and hardware components, in order to maximize the adaptability and reusability of the system.
- This approach gives full control of the installation to the owner of the buildings and prevents any vendor-lock situations, since the operation and maintenance of the system can be assigned to any IT company or handled internally by the Ministry of Education's IT department.

Additionally, using a simple and open-source platform allows educators to use hardware designs and device software to create educational material for students, a movement that is highly favored by major institutions [4]. It also adds to the dimension of reproducing our implementation and results, since all hardware components utilized in this work can be easily acquired through electronics markets and be assembled as DIY projects even by the students themselves.

3. System design

The overall architecture of the system is built upon multiple heterogeneous installations in school buildings. Each installation consists of a multitude of IoT nodes, which communicate with a cloud infrastructure via an IoT gateway device. Depending on the installation, each school room includes one or more IoT node. IoT nodes include multiple sensing devices, while the gateway nodes coordinate communication and enable interaction with cloud-based services and other Internet-connected devices. All devices in our system use a wireless radio interface to communicate locally and fixed wired interfaces to report measurements to the cloud services of the system and receive commands or configuration updates.

The overall design pattern for the installation of IoT infrastructure in the school buildings participating in the project, is as follows:

- *Power Consumption Meter* IoT nodes installed to monitor the power consumption of the building as a whole, or specific floors/sectors (depending on the peculiarities of each building).
- *Environmental Comfort Meter* IoT nodes installed in classrooms and other supporting rooms to monitor a set of environmental parameters such as temperature, humidity, activity, and noise levels (see Fig. 6).

- A *set of IoT gateway nodes* installed in central points of the building to bridge the *IoT nodes* that communicate using IEEE 802.15.4 with the Internet.

To enable communication between the IoT nodes we use IEEE 802.15.4 provided by XBee devices connected to each IoT node using the Arduino XBee [5] and XbeeRadio [6] software libraries. All IoT nodes form an ad hoc networks and report their measurements through the designated IoT gateways. It is reasonable to expect that in some installations multiple gateways will be required due to various parameters of the buildings (e.g., outdoor vs indoor deployments, nodes spanning across multiple floors of the building, or buildings with too thick walls for the IEEE 802.15.4 signal to penetrate). For this, the ad hoc network needs to adapt to the real conditions and allow IoT nodes to connect to the gateway that provides the best communication quality (e.g., in terms of reliability and stability).

As a result, all indoor IoT nodes form an ad hoc overlaying multi-hop bidirectional tree network, and maintain it throughout the lifetime of the network. The gateway node is the root of this tree and the orchestrator of the network. Nodes that are not part of the tree structure can join at any time either directly below the gateway, or as a child under the node that communicates with the stronger signal. The resulting routing tree allows for bidirectional communication between the IoT nodes and the gateway. The routing library developed for the Arduino and XBee devices is also available on GitHub [7].

4. IoT nodes

Our design builds upon the open-source (hardware and software) Arduino platform, mainly for two reasons. First, as it is a widely used electronics prototyping platform, thus making it easy to maintain and extend. Second, because there exist many variations in the market thus making it a low-cost solution. In this section we present our design of a power consumption meter and an environmental comfort meter suitable for educational buildings.

4.1. Power consumption meter

The *power consumption meter* measures the apparent power in a 3-phase electrical installation and calculates its true power consumption remotely. The meter is designed to be installed in the general electrical distribution panel of the building and the measured data are transmitted wirelessly using an IEEE 802.15.4 network to an IoT gateway. Regarding the electrical setup, 3-phase electric power installations are a common practice for most public and private non-housing buildings, such as schools. Three separate single-phase supplies, with a fourth neutral connection, provide a constant voltage to power most common single-phase appliances. In order to measure the total power consumption of such an installation, it is necessary to independently measure the power consumption of each phase and add up the total consumption, as if the installation consisted of three separate lines.

4.1.1. Power measurement theory

Apparent power is the voltage of an *AC* system multiplied by all the current that flows into it. It can be computed as the product of RMS voltage and RMS current, as shown in Eq. (1) and is expressed in units of volt-amperes (VA). I_{RMS} and V_{RMS} values are calculated using Eqs. (2) and (3) where N is the number of samples and $i(n)$ and $u(n)$ are the samples of the electrical current and voltage signals.

$$Apparent\ Power = I_{RMS} \times V_{RMS} \tag{1}$$

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

$$V_{RMS} = \sqrt{\sum_{n=0}^{N-1} u(n)^2 / N} \tag{3}$$

On the other hand, *Real Power* is the capacity of the circuit for performing work at a time. This can only be calculated by measuring voltage and current simultaneously and multiplying them, and averaging over time:

$$RealPower \equiv \sum_{n=0}^{N-1} u(n) \times i(n) \tag{4}$$

The ratio of Real Power to Apparent Power is called the Power Factor (Eq. (5)) and refers to the ability of the electrical systems on the installation to convert electric current into useful workload such as heat or light.

$$Power\ Factor = Real\ Power / Apparent\ Power \tag{5}$$

We keep in mind that it is not always possible to determine the value of the real power by calculating the apparent power due to energy stored in the load and returned to the source, or due to non-linear loads that distort the wave shape of the current drawn from the source. In such cases the power factor will not be 100%. However, in most cases the apparent power can be considered an upper bound of the real power.

41.2. Power measurement sensor

Our solution is based on the usage of a non-invasive current sensor to measure the I_{RMS} of the installation by sampling the current signal. We prefer to use split core (open) current transformers (CT) instead of solid core (closed) ones, as they can be easily placed around existing wirings without disrupting the circuit. The split CT is an electric device that it is clipped around a wire (either phase or neutral), to provide and measure the amount of current passing through it. We need to note that the placement of the sensor around the cable of the electrical installation is crucial to the quality and the accuracy of the sensor's output, as its error rate depends heavily on the load that is attached to the system.

In our design we use the SCT-013-030 current sensor [8] that can detect *IAC* values from 0 to 30 A. The inductive current rate of the sensor is $N = 1800$ and the output is a proportional *V AC* signal with a maximum of 1 V to the 62 Ω built-in sampling resistance as it is depicted in Eq. (6). This sensor has been chosen due to the extended work made available by the OpenEnergyMonitor Project [9], its accuracy and ease of use.

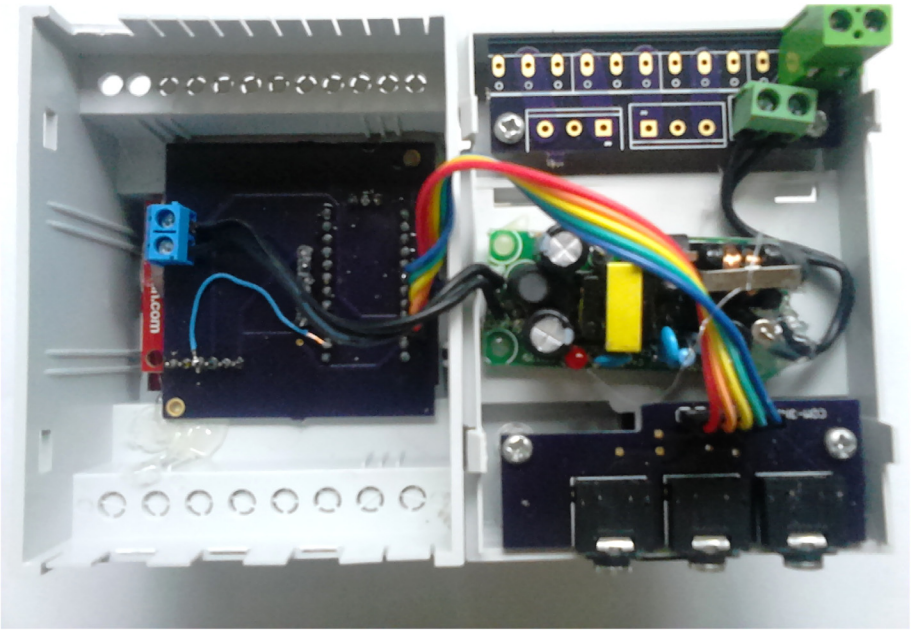

Fig. 1. PCB positioned inside its enclosure.

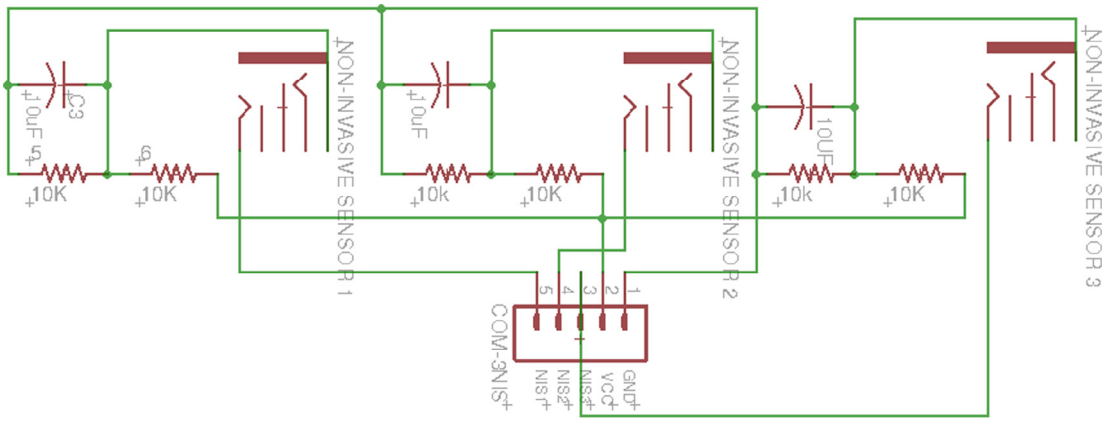

Fig. 2. Current measurement circuit schematic.

The output signal of the sensor is processed in time windows of 10 AC signal cycles to guarantee a statistically correct average value for the I_{RMS}. Based on European standards, all electrical installations have a frequency of 50 Hz, resulting in a sampling window of at least 200 ms to avoid aliases based on the Nyquist-Shannon sampling theorem. However, based on our experiments using a window below 250 ms does not provide accurate enough measurements, especially when comparing our results to those of commercially available products, or measuring tools.

In addition to the hardware sensor, we use a low-pass software filter on the sampled signal to remove the constant offset introduced, sample negative values of the signal and calculate the I_{RMS} value.

$$V_{read} = \frac{I \times 62}{1800} \tag{6}$$

Our non-invasive device does not measure the V_{RMS} of the installation.

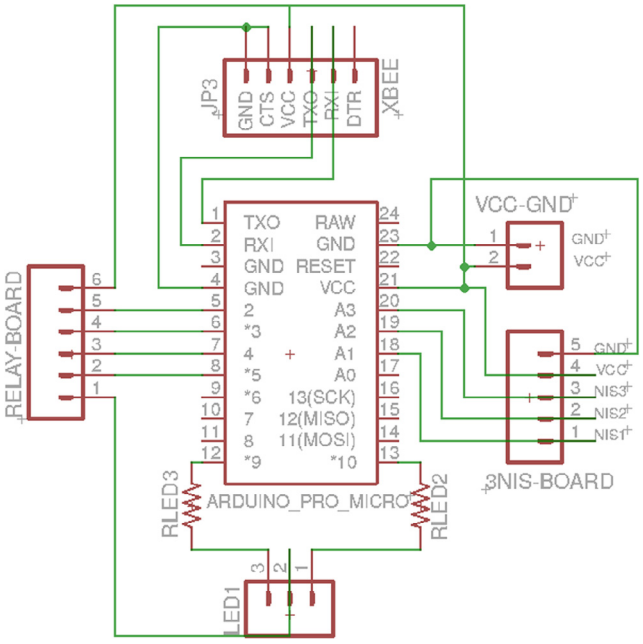

Fig. 3. Arduino-XBee module schematic.

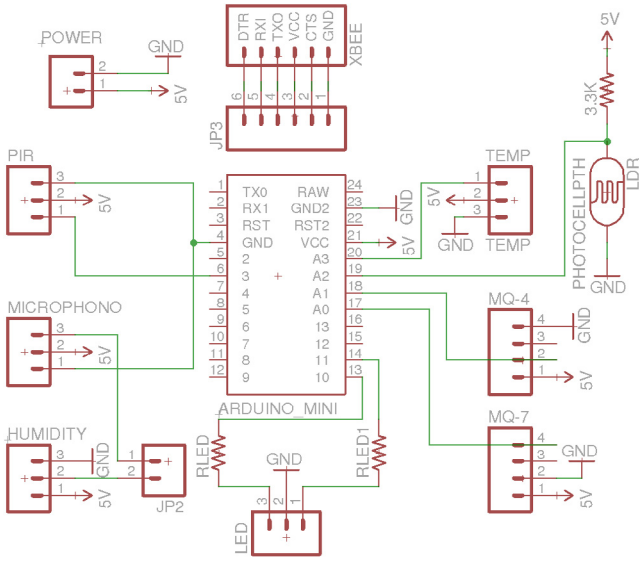

Fig. 4. Schematic of Analog environmental comfort meter (V1).

We assume a constant V_{RMS} for the specific electrical networks (around 230 V *A* in the EU). Indeed this hinders the overall accuracy of our measurements, however it minimizes any alterations on the distribution panels of schools buildings. We thus simplify the installation process of our system and thus reduce the installation costs. More importantly we limit any possible problems that can be caused by the operation of the device (e.g., a hardware failure) and do not affect the safety of the overall system.

4.1.3. System hardware design

The power consumption meter consists of three PCB blocks: a) the sensor board, b) the microcontroller board, and c) the wireless communication board. All three PCBs are specifically designed to be placed in the Z-108F [10] enclosure that can be mounted in DIN rails inside standard electrical distribution boxes. The current sensors are attached to the board using

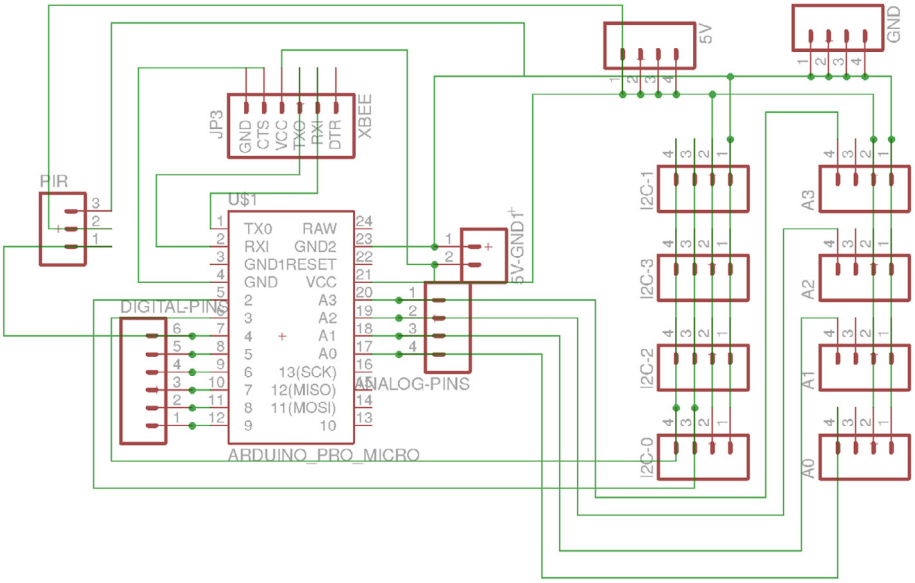

Fig. 5. Schematic of Digital environmental comfort meter (V2).

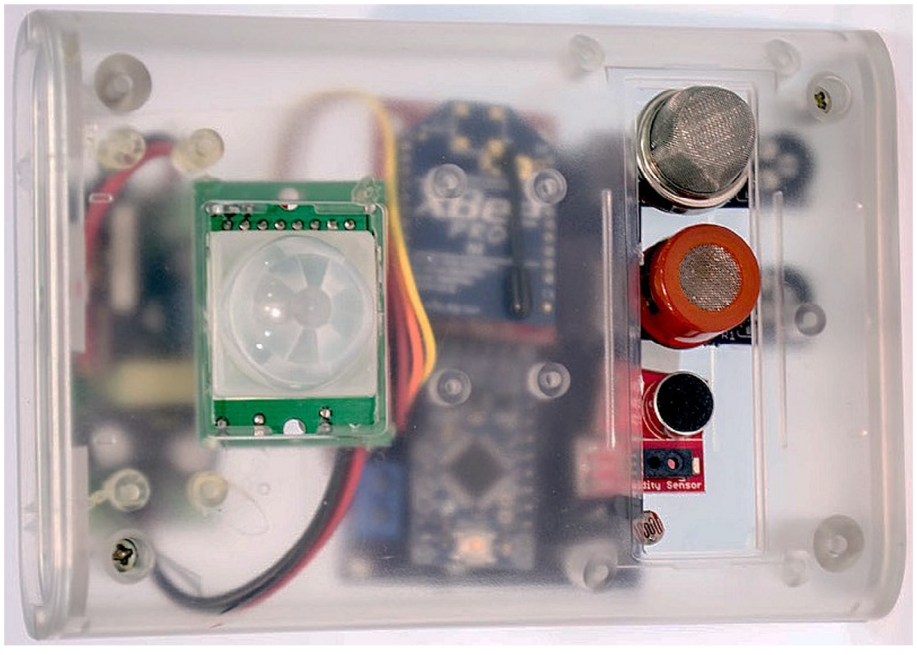

Fig. 6. Assembled environmental comfort meter device close-up.

Table 1
Power Meter BOM.

	Product	Qty	Cost (€)	Total (€)
1.	Z-108F Enclosure Details: Casing for DIN rail mounting	1	2.18	2.18
2.	Switching Power Supply Board Module Details: Power Supply 5 V 1 A	1	4.41	4.41
3.	PCB-1 Details: Current Measurement Circuit Board	1	3.24	3.24
4.	PCB-2 Details: Arduino-XBee module Board	1	6.60	6.60
5.	PCB-3 Details: Power connector Board	1	2.80	2.80
6.	Arduino Pro Micro compatibility Details: 5 V/16 MHz with ATmega32U4 on board	1	3.70	3.70
7.	XBee Pro S1, Maker Seedstudio Details: 60 mW Wire Radio Antenna – S1 (802. 15.4)	1	31.63	31.63
8.	XBee Explorer regulated Details: Connector shield for XBee module	1	7.79	7.79
9.	SCT-013-030 Details: Non-Invasive AC Current Sensor (30A)	3	6.95	20.85
10.	Capacitor 10uF Details: SMD;0805 10uF; 6.3 V; X5R; ±10%	3	0.01	0.04
11.	Resistor-1 KΩ Details:SMD;0805; 1 kΩ; 0.125 W; ±1%; −55,125 °C	8	0.01	0.08
12.	Resistor-10 KΩ Details: SMD; 0805; 10kΩ; 0.1 W; ±0.1%; 25 ppm/°C	2	0.01	0.02
13.	Screw for plastic Details: Used to screw the PCB-1 & PCB-2 position	4	0.01	0.04
14.	Terminal Block Details: 5 mm; angled 90°; 1.5 mm^2; ways:2; tinned; 16 A	1	0.14	0.14
15.	Pinhead male Details:2.54 mm 1 × 40 Pin Breakaway Straight Male	1	0.23	0.23
16.	Pinhead female Details: 2.54 mm 40-Pin Single-row Female Header	1	0.21	0.21
17.	Pluggable terminal block Details: Terminal Block 240 V AC/10 A	1	0.50	0.50
18.	Sensor Connector Details: Audio Jack 3.5 mm	3	0.11	0.33
19.	Jumper Wire Details: 2.54 mm, 5-pin, 10.16 cm	1	0.90	0.90
	Total cost per device			85.69

3.5 mm audio jack connectors. A 230 V *AC* switching power supply module is also included in the device to provide it with the 5 V *DC* regulated voltage that feed the microprocessor's circuit. Fig. 1 gives an image of the assembled device in its enclosure. Each electrical and board design file can be found on GitHub [11].

The MCU board is a standalone Arduino Pro Micro board [12] with an ATmega32U4 [13] and a USB connector. Running at 16 MHz and with an operating voltage of 5 V, it can be directly powered by its (*Vcc, GND*) inputs pins. The current measurement circuit connects to the Arduino board trough the analog pins 1, 2 and 3. Notice that the *V AC* output of the SCT sensor (±1 V) cannot be interfaced directly to the Arduino analog inputs, as the reverse voltages could damage the microcontroller. To avoid this, we include a voltage divisor that shifts up the voltage by 2.5 V to measure both positive and negative components of the waveform with a capacitor that eliminates high frequency noise. The schematic of the specific board is available in Fig. 2.

To interface the XBee Pro 60 mW-Series1 module [14] with Arduino Pro Micro through the SPI pins, we have designed a separate board presented in Fig. 3. The bidirectional SPI link allows the Arduino to issue commands and receive data from the XBee module. As the XBee module operates using 3.3 V, we use the SparkFun XBee Explorer [15] to ensure the 3.3 V regulation and signal level shifting.

4.2. Environmental comfort meter

Environmental comfort is directly related to the energy consumption of both public and private buildings. It is important to ensure that the occupants and visitors of a building are feeling as comfortable as possible inside their living and working spaces. Our environmental comfort meter focuses on monitoring the thermal, visual and aural comfort of occupants. Additionally, we also monitor occupancy, to better understand different activity levels throughout the day and its effects on energy consumption.

We describe here two versions of our environmental comfort meter: a) one using *analog sensors* for Arduino devices (V1 presented in Fig. 4) and b) a second one with *digital sensors* (V2 presented in Fig. 5), offering more accurate results but requir-

Table 2
Environmental Sensor Box V1 (Analog) BOM.

	Product	Qty	Cost (€)	Total (€)
1.	Arduino Pro Mini Details: MCU 5 V-16 MHz with ATmega328	1	2.00	2.00
2.	XBee Pro S1, Maker Seedstudio Details: 60 mW Wire Radio Antenna – Series 1 (802.15.4)	1	31.63	31.63
3.	XBee Explorer regulated Details: Connector shield for XBee module	1	7.79	7.79
4.	Grove-Sound Sensor Details: electret microphone with amplifier LM386	1	4.44	4.44
5.	Mini Photocell Details: Photosensitive resistor	1	0.20	0.20
6.	Temperature Sensor Details: MN-EB-LM35A	1	3.11	3.11
7.	Humidity Sensor Details: HIH-4030 Breakout	1	16.05	16.05
8.	PIR-Motion Sensor Details: Passive Infrared Detection- Large leg	1	5.21	5.21
9.	Resistor-10 KΩ Details: SMD; 0805; 10kΩ; 0.1 W; ±0.1%; 25 ppm/°C	2	0.01	0.02
10.	Enclosure Details: ABS Clear Enclosure Case for pcDuino	1	7.75	7.75
11.	Switching Power Supply Board Module Details: Power Supply 5 V 1 A	1	4.41	4.41
12.	PCB Details: Environmental Comfort Meter V.1 Board	1	7.80	7.80
13.	Jumper Wire - 0.1″, 3-pin, 4″ Details: Connector cable - 2.54 mm, 3-pin, 10.16 cm	3	0.90	2.70
14.	Jumper Wire - 0.1″, 4-pin, 4″ Details: Connector cable - 2.54 mm, 5-pin, 10.16 cm	2	0.90	1.80
15.	Terminal Block Details: 5 mm; angled 90°; 1.5mm2; ways:2; tinned; 16A	1	0.14	0.14
16.	Pinhead male Details:2.54 mm 1 × 40 Pin Breakaway Straight Male Header	1	0.23	0.23
17.	Pinhead female Details: 2.54 mm 40-Pin Single-row Female Header	2	0.21	0.42
	Total cost per device			95.70

ing a more complicated design. The boards allow for an XBee Pro 60 mW communication module to be connected using the SPI pins to the Arduino microcontroller, for exchanging wireless messages with the gateway of the system. A view on the final device in its enclosure is available in Fig. 6.

Apart from their differences, both versions use a common subset of sensor types and capabilities. To detect the overall noise exposure, we include a sound pressure level sensor [16]. This sensor detects the noise level of the environment in two stages: (1) capturing noise samples using a simple electret microphone characterized by a specific sensitivity, and, (2) using an amplifier to provide output that can be read using an Arduino analog input.

The sensitivity of an analog microphone is typically specified in logarithmic units of dBV (decibels with respect to 1 V), giving an estimate of how many Volts the output signal is for a given sound pressure level. For our specific sensor 50 dBV correspond to a sensitivity of S = 316.22 V/Pa while the preamplifier was set to a gain of 26 dB (G = 19.95) using the onboard available potentiometer. These low cost sound pressure level sensors are generally used to detect the presence of sound in the environment, but in our software we calculate the sound level in dB by processing multiple samples of the analog output of the sound pressure level sensor. To calculate the absolute sound pressure level L_p (or SPL) we first use the measured v_{rms} to calculate the p_{rms} in Pa using Eq. (7).

$$P_{rms} = \frac{u_{rms}}{G}/S \tag{7}$$

We then calculate the L_p in dB using Eq. (8) with p_{ref} being the reference level of SPL (the threshold of SPL for the human voice).

$$L_P = 20\log_{10}\frac{P_{rms}}{P_{ref}} = 20\log_{10}P_{rms} - 20\log_{10}P_{ref} \tag{8}$$

To detect occupancy, we use a PIR Sensor that detects motion. The PIR sensor outputs an event that is used as an interrupt trigger in the Arduino. The PIR sensor used [17] is powered with a stable voltage of 5 V and its output pin is directly connected to an Arduino digital input pin.

4.2.1. Analog version (V1)

To capture thermal comfort, we measure temperature and relative humidity (percentage of water vapor in the air) using:

the LM35DZ [18] precision integrated-circuit temperature sensor that outputs voltage linearly proportional to the °C temperature. The sensor is supplied with a 5 V *DC* voltage and its output voltage is measured using an Arduino analog input pin. The temperature range of the sensor is from 2 to 150 °C on this basic configuration, which is enough for measuring indoor temperature conditions. The gain for this specific sensor is 10 mV per each °C, allowing for a straight conversion of the input voltage to the observed temperature. The sensor provides an accuracy of ±0.25 °C without any calibration or trimming.

Table 3
Environmental Sensor Box V2 (Digital) BOM.

	Product	Qty	Cost (€)	Total (€)
1.	Arduino Pro Micro compatible device Details: MCU 5 V/16 MHz with an ATmega32U4	1	3.70	3.70
2.	XBee Pro S1, Maker Seedstudio Details: 60 mW Wire Radio Antenna – Series 1 (802.15.4)	1	31.63	31.63
3.	XBee Explorer regulated Details: Connector shield for XBee module	1	7.79	7.79
4.	Grove-Temperature & humidity sensor Details: THO2 digital sensor	1	7.70	7.70
5.	Grove-Digital Light sensor Details: I2C light-to-digital converter TSL2561	1	8.97	8.97
6.	Grove-Sound Sensor Details: electret microphone with amplifier LM386	1	4.44	4.44
7.	PIR-Motion Sensor Details: Passive Infrared Detection- Large length version	1	5.21	5.21
8.	Grove universal 4 pin connector	0.5	1.36	0.68
9.	Pinhead male Details:2.54 mm 1 × 40 Pin Breakaway Male Header	1	0.23	0.23
10.	Pinhead female Details: 2.54 mm 40-Pin Single-row Female Header	2	0.21	0.42
11.	Terminal Block Details: 5 mm; angled 90°; 1.5mm2; ways:2; tinned; 16 A	1	0.14	0.14
12.	Switching Power Supply Board Module Details: Power Supply 5 V 1 A	1	4.41	4.41
13.	External PCB switch (10 item pack) Details:115 V–230 V 4.2 cm × 1.4 cm × 1.7 cm Slide Switch	0.1	3.78	0.38
14.	Cable plug (male/female) Details: Cable connector for easy installation	1	0.99	0.99
15.	Enclosure Details: ABS Clear Enclosure Case for pcDuino	1	7.75	7.75
16.	PCB Details: Environmental Comfort Meter V2 Board	1	7.83	7.83
	Total cost per device			92.27

Table 4
Gateway BOM.

	Product	Qty	Cost (€)	Total (€)
1.	IBoard Details: Arduino ATMega328 with WIZnet POE Ethernet	1	15.86	15.86
2.	XBee Pro S1, Maker Seedstudio Details: 60 mW Wire Radio Antenna – Series 1 (802.15.4)	1	31.63	31.63
	Total cost per device			47.49

Table 5
Accuracy and operation ranges of environmental sensors.

Sensor	Range	Accuracy
LM35DZ	2 … 150 °C	±0.25 °C
TH02 Temperature Sensor	0 … 70 °C	±0.5 °C
NetAtmo Temperature Sensor	−50 … 65 °C	±0.3 °C
HIH-4030	0 … 100 %RH	±3 %RH
TH02 RH Sensor	0 … 100 %RH	±4.5 %RH
NetAtmo RH Sensor	0 … 100 %RH	±3 %RH

the HIH-4030 [19] relative humidity sensor. Its output voltage, similarly to the LM35DZ sensor, is near linear to the actual relative humidity in *%RH* and is sampled again using an Arduino analog input pin and converted to the *%RH*, based on the characteristic function provided by the sensor's datasheet. We note that the environmental temperature strongly affects humidity sensors, and we therefore use the temperature measured to compensate for the temperature effects and provide more accurate *%RH* values.

Both sensors are placed outside the device's enclosure to reduce their exposure to the residual heat of different heat producing elements, like the 5 V transformer used to power the device.

To capture visual comfort, we measure the luminosity levels of the room using luminance. Building regulations and standards use luminance to specify the minimum light levels for specific tasks and environments and it is defined as the amount of light falling on a surface and measured in Lux (lumen/m^2). In our devices we measure luminance with a photo-resistor [20]. A variable resistance that changes its resistance based on the amount of light that falls on it. The relationship between the resistance *Rl* and light intensity Lux for a typical Light Dependent Resistor is approximated using the Eq. (9). The photocell is connected to the 5 V *DC* through a 10k resistor and the photoresistance value can be calculated from the input voltage in an Arduino analog input. The calculated values are not an exact value of the luminance observed in the room but an approximation, as the lighting conditions in a room can vary heavily.

$$R_l = \frac{500}{Lux}(K\Omega) \tag{9}$$

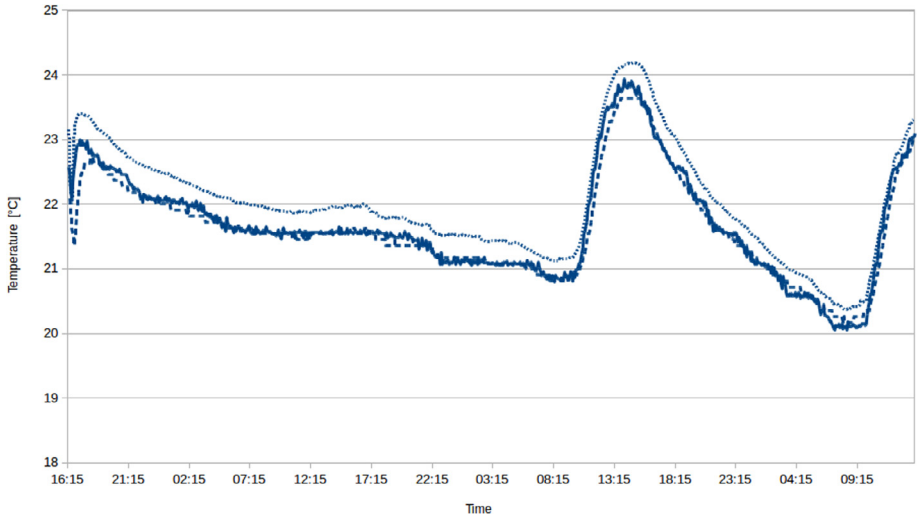

Fig. 7. Temperature (in °C): LM35 Analog sensor (continuous line), TH02 digital sensor (dotted line) and NetAtmo device (dashed line).

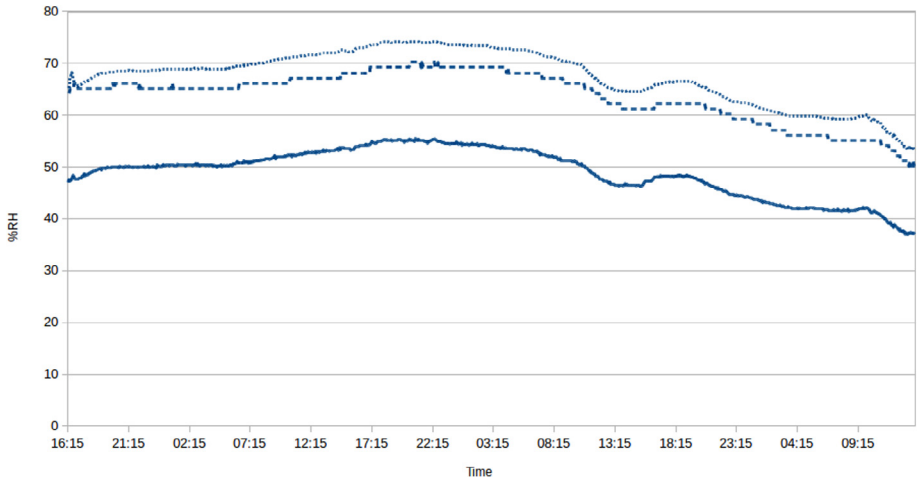

Fig. 8. Relative Humidity (in %RH): HIH-4030 Analog sensor (continuous line), TH02 digital sensor (dotted line), NetAtmo (dashed line).

4.2.2. Digital version (V2)

As an alternative to the analog temperature and humidity sensors, we used the digital THO2 I2C sensor. This sensor integrates both sensors and an analog-to-digital converter, signal processor, and calibration data together with an I2C host interface. It provides reliable readings for humidity between 0–100 *%RH* and 0−70 °C, more than enough for indoor environments. The main advantage in comparison to the analog sensor is that it is factory-calibrated, with the calibration data stored in an on-chip non-volatile memory. This reduces the time needed to perform the calibration necessary for the analog sensors and ensures that the sensors are fully interchangeable. In addition, the quantification error due to the analog to digital conversion is reduced as Arduino only offers an 8 bit ADC to convert while the sensor has an internal ADC of 14 bits for the temperature and 12 bits for the humidity. Also, the sensor is directly connected to the microcontroller without the need for any electronic components reducing the noise interference and thus guaranteeing an accuracy of ±0.5 °C for temperature and ±4.5 *%RH* for humidity in indoor conditions.

In the digital version (V2), for measuring the luminosity we used the TSL2560 [21] photo-diode I2C sensor. It offers a sensing range of 0.1 to 40,000 lux with 16 bit digital resolution, as well as lower quantification errors and reduced noise interference when compared to the photo-resistor included in the analog version (V1). Both digital modules are connected to the same I2C bus requiring us to open and close each sensor to get valid measurements. For this the THO2 module possesses a pin that controls the sensor from an Arduino output pin while the TSL2560 sensor is directly powered by on/off software commands. The specific libraries for controlling of each module are open source and can be downloaded from the manufacturers' websites.

5. Bill of materials and prices

The complete BOMs for all devices are available in this section. Table 1 concerns the power consumption meter, Tables 2 and 3 list the elements for the two versions of the environmental comfort meter, while Table 4 describes the components needed for the gateway device.

Regarding the cost of the devices, the XBee S1 Pro module used for the wireless communication between the IoT nodes is almost 1/3 of the device's total cost. In our design, we use the XBee Pro modules to achieve better communication ranges inside buildings with thick walls or metal enclosures, such as the electrical junction boxes of buildings. By using less powerful radios the total cost of the devices can be further reduced by at least CC10. The range of the normal XBee S1 modules is at most 100m in an unobstructed line of sight, while the Pro modules offer a 1 Km range. The software implemented on the IoT nodes allows us to extend the network range by using each device as a hub for wireless communication inside the buildings in an ad hoc manner.

In comparison to similar off-the-shelf devices, such as NetAtmo Weather station [22], our digital environmental comfort meter (V2 device) costs about 50% less. When considering also the costs for the gateway devices (for connecting the sensor to the cloud), then the total costs of our platform are a lot lower than 50%.

6. Results

In this section, we showcase the operation of our devices and compare them with commercial off-the-shelf solutions. During all experiments our devices are set to gather data every 20 s and transmit them to our back-end service for storage. For the comparison with the commercial products we use 5 min aggregated values of the transmitted results to be as fair as possible with the data available from the APIs of used products. The accuracy of all sensors tested is presented in Table 5.

6.1. Environmental comfort meter

For the evaluation of the performance of our environmental comfort meter we present the reported values of our environmental comfort meter (both V1 and V2) and compare them with a NetAtmo weather station. During the data collection period all 3 devices placed in the same room in close distance. It is important to note here that in our devices the temperature and humidity sensors are placed out of the enclosure to limit the effects of any heat sources due to inadequate ventilation. To capture various levels of indoor temperature and humidity the experiment's duration was 48 h.

6.1.1. Temperature

The temperature measurements from all 3 devices are available in Fig. 7. All time series follow the same pattern with minor deviations due to the accuracy of the sensors as there is a constant difference of less than 0.5 °C between them. The values recorded by the NetAtmo are tightly close to the analog LM35 sensor, while the TH02 digital sensor has a relatively stable offset (less than its accuracy) during the whole experiment. On the other side the values of the TH02 sensor are more stable and differ less from each other especially when compared to LM35 digital sensor that has a higher quantification error as the Arduino analog to digital converter module is used to sample the sensor's output.

6.1.2. Relative humidity

A similar graph for the Relative Humidity values collected during the 48 h of the experiment is presented in Fig. 8. We once again can note that all of them follow the same pattern with a different offset. The NetAtmo device provides values that are closer to the digital sensor during this experiment while the analog HIH-4030 sensor provides values of the same pattern but with a constant offset of −13 *%RH*. This offset can be justified by the lack of calibration for the analog sensor. The behavior of the HIH-4030 sensor is not linear to the humidity of the environment and its approximation based on the datasheet information requires specific procedures for calibration for each sensor using multiple reference it points.

6.2. Power consumption meter

To evaluate the performance of our power consumption meter we follow the same approach with the environmental comfort meter by comparing the performance of our device with a Meazon Energy Consumption meter. Meazon offers a cloud based solution (Bizy Platform) to monitor the power consumption measured by its devices and in 15 min intervals, although power is measured in a much higher rate and interfaces to extract the data gathered in a reporting format. During the experiment we measured the power consumed by three different devices: a 22 inch LCD monitor, a desktop computer

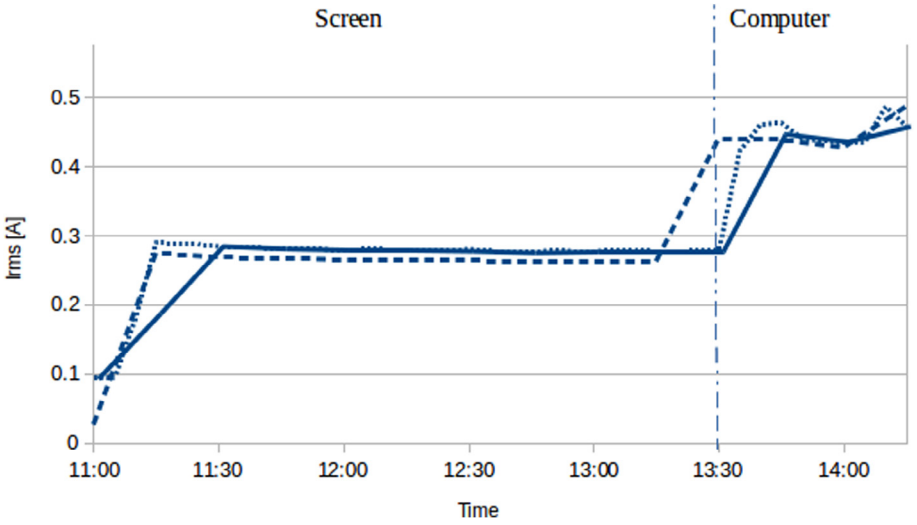

Fig. 9. Electrical Current for low power devices (in A): power consumption meter 5 min average values (continuous line), power consumption meter 15 min average values (dotted line), Meazon meter 15 min average values (dashed line).

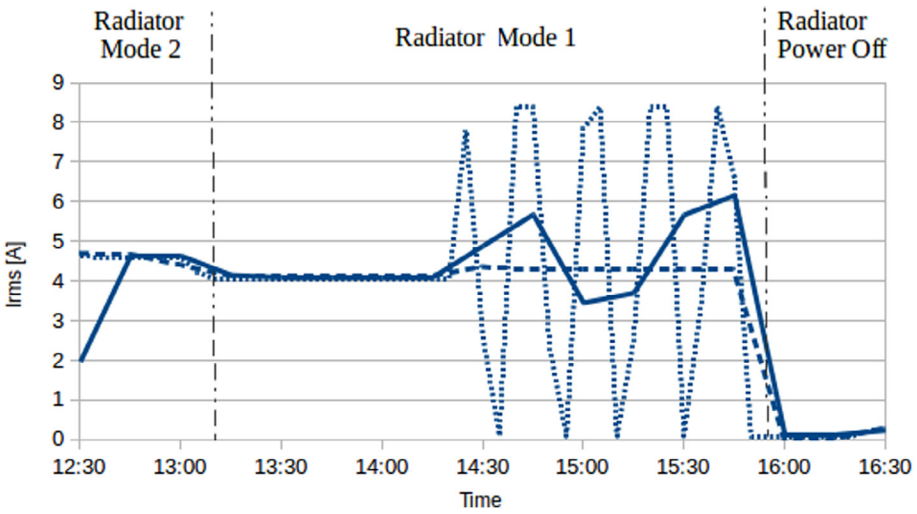

Fig. 10. Electrical Current for high power devices (in A): power consumption meter 5 min average values (continuous line), power consumption meter 15 min average values (dotted line), Meazon meter 15 min average values (dashed line).

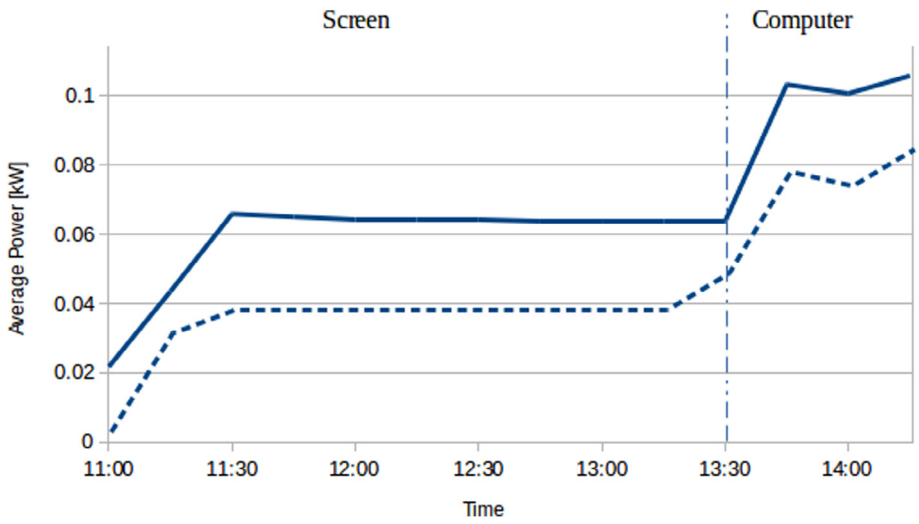

Fig. 11. Screen and Computer Power Consumption (in kW): power consumption meter 15 min average values (continuous line), Meazon meter 15 min average values (dashed line).

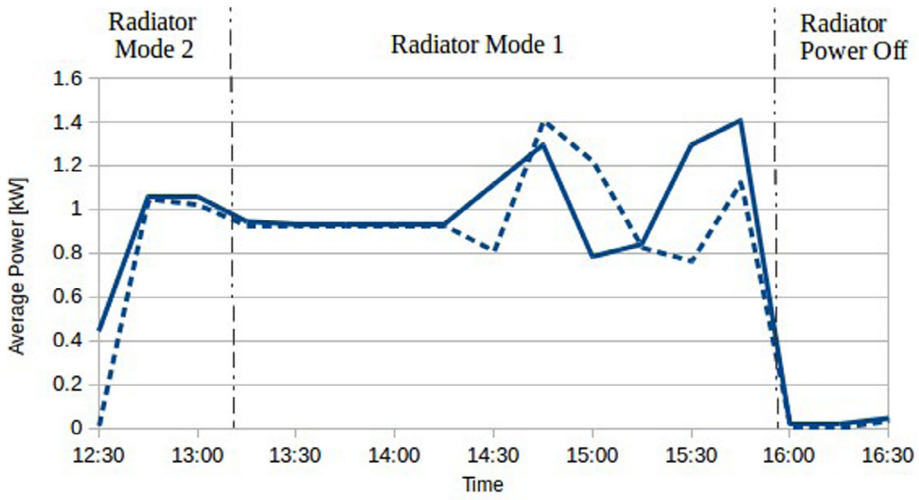

Fig. 12. Radiator Power Consumption (in kW): power consumption meter 15 min average values (continuous line), Meazon meter 15 min average values (dashed line).

and an electrical radiator on 2 different heating levels. The measured average electrical current I_{RMS} values from both power consumption meters is presented in Figs. 9 and 10 .

Results are presented in 5 and 15 min intervals for our power consumption meter for better comparison with the data from the Meazon service. Both power meters provide measurements very close to each other but a point to point comparison is not possible due to the way we can gather data from the Meazon cloud interface. For the low consumption devices (monitor and desktop computer) we can observe a small difference in the measured values that is caused due to the fact that the SCT-013 sensor has displays a noise signal of around 80 mA that is affecting heavily the measurements from low consumption devices. For the high consumption devices (portable radiator in two operation modes) we observe that the values reported are almost identical in the first part of the experiment (with some small differences due to the 15 min averaging of the Meazon service). For the second part, when the radiator reaches its target temperature and starts to switch on and off periodically we observe that the Meazon device reports a stable average measurement while our similar to our data (when using 15 min average values). When reviewing the 5 min average data we see that consumption shifts between 8 and 0 Amperes with an average of 4.5.

The Meazon meter also measures the V_{RMS} in real time and uses it to calculate the power consumption in real time and then presents aggregated values for every 15 min. On the other hand our power consumption meter calculates the average power using a stable V_{RMS} = 230 V *AC* (as mentioned in 4). The power consumption calculated on both devices is available in

Figs. 11 and 12Fig. 12. These figures prove that the accuracy of calculating the power consumption with a constant V_{RMS} is sufficient in higher consumption cases as the two lines in Fig. 11 differ mainly due to the noise sampled by the SCT-013 while in Fig. 12 are sufficiently close to each other. The differences in the last part of the graph in Fig. 12 are mainly due to the 15-min interval used by the Meazon meter, as we do not know exactly when each averaging window starts.

7. Conclusions

We presented a low-cost solution for implementing energy consumption and environmental monitoring using an open source IoT infrastructure, aiming specifically for educational buildings. We discussed in detail our hardware design, which along with the software running on the devices is available online on GitHub [7]. Our initial version of the devices utilized mostly analog sensors, which we replaced with digital ones in our second revision to achieve a more stable sensor outcome and achieve an easy setup without the need for a complex calibration phase.

As we follow the open source approach, we provide a basis that can be reused, modified and extended by similar projects around the world. Although there are numerous related works, there are few, if any, similar examples aiming to be used within an educational context. We also conducted a set of experiments showcasing that our devices in general perform in a similar manner to commercially available platforms. However, such devices provide more robust hardware, together with certifications for their operation. This is an aspect that we plan to work on in the future, along with other aspects such as e.g., providing educational lab material based on the hardware and adding other sensing components for a more complete hardware approach.

Acknowledgments

This work has been partially supported by the EU research project Green Awareness In Action" (GAIA), funded under contract number 696029. This document reflects only the authors' view and the EC and EASME are not responsible for any use that may be made of the information it contains.

Appendix A. Supplementary data

Supplementary data associated with this article can be found, in the online version, at http://dx.doi.org/10.1016/j.ohx.2017.02.002.

References

[1] Europe's buildings under the microscope: A country-by-country review of the energy performance of buildings, Tech. rep., Buildings Performance Institute Europe (BPIE), ISBN: 9789491143014, 2011.
[2] E. R. L. P. O. of the European Union, Funding of education in Europe 2000-2012: The impact of the economic crisis, Tech. rep. (European Commission/EACEA/Eurydice, 2013).
[3] Green Awareness In Action,< http://gaia-project.eu>.
[4] The Hour of Code global movement. URL: <https://hourofcode.com>, (last accessed on Nov. 2016).
[5] Arduino XBee library. URL: <https://github.com/andrewrapp/xbee-arduino>, (last accessed on Nov. 2016).
[6] mksense, Arduino XBee radio library. URL: <https://github.com/mksense>, (last accessed on Nov. 2016).
[7] CTI RU1, Arduino XBee routing library. URL: <https://github.com/CTI-ru1/sensorApps/tree/master/Arduino/Routing>, (last accessed on Nov. 2016).
[8] SCT-013-030 current sensor. URL <https://nicegear.co.nz/obj/pdf/SCT-013-datasheet.pdf> , (last accessed on Nov. 2016).
[9] OpenEnergyMonitor, Emontx project. URL: <http://openenergymonitor.org/emon/>, (June 2014).
[10] COMBIPLAST, 2811-CP-Z-107-ABS. URL: <http://tinyurl.com/jucw8s6>, (last accessed on Nov. 2016).
[11] CTI RU1 IoT Nodes. URL: <https://github.com/CTI-ru1/iot-nodes>, (last accessed on Nov. 2016).
[12] Arduino Pro Micro, URL: <https://github.com/sparkfun/ProMicro>.
[13] ATMega32U4. URL: <http://cdn.sparkfun.com/datasheets/Dev/Arduino/Boards/ATMega32U4.pdf>, (last accessed on Nov. 2016).
[14] Digi XBee pro 60 mw – series 1. URL: <http://www.digi.com/resources/documentation/Digidocs/90000982/Default.htm>, (last accessed on Nov. 2016).
[15] Sparkfun XBee explorer regulated. URL: <https://github.com/sparkfun/XBeeExplorerRegulated>, (last accessed on Nov. 2016).
[16] Grove - loudness sensor. URL: <http://wiki.seeed.cc/Grove-LoudnessSensor/>, (last accessed on Nov. 2016).
[17] Pir motion sensor. URL: <http://wiki.seeedstudio.com/wiki/PIRMotionsensormodule>, (last accessed on Nov. 2016).
[18] LM35 temperature sensor. URL: <http://www.ti.com/lit/ds/symlink/lm35.pdf>, (last accessed on Nov. 2016).
[19] HIH-4030 humidity sensor. URL: <https://www.pololu.com/file/0J324/HIH-4030-datasheet.pdf>, (last accessed on Nov. 2016).
[20] CDS photoconductive cells. URL: <https://cdn.sparkfun.com/datasheets/Sensors/LightImaging/SEN-09088.pdf>, (last accessed on Nov. 2016).
[21] TSL2560 light to digital converter. URL: <http://www.alldatasheet.com/datasheet-pdf/pdf/203058/TAOS/TSL2561T.html>, (last accessed on Nov. 2016).
[22] Netatmo weather station, <https://www.netatmo.com/product/weather>.